UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
DEPARTAMENTO DE ESTATÍSTICA

PÓS-GRADUAÇÃO EM ESTATÍSTICA

LORC: Classificação supervisionada baseada em grafos esparsos, robusta para dados com ruído no rótulo

Letícia Cavalari Pinheiro

Tese de Doutorado

BELO HORIZONTE
26 de Junho de 2015

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
DEPARTAMENTO DE ESTATÍSTICA

Letícia Cavalari Pinheiro

LORC: Classificação supervisionada baseada em grafos esparsos, robusta para dados com ruído no rótulo

Trabalho apresentado ao Programa de PÓS-GRADUAÇÃO EM ESTATÍSTICA do DEPARTAMENTO DE ESTATÍSTICA da UNIVERSIDADE FEDERAL DE MINAS GERAIS como requisito parcial para obtenção do grau de Doutor em ESTATÍSTICA.

Orientador: *Prof. Dr. Renato Martins Assunção*

BELO HORIZONTE
26 de Junho de 2015

Agradecimentos

Por mais distante que pareça, sempre chega... E chegou o dia em que, depois de 24 anos, me despeço dessa escola onde muito aprendi sobre a vida: a UFMG. Escola em que aprendi a ser curiosa, a estudar, a ter objetivos e buscá-los, a fazer e valorizar amigos de verdade, a aproveitar as oportunidades boas que a vida nos oferece, a encarar os obstáculos que aparecem no caminho. E assim foram muitos ciclos que começaram como se o final estivesse tão distante... e de repente se fecharam, para um novo ciclo começar.

Foi assim quando cheguei no Centro Pedagógico, com meus 7 aninhos. Lá tive ótimos professores que me fizeram gostar dos estudos e é incrível pensar quanto aquele tempo foi bom e importante. Nessa época fiz meus amigos de toda a vida, aos quais eu não poderia deixar de agradecer. Com eles tudo começou e são eles que até hoje me apoiam, me dão força, me fazem rir e me deixam com o coração tranquilo por saber que tenho amigos desde sempre e para sempre. Depois de 8 anos, este ciclo se fechou. Me lembro como se fosse ontem da mudança para o Coltec, um colégio de altíssima qualidade que nos formaria como pessoas responsáveis e capazes de cuidarem-se sozinhas. E assim o Coltec me preparou não só para a Universidade, mas para a vida. Outras grandes amizades foram conquistadas, e as antigas foram conservadas e fortalecidas. E ao fechar esse ciclo, chegava uma época cheia de dúvidas, em que um turbilhão de coisas passava em minha mente e eu precisava, naquele momento, escolher o que eu queria fazer para sempre.

Escolhi a Matemática Computacional. Confesso que muitas vezes pensei que tinha feito a escolha errada... me deparei com alguns professores sem boa vontade de ensinar, com matérias que eu pensava que não entenderia nunca, com pessoas muito diferentes de mim. Mas também encontrei professores inspiradores, descobri a alegria em entender aquelas coisas que eu achava que não entenderia nunca, percebi o prazer do conhecimento. E não é que também durante a graduação fiz grandes amigos? E mais esse ciclo de 4 anos se fechou.

Sem vontade de deixar essa minha segunda casa para trás, iniciei o mestrado, já trabalhando no Laboratório de Estatística Espacial (LESTE), e logo continuei com o doutorado. Quanta saudade eu tenho dos meus colegas e companheiros de trabalho e de estudo dessa fase, e até mesmo dos finais de semana e madrugadas estudando até o ponto de cair na gargalhada por não aguentar mais. Conseguiamos, apesar de tanta responsabilidade, levar com leveza e alegria. Tenho um grande carinho por esses amigos e digo que eles foram imensamente importantes para que eu esteja aqui, escrevendo esse agradecimento.

Ainda durante o doutorado, tive a grande oportunidade de assumir o cargo de pesquisadora no René Rachou (Fiocruz), que a partir deste momento passa a ser minha nova segunda casa. Também tive a sorte de encontrar ótimas pessoas, com as quais muitas vezes já pude rir e desabafar, que me deram um voto de confiança e que espero que sejam sempre meus parceiros

de trabalho, de conversas e de amizade.

Depois de todos esse ciclos, agradeço a Deus por ter me proporcionado todas essas oportunidades, por ter me dado força e sabedoria para trilhar meu caminho e colocar tantas pessoas especiais em minha vida. Agradeço muito a todos esses amigos que citei, que foram e são extremamente importantes para mim. Também agradeço ao Marcos Prates, que além de grande amigo foi quem me levou para o LESTE, confiando em meu trabalho, e que sempre me ajudou com dicas muito importantes. Ao Renato, que logo após a minha graduação entrou em meu caminho, e a partir daí já são 8 anos de convivência regada a incentivo, compreensão, apoio, inspiração e respeito. Algumas lágrimas enxugadas e muitas conquistas comemoradas. Como um pai "acadêmico"ele soube conduzir minha formação como pesquisadora com uma maestria única, e hoje merece toda minha gratidão e minha eterna admiração pelo grande profissional que é, e que tive a sorte de ter como orientador.

E durante todas essas etapas tive meu alicerce: minha família. A eles, que sempre me incentivaram e apoiaram, se orgulharam, estiveram ao meu lado durante toda a vida guiando meu caminho com carinho, cuidado, conselhos e força, meu enorme agradecimento. Minha mãe é meu colo mais aconchegante, minha amiga e defensora. Sempre fez tudo por mim e soube superar todas as dificuldades enfrentadas, me criando para a vida de uma forma admirável, demonstrando seu amor e carinho incondicionais. Meu pai, meu espelho de pessoa batalhadora e forte, que mesmo com a distância e com todos os compromissos nunca deixou de estar presente em minha vida me cercando de amor e confiança. Minha irmã Isabela, que faz com que todas as barreiras que a vida lhe colocou se tornem pequenas diante da vontade e alegria de viver, me inspira diariamente a dar valor às pequenas coisas e a encarar a vida de frente, fazendo de cada limão uma deliciosa limonada, de preferência uma *pink lemonade*. Meu irmão Ricardo, meu grande amigo e parceiro, que me nutre de carinho, conhecimento, piadas e poemas, e que costuma provocar meus sorrisos sinceros. Meu irmão Alexandre, com quem adoro conversar e ouvir algumas de suas inúmeras histórias de vida, que me transmite tranquilidade, admiração, carinho e amor.

Além de ter essa família tão especial, hoje ainda tenho mais uma família que a vida me deu de presente. Meu marido Marquinhos, que me cobre de amor e carinho todos os dias me fazendo sentir a pessoa mais especial e feliz do mundo. Meu amor, te agradeço muito não só pela força durante o doutorado, mas principalmente por ser meu melhor companheiro, por manter o sorriso constante em meu rosto, por sempre me apoiar e incentivar e por fazer da minha vida melhor de ser vivida. E à família dele, que hoje considero minha, não posso deixar de agradecer por todo o carinho e cuidado com que me tratam e por torcerem sempre por mim.

O doutorado, que está sendo encerrado agora, foi sendo construido enquanto o caminho da minha vida ia tomando novos rumos, muitas vezes inesperados. Hoje digo que tudo valeu a pena, e muito. Quantos lugares novos foram conhecidos, quantas pessoas queridas entraram em minha vida, quantas experiências inigualáveis foram vividas. Experiências... disso é contruída a vida. Espero que tenham sido só o começo de uma longa caminhada que continuará repleta de novidades, desafios, conquistas, pessoas boas, e experiências.

"Se os senhores disserem que tudo isso também pode ser calculado pela tabela - o caos, a treva, a maldição, de modo que a mera possibilidade de cálculo prévio pare tudo e a razão triunfe -, então nesse caso o homem ficará propositalmente louco, para ficar privado da razão e defender sua opinião!"

—DOSTOIEVSKI (Notas do Subsolo, 1864)

Resumo

Este trabalho apresenta e desenvolve novas metodologias para classificação supervisionada, baseadas em grafos esparços. A idéia inicial é utilizar as instâncias do conjunto de dados de treinamento do modelo para construir uma árvore geradora mínima (AGM) a partir das distâncias entre atributos e, posteriormente, obter uma partição do grafo ao podar arestas desta AGM utilizando uma medida de dissimilaridade calculada a partir dos rótulos. Essa partição definirá as regiões de classificação que buscam equilibrar grandes homogeneidades internas e grande heterogeneidade entre elas, proporcionando bons resultados de posteriores classificações de instâncias com rótulos desconhecidos. Um grande avanço apresentado pela metodologia desenvolvida neste trabalho é a potencial melhora na classificação quando o conjunto de dados de treinamento apresenta ruído no rótulo. Este tipo de ruído nos dados é bastante comum e acarreta prejuízos no desempenho de métodos tradicionais de classificação supervisionada. Basicamente, este trabalho explora os temas de classificação supervisionada e de ruído no rótulo, apresenta uma metodologia de classificação com 4 variações possíveis, proporcionando possibilidades de adequação aos dados, demonstra a eficiência do método em determinados tipos de conjuntos de dados e comprova a qualidade da classificação realizada através de comparações com outros métodos popularmente utilizados. Os resultados são promissores.

Palavras-chave: Classificação Supervisionada; Dados com Ruído no Rótulo; Árvore Geradora Mínima.

Abstract

This thesis presents the development of a new supervised classification method based in sparse graphs. The basic idea is to learn from data instances to build a minimum spanning tree (MST), based on the distances between attributes. Based on a dissimilarity measure calculated from the labels, we obtain a graph partition by pruning the MST edges. This partition defines the classification regions that seek to balance major intra-region homogeneity and great inter-region heterogeneity, providing good results for posterior classifications of instances with unknown labels. A great advancement presented by the developed methodology is the potential classification improvement when the training datasets have label noise. This type of noise is common and impairs the performance of most classification methods. This thesis includes a study about supervised classification and label noise data, the development of a new classification methodology with 4 possible variations making possible to adapt to diferent datasets, the proof of its efficiency under some assumptions, and the quality verification based on comparisions with other popular methods. The results are promising.

Keywords: Supervised Calssification; Label Noise Data; Minimum Spanning Tree.

Sumário

Lista de Figuras

Lista de Tabelas

CAPÍTULO 1

Introdução

A Aprendizagem de Máquina (*Machine Learning*, em inglês) [Izenman, 2008] tem como principal objetivo a criação de sistemas computacionais e algoritmos que possam "aprender"a partir da experiência prévia. Uma máquina aprende quando ela tem o poder de acumular experiência (a partir de dados, por exemplo) e desenvolver novo conhecimento de forma que sua performance melhore com o tempo. Esta idéia de aprender com a experiência é central em vários tipos de problemas de aprendizagem de máquina, especialmente os que envolvem classificação. O principal objetivo deste tipo de problema é encontrar uma forma de classificar um exemplo futuro. A classificação se baseia nos atributos desse futuro exemplo juntamente com o conhecimento obtido de uma amostra de treinamento composta por exemplos similares. A classe (ou rótulo) de cada exemplo é completamente determinada e conhecida e o número de classes é finito e conhecido.

Dentro da área de aprendizagem de máquina, as duas categorias mais relevantes são "Aprendizagem Supervisionada"e "Aprendizagem Não Supervisionada". Focaremos na aprendizagem supervisionada, que consiste em problemas nos quais o algoritmo recebe um conjunto de variáveis explicativas (contínuas ou categóricas) e uma variável resposta. Com esses dados, ele tenta encontrar uma função das variáveis de entrada para aproximar a resposta conhecida. Se a resposta é categórica, temos um problema de "Classificação". O principal objetivo deste tipo de problema é encontrar uma forma de classificar um exemplo futuro, baseada nos atributos desse futuro exemplo juntamente com o conhecimento obtido da amostra de treinamento composta por exemplos similares. A classe de cada exemplo é completamente determinada e conhecida e o número de classes é finito e conhecido.

Em conjuntos de dados reais utilizados em problemas de classificação, é comum encontrar dados com rótulos trocados. Este tipo de ruído nos conjuntos de treinamento costuma piorar a performance dos classificadores comumente utilizados em diversos problemas de classificação [Lawrence and Scholkopf, 2001], [A. Malossini, 2006]; [Yang et al., 2012]; [Yasui et al., 2004]. Mesmo assim, muitas vezes esse tipo de ruído é ignorado na prática. Algumas tentativas de contornar o problema têm sido desenvolvidas na literatura. Uma abordagem que parece simples é fazer um pré-processamento do conjunto de dados e remover ou trocar o rótulo de todas as amostras consideradas suspeitas de estarem rotuladas erradamente [Barandela and Gasca, 2000], [Brodley and Friedl, 1999], [Jiang and Zhou, 2004], [Maletic and Marcus, 2000], [Muhlenbach et al., 2004], [Sánchez et al., 2003]. Ao retirar as amostras suspeitas de estarem mal rotuladas, há a desvantagem de perder dados que podem ser importantes. Em problemas de classificação de microarranjo, por exemplo, nos quais o número de amostras geralmente é pequeno, remover algumas delas pode ser prejudicial. Para contornar esse problema, uma alternativa utilizada é tentar detectar e re-rotular os prováveis mal rotulados, "corrigindo"os rótulos

trocados. Ao conjunto de dados resultante desse pré-processamento, aplica-se algum dos algoritmos de classificação existentes [A. Malossini, 2006], [Zhang et al., 2009]. Finalmente, a terceira abordagem utilizada para contornar o problema consiste em desenvolver algoritmos robustos que gerem classificadores eficientes mesmo na presença desse tipo de ruído, isto é, algoritmos "insensíveis"ao ruído no rótulo.

Existem principalmente dois tipos de paradigmas a serem seguidos pelos métodos de classificação supervisionada: generativo e discriminativo. O generativo assume que a distribuição dos dados segue uma das distribuições de probabilidade conhecidas. De acordo com essa abordagem, deve-se estimar os parâmetros das distribuições de probabilidade condicionais, e depois disso *a posteriori* é calculada usando o teorema de Bayes. *Normal Discriminant Analysis* é um exemplo de classificador generativo. Por outro lado, o princípio discriminativo assume que precisamos apenas encontrar a regra ótima de decisão que divide os dados sem preocupação com a distribuição de probabilidade que os modela. Dessa forma a abordagem discriminativa exige suposições mais fracas a respeito da distribuição dos dados do que a generativa, sendo mais facilmente aplicada. Apesar disso, grande parte dos trabalhos publicados propondo classificadores robustos na presença de dados com ruído no rótulo utilizam abordagem generativa [Lia et al., 2007], [Bootkrajang and Kabán, 2013]. Os algoritmos propostos utilizando a abordagem discriminativa ainda são bastante limitados. Dentro desta abordagem, [Magder and Hughes, 1997] estudaram a regressão logística com probabilidade de troca de rótulos conhecida, mas reportaram problemas quando essa probabilidade é desconhecida.

Com a escassez de métodos discriminativos apropriados para lidar com esse tipo de ruído, muitas vezes são utilizados os algoritmos tradicionais de classificação (por exemplo, Regressão Logística, SVM, CART, etc), que costumam ter o desempenho significativamente afetado pelos dados mal rotulados no conjunto de treinamento. Essa baixa no desempenho costuma se agravar ainda mais quando o ruído é desbalanceado entre as classes. Como o objetivo de preencher essa lacuna, propomos um novo método de classificação robusto, utilizando a abordagem generativa, que lida bem com dados com ruído no rótulo inclusive quando o ruído ocorre de forma desbalanceada entre as classes.

O método proposto é baseado em grafos esparsos, mais especificamente no conceito de Árvore Geradora Mínima (AGM), que vem sendo utilizado com sucesso no contexto de clusterização em diversas aplicações, como processamento de imagens [Theoharatos et al., 2005], [Banerjee et al., 2014] e análise de dados biológicos [Xu et al., 2002], [Olman et al., 2009]. Novas metodologias para clusterização baseadas em AGMs estão sendo recentemente desenvolvidas [Guan-Wei Wang, 2014], apresentando bons resultados. No contexto de classificação supervisionada com foco no problema de conjuntos de dados com ruído no rótulo, essa metodologia ainda não foi utilizada. Os resultados apresentados neste trabalho são bastante promissores neste cenário.

1.1 Organização da Tese

Esta tese está organizada da seguinte forma:

- O Capítulo 2 apresenta conceitos importantes para o desenvolvimento do trabalho, já bem

estabelecidos na literatura, como os conceitos principais de aprendizagem de máquina e classificação, o problema de ruído no rótulo, conceitos básicos sobre grafos e árvores geradoras mínimas, entre outros.

- Entendendo melhor o campo de pesquisa que estamos trabalhando, no Capítulo 3 é feita a descrição da metodologia desenvolvida na tese, além de demonstrações matemáticas de sua eficiência em determinados tipos de conjuntos de dados nos quais não há problema de ruído no rótulo.

- O Capítulo 4 mostra as demonstrações matemáticas de eficiência da metodologia desenvolvida em conjuntos de dados com ruído no rótulo, de tipos específicos.

- No Capítulo 5 apresentamos aplicações do método desenvolvido em conjuntos de dados simulados. Os testes são descritos detalhadamente, assim como os conjuntos de dados. Os métodos são aplicados e comparados a outros, tradicionalmente utilizados para classificação. Resultados interessantes são obtidos e discutidos.

- Após a análise de desempenho em conjuntos de dados simulados, no Capítulo 6 apresentamos testes em conjuntos de dados reais. Novamente a metodologia desenvolvida é comparada a outras e os resultados obtidos são discutidos.

- Finalmente, no Capítulo 7 apresentamos as conclusões finais do trabalho e a avaliação do que foi realizado.

CAPÍTULO 2

Conceitos

Este capítulo tem o objetivo de introduzir alguns conceitos importantes que serão utilizados no desenvolvimento do trabalho. Após este capítulo, teremos o suporte teórico necessário para um melhor entendimento da metodologia desenvolvida assim como das análises de desempenho que serão apresentadas nos capítulos seguintes.

2.1 Grafos e Árvore Geradora Mínima

A metodologia que será apresentada neste trabalho tem como base os grafos esparsos. Mais especificamente, as árvores geradoras mínimas. Logo adiante, no Capítulo 3, veremos como é feito o uso desses conceitos no método proposto. Antes de utilizá-los, precisamos entendê-los mais detalhadamente.

Um **grafo** $G(V,E)$ consiste de um conjunto V de **vértices** (também denominados **nós**), e um conjunto E de **arestas**. Cada aresta corresponde a um par distinto de vértices, e é possível atribuir pesos (ou **custos**) às arestas. Geralmente, quanto maior é o peso de uma aresta, mais forte é relação entre os dois vértices ligados por ela.

Um **caminho** de um vértice v_1 para algum outro vértice v_k em um grafo G é uma sequência de vértices $v_1, v_2, ..., v_k$, conectados pelas arestas $(v_1, v_2), (v_2, v_3), ..., (v_{k-1}, v_k)$. Considere um grafo $G(V,E)$ no qual cada aresta de V tem um peso que corresponde a um valor real. O **custo do caminho** $p = (v_0, v_1, ..., v_k)$ é a soma dos pesos das arestas que o compõem. Um **ciclo** é um caminho no qual o primeiro e o último vértice são o mesmo. Um grafo $G(V,E)$ é **acíclico** quando ele não tem nenhum ciclo. Um grafo é chamado **conexo**, se existe um caminho entre qualquer par de vértices. Dizemos que $G(V,E)$ é **completo** quando existe uma aresta em E ligando quaisquer dois vértices de V, **denso** quando o número de arestas em E é próximo ao número máximo de arestas possível, e **esparso** quando ele tem poucas arestas (da ordem do número de vértices em V). A Figura 2.1 mostra exemplos destes três tipos de grafos.

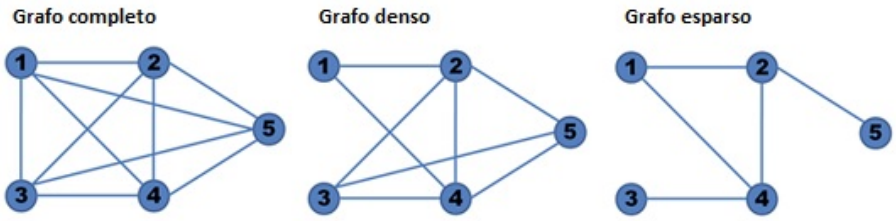


Figura 2.1 Exemplos de grafos completo, denso e esparso

Um **subgrafo** de um grafo $G(V,E)$ é um grafo $H(U,F)$ tal que $U \subseteq V$ e $F \subseteq E$. Uma **árvore** é um grafo que é acíclico e conexo $T(V,E)$. Uma **árvore geradora** de um grafo não-direcionado G é um subgrafo de G que é uma árvore e contém todos os vértices de G.

Considere um grafo não direcionado $G(V,E)$ no qual cada aresta $(u,v) \in E$, tem um custo (peso) $c(u,v)$ associado. Deseja-se encontrar um subconjunto E_T de E que forme uma árvore conectando todos os vértices de G e cuja soma total dos seus custos é minimizada. Como $T(V,E_T)$ é um grafo acíclico e conecta todos os vértices de G, forma uma árvore geradora de G. A árvore T encontrada desta forma é uma árvore geradora de custo mínimo de G [Thomas H. Cormen and Stein, 2009], conhecida como **Árvore Geradora Mínima (AGM)**.

Alguns métodos já foram propostos e implementados para construir uma AGM, como os algoritmos de Kruskal [Kruskal, 1956] e de Prim [Prim, 1957]. Neste trabalho o algoritmo utilizado é o de Prim, descrito brevemente no Algoritmo 1.

Algorithm 1 Algoritmo de Prim

1: **procedure** PRIM($G(V,E)$: grafo conexo com n vértices)
2: $T \leftarrow$ um vértice de V.
3: **for** $i \leftarrow 1$ até $n-1$ **do**
4: $e \leftarrow$ uma aresta de peso mínimo incidente em um vértice em T e que não forme um ciclo em T se for adicionada a T.
5: $T \leftarrow T$ com e adicionada.
6: **retorna** T.

2.2 Aprendizagem de Máquina e Classificação Supervisionada

Podemos dizer que a Aprendizagem de Máquina é um campo de interseção entre a Estatística e a Ciência da Computação. A idéia principal é construir algoritmos computacionais que possam acumular experiência e "aprender"a partir de dados. Dessa forma, esses algoritmos são treinados para desempenhar tarefas de previsão ou decisão.

É um campo de estudo metodológico que é atualmente aplicado nas mais diversas áreas de conhecimento. [Hastie et al., 2009] citam alguns exemplos:

- Prever quando um paciente, hospitalizado em função de um ataque cardíaco, terá um segundo ataque cardíaco. A previsão se baseia em medidas demográficas, clínicas e de dieta relacionadas ao paciente.
- Prever o preço de um produto daqui a 6 meses, baseado em medidas de performance da empresa fabricante e em dados econômicos.
- Identificar os números em um número de CEP escrito a mão, a partir da imagem digitalizada.
- Estimar a quantidade de glicose no sangue de uma pessoa diabética, a partir do espectro de absorção de infravermelho do sangue desta pessoa.

- Identificar os fatores de risco para câncer de próstata, baseado em medidas clínicas e demográficas.

Um cenário tradicional de Aprendizagem Supervisionada [Hastie et al., 2009] é constituído por uma variável resposta, normalmente contínua ou categórica, que queremos prever com base em um conjunto de características chamados atributos(ou variáveis explicativas). Temos um conjunto de dados de treinamento, no qual observamos a resposta e os atributos de suas instâncias (indivíduos do grupo, por exemplo). Utilizando este conjunto de dados de treinamento, construimos um modelo de previsão que nos permite prever a resposta para novas instâncias, para as quais apenas os valores dos atributos são conhecidos. Um bom modelo de previsão deve prever com grande acurácia a resposta de novas instâncias.

A Aprendizagem Supervisionada é caracterizada pela presença da variável resposta que irá "guiar"o processo de aprendizagem. Em um problema de Aprendizagem Não-Supervisionada, por outro lado, só há informações dos atributos e não da variável resposta. Nesse caso, o objetivo é tentar descrever como os dados são organizados ou agrupados.

A Aprendizagem Supervisionada, foco deste trabalho, se divide em 2 principais tipos. Se a resposta é contínua, temos um problema denominado Regressão. Caso ela seja categórica, temos o denominado problema de "Classificação", que será o foco deste trabalho. Mais especificamente, abordaremos problemas de classificação binária, onde a resposta só tem 2 possíveis valores (duas classes).

Os conceitos referentes à geração de um classificador a partir do aprendizado supervisionado são representados de forma simplificada na Figura 2.2. Nela, temos um conjunto com n dados, no qual cada dado x_i possui m atributos, ou seja, $x_i = (x_{i1}; ...; x_{im})$ e um rótulo y_i representado a classe. A partir dos exemplos e as suas respectivas classes, o algoritmo produz um classificador.

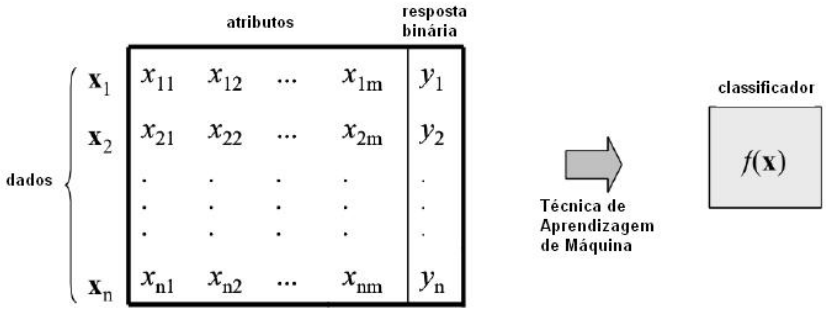


Figura 2.2 Processo de Aprendizagem Supervisionada: Produzindo um Classificador

Para estimar as taxas de predições corretas (taxa de acerto ou acurácia) ou incorretas (taxa de erro) obtidas por um classificador sobre novos dados, o conjunto de exemplos é, em geral, dividido em dois subconjuntos disjuntos: de treinamento e de teste. O subconjunto de treinamento é utilizado no aprendizado do conceito e o subconjunto de teste é utilizado para medir a qualidade do classificador obtido na predição da classe de novos dados.

Existem vários possíveis classificadores que podem ser produzidos a partir de um conjunto de dados de treinamento composto por n itens $(x_i; y_i)$. Vamos considerar, por exemplo, o con-

junto de treinamento da Figura 2.3, no qual cada elemento tem 2 atributos ($x_i = (x_{i1}, x_{i2})$ e uma resposta binária ($y_i = 0$, representado pelos círculos, ou $y_i = 1$, representado pelos triângulos) ([Scholkopf and Smola, 2002]). O objetivo do processo de aprendizado é encontrar um classificador que separe os dados das classes 0 e 1. As funções ou hipóteses consideradas são ilustradas na figura por meio das bordas, também denominadas fronteiras de decisão, traçadas entre as classes (formando as chamadas regiões de classificação).

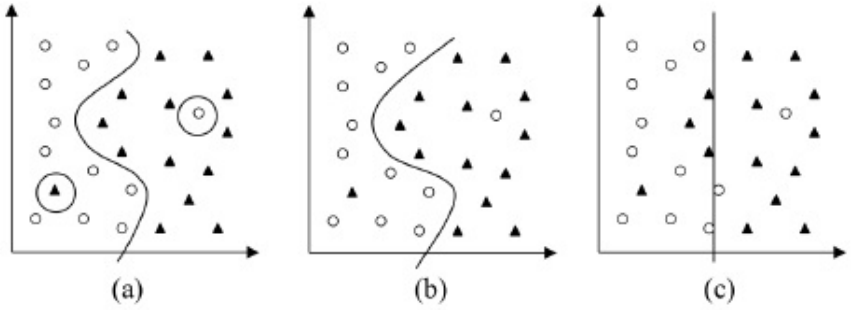


Figura 2.3 Exemplos de três diferentes classicadores gerados a partir do conjunto de dados com rótulos binários

Na Figura da esquerda, as regiões de classificação obtidas baseiam-se na classificação correta de todos os exemplos do conjunto de treinamento, incluindo dois possíveis pontos mal rotulados (rótulo errado). Podemos observar que este classificador é muito específico para o conjunto de treinamento utilizado, portanto pode ser muito suscetível a cometer erros quando for classificar novos dados, diferentes destes que foram utilizados para treinar o modelo. Esse caso representa a ocorrência de um superajustamento do modelo aos dados de treinamento. Na Figura da direita, temos um caso oposto, de sub-ajustamento do modelo, que ocorre quando o classificador gerado não é capaz de se ajustar nem mesmo aos exemplos do conjunto de dados de treinamento. Este tipo de classificador também comete muitos erros, até mesmo para casos considerados simples. Na Figura do meio, o classificador classifica corretamente grande parte dos dados, sem se fixar demais em nenhum ponto individualmente. Este classificador tem uma complexidade intermediária entre os outros dois, e representa o mais adequado dos classificadores apresentados na Figura 2.3. Em geral, um bom classificador a ser obtido a partir de um conjunto de dados de treinamento deve levar em conta seu desempenho no próprio conjunto de treinamento e sua complexidade.

Classificação supervisionada é um dos problemas mais estudados na área de aprendizagem de máquina. Atualmente existem excelentes métodos disponíveis, que vão desde os mais simples, como a regressão logística, até os mais sofisticados, como as Florestas Aleatórias (*Random Forests*) e as Máquinas de Suporte de Vetores (*Support Vector Machines*, conhecidas como SVM). Neste trabalho foram utilizados alguns destes métodos como critério de comparação com a metodologia desenvolvida. Eles serão brevemente descritos a seguir.

2.2.1 Regressão Logística

Através da Regressão Logística [Hosmer and Lemeshow, 1989], [Ferreira et al., 2001] é possível estabelecer a relação entre uma variável resposta dicotômica, normalmente representada pelos termos sucesso e fracasso, e variáveis explicativas categóricas ou contínuas. Matematicamente, o modelo logístico é apresentado a partir da seguinte expressão:

$$\log \frac{P[Y=1|x]}{P[Y=0|x]} = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p,$$

onde $P[Y=1|x]$ é a probabilidade de "sucesso", $P[Y=0|x]$, a probabilidade de "fracasso", x_i é a i-ésima componente do vetor de variáveis explicativas x e $\beta_1, \beta_2, ..., \beta_p$ são seus respectivos coeficientes no modelo.

A idéia básica do modelo logístico consiste em estabelecer uma relação linear entre as variáveis explicativas (ou alguma transformação delas, comumente a função logit) e a variável resposta. O ajuste do modelo de regressão logística, dada uma amostra de observações independentes, constituídas pelos pares (y_j, x_j), consiste em estimar os valores dos parâmetros $\beta_1, \beta_2, ..., \beta_p$, a partir do método da máxima verossimilhança. Em síntese, esse método retorna, para um dado conjunto de observações, estimativas para os parâmetros desconhecidos, de forma a maximizar a probabilidade de que os dados tenham sido originados da população correspondente.

A partir das estimativas dos valores dos parâmetros obtidos com os dados de treinamento, temos um modelo pronto para classificar novos dados cujo rótulo (a variável resposta y_i) seja desconhecida. Observe que serão obtidas as probabilidades de sucesso e fracasso, ou seja, $P[Y=1|x]$ e $P[Y=0|x]$. Neste caso, consideramos o rótulo apropriado o que tiver maior probabilidade de ser o verdadeiro, baseado nos valores das variáveis explicativas (nos atributos x_i).

2.2.2 Árvores de Regressão e Classificação (CART)

O método Árvores de Classificação e Regressão [Bell, 1996], [Ferreira et al., 2001], conhecido como CART (*Classification And Regression Tree*), é um modelo de regressão não paramétrico,que têm por objetivo estabelecer uma relação entre um vetor de variáveis preditoras x_i e uma única variável resposta y_i. Este modelo é ajustado mediante sucessivas divisões binárias no conjunto de dados, de modo a tornar os subconjuntos resultantes cada vez mais homogêneos, em relação à variável resposta. Essas divisões são convenientemente representadas por uma estrutura de árvore binária, na qual cada nó corresponde a uma divisão em uma covariável particular.

Em uma CART, tanto os atributos (variáveis explicativas) quanto o rótulo (variável resposta) podem assumir valores contínuos ou (categóricos). Se a variável resposta for numérica, o modelo recebe o nome de árvore de regressão; caso contrário, é tratada como uma árvore de classificação. Neste trabalho, como o tipo de resposta que estamos tratando é categórica, utilizamos as árvores de classificação.

O método CART consiste em sucessivas divisões do conjunto de dados, baseado nas regras de divisão obtidas em função dos valores dos atributos. As regras de divisão são representadas

por expressões do tipo "$idade < 14.5$", caso a covariável considerada para a divisão seja numérica, ou do tipo "$x_i \in A, B$", caso a covariável seja categórica. Para covariáveis categóricas, existem $2^{k-1} - 1$ possíveis divisões, onde k corresponde ao número de categorias possíveis para a variável.

Geralmente, as implementações computacionais consideram que as regras de divisão são baseadas em apenas uma das covariáveis de cada vez. Isto significa que combinações lineares entre elas não são permitidas. Esta heurística se justifica devido ao fato de que, caso combinações fossem permitidas, haveria um número explosivo de possibilidades, tornando o algoritmo tão lento, a ponto de tornar-se sem utilidade prática. Dessa forma, o CART costuma obter resultados muito bons quando as divisões entre as classes ocorrem de forma perpendicular aos eixos. Caso contrário, o método pode se tornar inadequado por não levar em conta a topologia dos dados.

2.2.3 Florestas Aleatórias

O método Florestas Aleatórias [Breiman, 2001] constrói diversas árvores de classificação (ou de regressão, quando for o caso) como as citadas na seção anterior. Para classificar um novo objeto a partir do seu vetor de atributos, ele observa a classificação deste objeto baseada em cada uma das árvores, considerando como um "voto"para a classe na qual ele foi classificado. A classificação final do objeto com base no método Florestas Aleatórias é a que obteve maior quantidade de votos em todas as árvores CART que compunham a floresta.

Cada uma das árvores CART que compõem a floresta é construída da seguinte forma:

- Se o número de elemento no conjunto de treinamento é n, então escolha n elementos aleatoriamente, com reposição, dos dados originais. Essa amostra será a amostra de treinamento de uma árvore.

- Se o número de atributos (variáveis explicativas) é M, escolha aleatoriamente $m << M$ atributos e particione o conjunto de dados em relação a um desses m atributos que melhor divide os dados do conjunto. O valor m é o mesmo para todas as partições realizadas na formação da árvore.

- Cada árvore irá "crescer"o quanto for possível, baseado nas partições em cada atributo.

O método de Florestas Aleatórias costuma obter melhores resultados que o CART e, apesar de ser computacionalmente mais complexo, tem um tempo de execução viável para tratar conjuntos de dados rezoavelmente grandes, tanto em relação ao número de atributos quanto em relação o número de instâncias (elementos) no conjunto de treinamento.

2.2.4 Maquinas de Suporte de Vetores (SVM)

A proposta do SVM [Cortes and Vapnik, 1995] é fazer a divisão do espaço dos atributos em hiperplanos ou superfícies de decisão, separando as amostras do treinamento em positivas e negativas (ou 0 e 1). A superfície que maximiza a margem de separação entre as classes (chamada de hiperplano ideal) é selecionada. Os pontos que estão mais próximos ao hiperplano ideal são

chamados vetores de suporte e são elementos importantes para a obteção do classificador na fase de treinamento. Uma qualidade do SVM é a possibilidade de adaptação para conjuntos não lineares através da utilização das funções de kernel. Em geral, os dados a serem analisados estão em um espaço de dimensão finita e é comum que eles eles não sejam linearmente separáveis neste espaço. Dessa forma, o SVM mapeia o espaço original em um espaço de dimensão mais alta, de forma que a separação das classes seja mais fácil neste novo espaço. Esse mapeamento é feito através das funções de kernel, que podem ser lineares, polinomiais, gaussianos, entre outras. Neste trabalho utilizaremos o SVM com a função de Kernel Gaussiano.

O SVM normalmente precisa ter 2 parâmetros estabelecidos:

- O parâmetro *C*, chamado de parâmetro de regularização. O parâmetro C faz o equilíbrio entre a classificação errada de um ponto do conjunto de treinamento e a simplicidade da função de classificação. De forma bastante superficial, podemos dizer que se o valor de C é mais alto, maior a importância dada a cada ponto, de forma que todos (ou aproximadamente todos) os pontos do conjunto de treinamento serão classificados corretamente e as margens do hiperplano geradas serão menores. Se o valor de C for baixo, a separação tende a ser mais suave, pois poderá não dar muito peso para alguns pontos do conjunto de treinamento. Escolhas ruins para o valor do parâmetro C podem ser responsáveis por casos de superajustamento e subajustamento.

- o parâmetro *gamma* é exigido para todos os tipos de kernel utilizados. Intuitivamente, esse parâmetro define o tamanho da influência de um único ponto do conjunto de dados de treinamento. Valores baixos de *gamma* implicam em alta influência e valores altos em baixa influência.

Não é fácil estabelecer valores para os parâmetros do SVM. Em geral, os algoritmos implementados colocam valores padrão para eles, mas estes podem passar longe dos ideais para alguns conjuntos de dados. Neste trabalho, testamos algumas combinações de possíveis valores dos dois parâmetros, baseados nos conjuntos de dados de treinamento, para escolher os mais adequados.

2.2.5 *k* Vizinhos Mais Próximos (kNN)

O método dos *k* Vizinhos Mais Próximos [Altman, 1992], [Dasarathy, 1991], conhecido como kNN (*k nearest neighbors*) tem sido bastante utilizado na solução de problemas de classificação desde o início das pesquisas nessa área e, apesar de simples, tem se mostrado um método eficaz. Para classificar um objeto ainda não classificado (objeto do conjunto de dados de teste), esse método opera da seguinte forma:

- A similaridade entre o objeto do conjunto de teste e cada uma das instâncias do conjunto de treinamento, cuja classe (rótulo) é previamente conhecida, é calculada utilizando alguma medida de similaridade os objetos. No caso deste trabalho, a medida de similaridade utilizada é a distância euclidiana entre os vetores de atributos dos objetos.

- As *k* instâncias do conjunto de treinamento mais similares ao objeto a ser classificado são selecionadas (k vizinhos mais próximos).

- O objeto é classificado em determinada categoria de acordo com algum critério de agrupamento dos k vizinhos mais próximos selecionados na etapa anterior (por exemplo, a categoria que possuir a maioria dos k vizinhos mais próximos ao objeto a ser classificado).

O parâmetro k indica o número de vizinhos que serão usados pelo algoritmo para classificar o novo objeto. Este parâmetro faz com que o algoritmo consiga uma classificação mais ou menos refinada, porém o valor ótimo de k varia de um problema para o outro. Dessa forma, o ideal é que sejam testados vários valores diferentes de forma a descobrir qual o melhor valor de k para determinado problema, baseado nos dados do conjunto de treinamento utilizado.

2.3 Medidas de Avaliação de Classificação

A avaliação dos classificadores obtidos a partir dos métodos de aprendizagem de máquina é de extrema importância. Como não há um método que costuma ter os melhores resultados para todos os tipos de conjuntos de dados, é preciso verificar qual o método que produz o classificador mais adequado aos dados que estão sendo analisados.

Para avaliar um método de classificação através do classificador obtido após a etapa de treinamento, é preciso ter disponível um conjunto de dados de teste. Este conjunto deve conter dados do tipo (x_i, y_i), ou seja, com rótulos (ou classes) conhecidos, que sejam diferentes dos dados que compunham o conjunto de treinamento. Omitindo os rótulos dessas instâncias do conjunto de teste, o classificador a ser avaliado é utilizado para encontrar os rótulos. Baseado nos valores dos rótulos encontrados, podemos comparar com os reais e verificar os acertos (ou erros) cometidos.

Para que a avaliação dos métodos seja mais justa e imparcial, na maioria das vezes é utilizada a **validação cruzada**. O conceito central das técnicas de validação cruzada é o particionamento do conjunto de dados em subconjuntos mutualmente exclusivos, e posteriormente, a utilização de alguns destes subconjuntos como conjunto de dados de treinamento e o restante dos subconjuntos como dados de validação ou de teste. Diversas formas de realizar o particionamento dos dados podem ser utilizadas, sendo as três mais comuns o método *holdout*, o *k-fold* e o *leave-one-out*.

O método *Holdout* consiste em dividir o conjunto de dados em dois subconjuntos mutuamente exclusivos, um para treinamento e outro para teste (validação). O conjunto de dados pode ser separado em quantidades iguais ou não. Uma proporção muito comum é considerar 2/3 dos dados para treinamento e o 1/3 restante para teste. Após o particionamento, o classificador é obtido com base nos dados de treinamento e, posteriormente, os dados de teste são aplicados e o erro de predição calculado. Esta abordagem é indicada quando está disponível uma grande quantidade de dados. Caso o conjunto total de dados seja pequeno, o erro calculado na predição pode sofrer muita variação.

O método *k-fold* consiste em dividir o conjunto total de dados em k subconjuntos mutuamente exclusivos do mesmo tamanho e, a partir disto, um subconjunto é utilizado como conjunto de teste e os $k-1$ restantes são utilizados como conjunto de treinamento. Este processo é realizado k vezes alternando o subconjunto de teste, de forma que em cada vez um conjunto

diferente de dados seja utilizado para teste. Para cada iteração, observa-se os acertos e os erros da classificação com base nos conjuntos de treinamento e de teste utilizados. Ao final tem-se o resultado total de todas as iterações, obtendo assim uma medida mais confiável sobre a eficácia do modelo naquele tipo de dados.

O método *leave-one-out* é um caso específico do *k-fold*, com k igual ao número total de dados n. Nesta abordagem são realizados n iterações, uma para cada dado como conjunto de teste. Apesar de apresentar uma investigação completa sobre a variação do modelo em relação aos dados utilizados, este método possui um alto custo computacional, sendo indicado para situações onde poucos dados estão disponíveis.

Para analisar o desempenho dos métodos de classificação em conjuntos de dados reais (Capítulo 6), optamos pelo método *k-fold*. A forma como ele foi utilizado será melhor explicada na descrição dos testes.

Definidos os conjuntos de treinamento e de teste, a medida mais utilizada para avaliar os métodos de classificação é a **acurácia**. Ela representa o percentual de acertos de classificação e, supondo que o conjunto de teste seja composto por n_t elementos, pode ser descrita pela fórmula:

$$Acuracia = \frac{acertos}{n_t},$$

onde *acertos* representa o número de instâncias entre as n_t do conjunto de teste que foram classificadas corretamente. É claro que o ideal é que o valor da acurácia seja o maior possível, significando que um percentual alto dos elementos foi classificado corretamente.

Mesmo a acurácia sendo uma medida bastante adequada, em geral, para avaliar os classificadores, para problemas altamente desbalanceados (ou seja, a quantidade de elementos de uma das classes é bem maior que de outra), a acurácia pode não fornecer informação adequada sobre a capacidade de discriminação de um classificador em relação a um dado grupo específico (de interesse). Considere, por exemplo, um conjunto de dados em que a classe minoritária é representada por apenas 2% das observações. Um classificador com acurácia de 98% pode ser diretamente obtido, simplesmente classificando todo exemplo como pertencente à classe majoritária. Apesar da elevada taxa de acurácia obtida, tal classificador torna-se inútil se o objetivo principal é a identificação de exemplos raros. Dessa forma, caso o conjunto de dados a ser analisado seja proveniente de um problema desse tipo, uma maneira mais eficaz de avaliar o classificador é através da distinção dos erros (ou acertos) cometidos para cada classe. Isso pode ser obtido a partir dos seguintes valores:

- Verdadeiros negativos (TN): Número de elementos nos quais o rótulo original é negativo e o rótulo dado pelo classificador é negativo.
- Verdadeiros positivos (TP): Número de elementos nos quais o rótulo original é positivo e o rótulo dado pelo classificador é positivo.
- Falsos positivos (FP): Número de elementos nos quais o rótulo original é negativo e o rótulo dado pelo classificador é positivo.
- Falsos negativos (FN): Número de elementos nos quais o rótulo original é positivo e o rótulo dado pelo classificador é negativo.

Baseados nestes valores, algumas medidas são comumente utilizadas para avaliação dos métodos de classificação:

- Sensibilidade (Revocação - *Recall*): É o percentual de elementos corretamente classificados como positivos dentro do total de elementos com rótulos positivos originais: $\frac{TP}{TP+FN}$.
- Especificidade: É o percentual de elementos corretamente classificados como negativos dentro do total de elementos com rótulos negativos originais: $\frac{TN}{TN+FP}$.
- Precisão: É o percentual de elementos corretamente classificados como positivos dentro do total de elementos classificados como positivos: $\frac{TP}{TP+FP}$.

Com todas essas ferramentas em mãos, basta explorar o conjunto de dados que será analisado para escolher as medidas mais adequadas para avaliar a eficiência dos classificadores.

2.4 Dados com Ruído no Rótulo

Conforme visto na descrição de um problema de classificação de dados, o conjunto de treinamento dos algoritmos (composto por instâncias com os valores dos atributos e do rótulo conhecidos) é fundamental para a obtenção de um bom classificador, independente do método escolhido. É fácil perceber que os dados deste conjunto de treinamento estarem rotulados de forma correta faz com que este conjunto seja uma base de melhor qualidade para a obtenção dos classificadores, afinal isso proporciona uma avaliação mais correta da relação entre os atributos e os rótulos, que é o objetivo do algoritmo de classificação. Porém, em grande parte dos casos a serem analisados, nos deparamos com problemas de dados que foram rotulados de forma errada. Infelizmente, muitas vezes não há garantia de que os rótulos dados aos elementos são realmente corretos. Hoje em dia, como o tamanho dos conjunto de dados e o grau de complexidade são cada vez maiores, torna-se quase impossível obter um conjunto de dados cuja atribuição de rótulos é perfeita. Estes erros na atribuição dos rótulos podem ser originados por diferentes motivos, incluindo a natureza subjetiva da tarefa de rotulagem, o efeito de ruído na comunicação e a falta de informação para determinar o rótulo verdadeiro de um exemplo.

Segundo a literatura, podemos diferenciar dois tipos de ruído: ruído nos atributos e ruído no rótulo. Em [Zhu and Wu, 2004], podemos observar que o ruído no rótulo costuma ser mais prejudicial à classificação do que o ruído nos atributos, o que mostra a importância de estudar esse tipo de problema e buscar algoritmos robustos em relação a ele. Essa maior importância do ruído no rótulo pode ser explicada devido ao fato de haverem vários atributos e apenas um rótulo em cada instância do conjunto de dados, ao mesmo tempo em que a importância de cada atributo para o processo de treinamento do algoritmo é diferente, enquanto os rótulos sempre têm um grande impacto nesse processo.

Para entendermos melhor o problema do ruído no rótulo dos elementos que compoem um conjunto de dados de treinamento e como ele afeta os algoritmos de classificação, vamos definir uma taxonomia para este tipo de ruído, com base em [Frénay and Verseysen, 2014]. Considere as seguintes variáveis aleatórias: X é o vetor de atributos, $\tilde{Y}$ é a classe real da observação, Y é

o rótulo observado e E é uma variável binária que indica quando uma troca de rótulo ocorreu $(Y \neq \tilde{Y})$. O ruído no rótulo pode ser classificado segundo 3 possíveis modelos estatísticos:

2.4.1 Modelo de Ruído Completamente Aleatório (*Noise Completely at Random Model (NCAR))*

A ocorrência de um erro (um rótulo observado trocado) é independente das outras variáveis aleatórias, inclusive da classe verdadeira da observação. No caso NCAR, o rótulo observado é diferente do rótulo real com probabilidade $p_e = P(E = 1) = P(Y \neq \tilde{Y})$. No caso da classificação binária, este tipo de ruído é necessariamente simétrico, ou seja, o percentual de troca de rótulos nas duas classes é o mesmo. No caso de classificação com mais de duas classes, quando $E = 1$ costuma-se assumir que o rótulo incorreto é escolhido aleatoriamente entre as demais classes de rótulos possíveis. Este modelo é chamado de "ruído uniforme no rótulo".

2.4.2 Modelo de Ruído Aleatório (*Noise at Random Model (NAR))*

A probabilidade de erro depende da classe verdadeira $\tilde{Y}$. Nesse modelo, E ainda é independente de X, mas é possível modelar ruídos assimétricos nos rótulos, que ocorrem quando observações de determinada(s) classe(s) tendem a ter mais troca de rótulos do que outras. Pode-se definir as probabilidades de observar cada rótulo como:

$$P(Y = y|\tilde{Y} = \tilde{y}) = \sum_{e \in \{0,1\}} P(Y = y|E = e, \tilde{Y} = \tilde{y}) P(E = e|\tilde{Y} = \tilde{y})$$

Observe que o NCAR é um caso específico do NAR. Por exemplo, para o modelo de ruído uniforme no rótulo, se o número de classes é dado por n_Y, temos que:

$$P(Y = y|\tilde{Y} = \tilde{y}) = \begin{cases} 1 - p_e, & \text{se} \quad y = \tilde{y} \\ \frac{p_e}{n_Y - 1}, & \text{se} \quad y \neq \tilde{y} \end{cases}$$

2.4.3 Modelo de Ruído Não Aleatório (*Noise Not at Random Model (NNAR))*

Este é o modelo mais complexo e também mais realista para ruído no rótulo. A variável E depende das variáveis aleatórias $\tilde{Y}$ e X, o que permite que as trocas de rótulos sejam mais prováveis para determinadas classes e em certas regiões do espaço dos atributos X. Por exemplo, trocas de rótulos mais prováveis perto das fronteiras das regiões de classificação ou em regiões de baixa densidade podem ser modeladas apenas pelo NNAR.

A confiança nos rótulos é mais complexa de estimar do que para o NCAR e o NAR, afinal a probabilidade de erro também depende do valor de X. Em certos casos, a densidade de trocas de rótulos pode apresentar picos importantes em determinadas regiões. Nesse caso, o mais adequado é caracterizar a confiança nos rótulos observados a partir da quantidade dada por: $p_e(x, (\tilde{y})) = P(E = 1|X = x, \tilde{Y} = \tilde{y})$.

Definidos os três tipos de ruído no rótulo que podem ocorrer nos conjuntos de dados, já podemos ter uma noção da forma com que eles podem afetar os métodos de classificação. A principal consequência da utilização de dados com ruído no rótulo para treinar um algoritmo

é a diminuição da performance na classificação. Alguns trabalhos publicados estudaram as consequências deste tipo de ruído nos resultados de métodos de classificação utilizados comumente.

[Bi and Jeske, 2010] mostraram que o ruído no rótulo afeta o método discriminante normal e a regressão logística: as taxas de erro de classificação aumentam e os parâmetros se tornam viciados. A regressão logística parece ser menos afetada. A performance de classificação do kNN também é afetada pelo ruído no rótulo [Wilson and Martinez, 2000], em particular quando $k = 1$ [Okamoto and Nobuhiro, 1997]. Para pequenos conjuntos de dados de treinamento, sem ruído no rótulo, o classificador $1NN$ costuma ser ótimo. Porém, na presença de ruído no rótulo, o número ótimo de vizinhos k cresce de acordo com o número de instâncias no conjunto de treinamento e com a quantidade dados mal rotulados. Alguns estudos comparam a performance de classificadores na presença de ruído no rótulo. Em [Nettleton et al., 2010], os resultados da classificação utilizando o SVM para este tipo de dado se mostram bem fracos, o que é atribuido à sua dependência dos vetores de suporte e à suposição de dependência entre as variáveis.

Além dos trabalhos citados, há diversos outros estudos comprovando que dados do conjunto de treinamento com ruído no rótulo afetam consideravelmente os métodos de classificação supervisionada. Dessa forma, é de extrema importância que sejam desenvolvidos novos métodos robustos a esse tipo de ruído, ou que façam um pré-processamento do conjunto de dados, de forma a "corrigí-lo"para proporcionar bons resultados de classificação.

CAPÍTULO 3

Metodologia

3.1 Definição do Método

Nesta seção apresentamos a descrição da metodologia de classificação robusta para dados com ruído no rótulo baseada em grafos esparsos, mais especificamente em Árvores Geradoras Mínimas - AGMs - (***l**abel n**o**ise **r**obust **c**lassification method*), que chamaremos de agora em diante de **LORC**. LORC é um método simples para classificação de dados, não-paramétrico, que gera bons classificadores em conjuntos de dados com formatos diversos (sem grandes restrições de formatos, como ocorre com o CART e com a Regressão Logística, por exemplo) e que é capaz de lidar bem com conjuntos de dados de treinamento com dados mal rotulados.

Considere um conjunto de dados de treinamento $V = \{(x_1, y_1), ..., (x_D, y_D)\}$, onde $x_i \in R^M$ e $y_i \in \{0, 1\}$, de forma que x_i é um vetor que representa o(s) atributo(s) do objeto e y_i representa o rótulo. Baseado na idéia do SKATER [Assunção et al., 2006], que é um algoritmo proposto para clusterização de dados espaciais, nosso algoritmo é composto por três etapas: a primeira, na qual é construída uma AGM a partir do conjunto V; a segunda, na qual são realizadas as podas na AGM, formando as regiões de classificação a serem utilizadas; e a terceira, que consiste na classificação dos novos pontos cujo rótulo é desconhecido. Observe que as duas primeiras etapas consistem na parte de aprendizagem do algoritmo a partir de exemplos anteriores, ou seja, de exemplos cujos rótulos são conhecidos. Após a conclusão destas, o algoritmo estará apto a classificar novas instâncias, atribuindo a elas o rótulo que considerar adequado.

Cada um dos objetos é considerado um nó no grafo conexo e não direcionado $G(V;E)$. Na primeira etapa, de construção da AGM, o custo associado à aresta que liga os vértices (v_i, v_j) é dado por uma medida de distância (no caso do algoritmo desenvolvido neste trabalho, a distância euclidiana) entre os atributos (x) dos objetos correspondentes, medindo a dissimilaridade entre eles. A partir deste grafo, é gerada uma AGM $T(V;E_T)$ utilizando o algoritmo de Prim [[Prim, 1957]]. $T(V;E_T)$ é um grafo reduzido, com custo mínimo, no qual há um caminho possível entre quaisquer 2 vértices do grafo, ao percorrer sucessivas arestas.

Após concluída a primeira etapa do processo, temos uma AGM construída a partir de $G = (V;E)$. A segunda etapa consiste em "podar"essa AGM de forma a obtér *clusters* o mais homogêneos possível entre si e o mais heterogêneos possível uns dos outros, em relação aos rótulos y_i's. Portanto, nesta etapa serão consideradas novas medidas de dissimilaridade como peso das arestas que compõem o conjunto E_T. Essas medidas serão agora baseadas apenas nos rótulos y_i's, e não mais nos atributos x_i's. Elas são medidas globais, pois levam em conta todos os vértices do grafo. Primeiramente, antes de ser feita qualquer poda, calcula-se uma medida da dissimilaridade total entre todos os objetos do grafo e a média. Por exemplo, a medida uti-

lizada neste trabalho é $SSTO = \sqrt{\sum_i (y_i - p)^2}$, onde p é a proporção de rótulos iguais a 1 no grafo. Podemos observar que SSTO mede a dissimilaridade entre os rótulo de todos os pontos do grafo e a média dos rótulos.

A poda da árvore é feita sequencialmente, partindo do conjunto inicial de todas as arestas do conjunto E_T. A cada iteração, o peso atribuído a cada aresta será referente ao "ganho"obtido ao retirar essa aresta do grafo, dividindo-o em grupos separados (sem nenhuma aresta unindo tais grupos). Neste trabalho, o peso referente a cada aresta e_i é definido da seguinte forma:

$$Q(e_i) = SSTO - SSW \tag{3.1}$$

$$SSW = \sqrt{\sum_{q=1}^{C} \sum_{i \in T_q} (y_i - p_q)^2}$$

onde p_q é a proporção de $1's$ no grupo T_q, e C é o número de grupos em que será dividido o conjunto de dados. É importante observar que SSW contém a soma das medidas de dissimilaridade em relação aos rótulos dentro de cada grupo formado. Quanto maior o valor de SSW, mais heterogêneos os grupos são (entre seus próprios elementos). Logo, o ideal é que o valor de SSW seja pequeno, de forma que os grupos formados sejam compostos por muitos elementos de mesmo rótulo. Na melhor da hipóteses, SSW pode atingir o valor 0, o que implicaria em $Q(e_i) = SSTO$ ao retirar a aresta e_i do grafo. Porém, nem sempre é possível alcançar esse valor. Nas próximas seções, este tópico será melhor discutido.

A cada iteração, considerando a medida Q referente à partição resultante da retirada de cada uma das arestas, a aresta que com maior peso (maior valor de Q) será podada (retirada do grafo). Após $C-1$ arestas podadas, C subgrupos estarão formados. Dessa forma, o grafo inicial é dividido em C subgrafos. Cada um desses grupos formados será considerado um conjunto de objetos de um determinado rótulo y. Para atribuir o rótulo de cada grupo formado, o critério utilizado foi de utilizar o rótulo mais frequente entre os objetos do grupo. Suponha que um dos grupos formados, o T_j, tem n elementos. Desses n elementos, r têm rótulo 0 e $s = n - r$ têm rótulo 1. Então, se $r \geq s$, T_j é rotulado como 0. Caso contrário, ele é rotulado como 1.

Definidos os grupos, o procedimento para classificar novos objetos se assemelha ao kNN, porém utilizamos os rótulos dos grupos definidos na fase de aprendizagem do algoritmo ao invés dos rótulos originais dos dados. Levando em consideração que objetos com atributos semelhantes tendem a ter o mesmo rótulo, um novo objeto cujo rótulo é desconhecido será classificado no grupo que contém o seu vetor de atributos. Este grupo é definido como o que contém a maioria dos k vetores de atributos mais próximos ao novo vetor. Se $\hat{y}_i$ é o rótulo do grupo ao qual o vetor x_i pertence, temos então uma regra de classificação h_S dada por:

$$h_S(x) = \begin{cases} 1, & \text{se} \quad \sum_{i=1}^{m} \hat{y}_i w_i \geq \frac{1}{2}, \\ 0, & \text{caso contrário} \end{cases}$$

onde

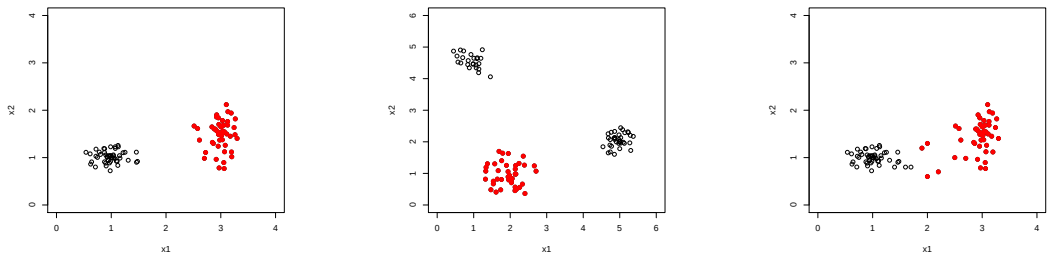
$$w_i = \begin{cases} \frac{1}{k}, & \text{se } x_i \text{ é um dos } k \text{ vetores de atributos mais próximos a } x \\ 0, & \text{caso contrário} \end{cases}$$

3.2 Demonstração da eficiência do método

Nos testes de desempenho que apresentaremos posteriormente, o método LORC se mostrou eficiente para diversos tipos de conjuntos de dados. Nesta etapa do trabalho, vamos exibir as demonstrações teóricas de eficiência do método para conjuntos de dados compostos por *cluster* rotulados compactos. Porteriormente, veremos as demontrações para conjuntos de dados com ruído no rótulo, mas neste primeira parte é importante destacar que estamos tratando de conjuntos de dados compostos por *clusters* bem definidos, onde cada *cluster* é composto por instâncias com rótulos idênticos. Então, vamos definir tais *clusters* :

Definição 1 (*Cluster* rotulados compactos)**.** Considere um conjunto de pontos rotulados V. Para uma dada métrica de distância, um *cluster* rotulado compacto C é um sub-conjunto de V, no qual todos os pontos têm o mesmo rótulo y, tal que para qualquer ponto $v_i \in C$, $dist(v_i, v_j) < dist(v_i, v_k)$, para todo ponto $v_j \in C$ e todo ponto $v_k \notin C$.

A Figura 3.1 mostra exemplos de dois conjuntos de dados compostos por *clusters* rotulados compactos, sendo que o representado em 3.1(a) é composto por 2 *clusters* e o representado em 3.1(b) por 3 *clusters*. Na Figura 3.1(c) o conjunto de dados é formado por 2 *clusters* que não atendem a Definição 1, ou seja, não são *clusters* rotulados compactos. Os pontos em vermelho têm rótulo 1 e os demais têm rótulo 0.



(a) (b) (c)

Figura 3.1 Exemplos de conjuntos de dados formados por *clusters* que atendem a Definição de rotulados compactos (em 3.1(a) e 3.1(b)) e que não a atendem (em 3.1(c)). As cores distintas representam os rótulos distintos das instâncias.

Objetivando mostrar que os *clusters* obtidos após a etapa da poda da AGM são os melhores possíveis, vamos considerar mais uma Definição:

Definição 2 (*Cluster* ótimos em relação ao rótulo e *cluster* ideais em relação ao rótulo)**.** Ao particionar um conjunto de dados rotulados em C *clusters* a partir da poda de $C-1$ arestas da AGM correspondente, os C *clusters* ótimos com relação ao rótulo são os que resultam no valor máximo possível de Q, definido em (3.1). Se $Q = SSTO$, então os *clusters* ótimos em relação ao rótulo obtidos são exatamente os representados nos dados, ou seja, a partição encontrada é a ideal. Nesse caso, diremos que além de ótimos com relação ao rótulo, eles são os *clusters* ideais em relação ao rótulo.

Os *clusters* ótimos em relação ao rótulo têm a principal característica de tentarem ser os mais homogêneos possível dentro de cada cluster, em relação ao rótulo. No caso dos *clusters* ideais em relação ao rótulo, quando alcançamos $Q = SSTO$ (o que significa que $SSW = 0$) a partição considerada do grafo gera C *clusters* sendo que cada um deles é formado por pontos com mesmo rótulo, ou seja, são todos *clusters* completamente homogêneos em relação ao rótulo.

Finalmente vamos definir mais um conceito que será utilizado na demonstração: o conceito de uma sub-árvore dominada.

Definição 3 (Sub-árvore dominada)**.** Em uma AGM $T(V, E_T)$, seja u um vértice da aresta $e \in E_T$. Suponha que tenham sido calculados os pesos das arestas de E_T, conforme a fórmula dada em 3.1. Uma sub-árvore $T_S(V_S, E_S)$ é dita dominada se:

- $u \in V_S; e \notin E_S; |E_S| > 0;$
- $\max\{Q(e_i)|e_i \in E_S\} < Q(e)$.

Feitas as definições necessárias, vamos propor um primeiro Teorema para mostrar uma característica importante das AGMs que geram um conjunto de dados qualquer formado por *clusters* rotulados compactos.

Teorema 1. *Seja um conjunto de dados rotulados V com a respectiva AGM $T(V, E_T)$. Se V é formado por n_C clusters rotulados compactos, então existem exatamente $n_C - 1$ arestas em E_T que ligam pontos com rótulos distintos. Isso significa que se existe ligação entre pontos que pertencem a clusters distintos (clusters compostos por pontos com rótulos diferentes), essa ligação é feita por uma única aresta $e \in E_T$.*

Prova. Sem perda de generalidade, suponha um conjunto de dados V formado por dois *clusters* rotulados compactos C_1 e C_2 e $v_s \in C_1$ é o vértice inicial a entrar na AGM, na execução do algoritmo de Prim. O vértice $v_1 \in V$ é o próximo a entrar na árvore logo após v_s, ainda pelo algoritmos de Prim. É claro que $v_1 \in C_1$, pois a distância entre v_s e v_i, para qualquer $v_i \in C_1$ é menor que a distância entre v_s e v_j, para qualquer $v_j \in C_2$, pela hipótese de que eles são *clusters* rotulados compactos. Utilizando o mesmo argumento é fácil perceber que, até que todo ponto $v_i \in C_1$ já esteja na árvore T, os próximos vértices a serem selecionados através do algoritmo de Prim serão pontos de C_1. Dessa forma, com os índices indicando a ordem de entrada na árvore pelo algoritmo de Prim, temos que o conjunto dos vértices $\{v_k|1 \leq k < |C_1|\} \cup v_s = C_1$ e que o conjunto das arestas $\{e_k|1 \leq k \leq |C_1|\}$ é constituído apenas de arestas cujos dois vértices pertencem a C_1.

Se $|C_1| = n_1$, então o vértice $v_{n_1} \in V$ é o primeiro vértice de C_2 a ser selecionado para entrar na AGM. Logo, e_{n_1} tem um vértice em C_1 e outro em C_2. Nas etapas $\{i|n_1 < i \leq |C_1 \cup C_2|\}$ seguintes do algoritmo de Prim, faltam os demais pontos de C_2 para entrarem na AGM. Similarmente ao que ocorreu ao selecionar os pontos de C_1 no algoritmo de Prim, o próximo vértice a entrar na árvore, $v_{n_1+1} \in C_2$ será mais próximo de v_{n_1} do que de qualquer ponto de C_1, de forma que a aresta e_{n_1+1} tem os dois vértices em C_2. Assim, sucessivamente, a AGM ficará completa, de forma que nas etapas $\{i|n_1 < i < |C_1 \cup C_2|\}$ teremos apenas arestas cujos vértices são ambos de C_2. Portanto, apenas a aresta e_{n_1} liga pontos de *clusters* distintos.

A demonstração para mais de 2 *clusters* segue de forma análoga.

A principal conclusão é que, para conjuntos de dados formados por *clusters* rotulados compactos, a ligação entre quaisquer dois *clusters* é sempre realizada por apenas uma aresta na AGM. Isso tem implicações interessantes, como por exemplo a certeza de que os subconjuntos resultantes da poda dessas $n_C - 1$ arestas da AGM resultará em n_C sub-árvores da AGM (sendo que cada uma delas representa um dos *clusters* rotulados compactos), conforme veremos mais adiante.

Agora temos em mãos as ferramentas necessárias para concluir que baseados nas medidas de dissimilaridade referentes aos valores possíveis de Q, podemos encontrar as arestas da AGM que devem ser podadas para obtermos os *clusters* ótimos em relação ao rótulo, fazendo com que a partição do conjunto de dados estabelecida seja a mais correta possível. Antes de apresentarmos o Teorema final, precisamos mostrar que a aresta com maior valor Q referente à partição resultante de sua poda é a aresta correta a ser retirada. Inicialmente, consideraremos o caso particular no qual o conjunto de dados é formado por apenas 2 *clusters* rotulados compactos. Em seguida, estenderemos as demonstrações para conjuntos de dados compostos por qualquer número de *clusters* rotulados compactos.

3.2.1 Caso Particular: 2 *clusters* rotulados compactos

No caso em que o conjunto de dados V é formado por apenas 2 *clusters*, precisamos mostrar que a aresta com maior peso (dado pelo valor de Q calculado a partir da partição resultante de sua poda) na AGM de V é a aresta de ligação entre os *clusters*. Quando tratarmos de mais de 2 *clusters* compondo o conjunto de dados, é necessário lembrar que cada *cluster* C_i (desde que $|C_i| > 1$) tem 1 ou mais vértices que são vértices de uma aresta de ligação a outro cluster. Caso o *cluster* tenha apenas 1 vértice deste tipo, o LORC nunca irá podar uma aresta que une dois pontos pertencentes a este *cluster* antes de podar a aresta de ligação cujo vértice pertence a ele. Considere, então, o Lema a seguir:

Lema 1. *Seja V um conjunto de dados composto por n_C clusters rotulados compactos e $T(V, E_T)$ a AGM correspondente. Seja C_1 um dos n_C clusters rotulados compactos de V, tal que só existe um vértice $v_a \in C_1$ que seja vértice de uma aresta $e = (v_a, v_b)$ de ligação de C_1 com outro cluster rotulado compacto C_2 de V, onde $e \in E_T$ e $v_b \in C_2$. Então, a medida de dissimilaridade Q referente à poda da aresta e é maior do que a referente a qualquer outra aresta $(v_{a'}, v_{a''})$ tal que $v_{a'}, v_{a''} \in C_1$.*

Prova. Sem perda de generalidade, vamos fazer algumas suposições:

- Suponha que o número de instâncias em cada um dos *clusters* seja representado da seguinte forma: cada *cluster* C_i é composto por n_{C_i} vértices, com $i = 1, 2, ..., n_C$.

- Suponha que a ligação entre os n_C *clusters* rotulados compactos seja feita por arestas em E_T da seguinte forma: C_1 é ligado a C_2, C_2 a C_1 e C_3, C_3 a C_2 e C_4, e assim por diante, até o último *cluster* C_{n_C}, que é ligado apenas a C_{n-1}.

- Suponha também que os *clusters* C_i's tais que i é ímpar são formados por intâncias com rótulo 0 e os C_i's com i par por instâncias com rótulo 1.

Vamos calcular o valor de Q resultante da possível poda da aresta $e = (v_a, v_b)$ (chamaremos este de Q_1) e também o resultante da poda de uma outra aresta $(v_{a'}, v_{a''})$ qualquer, tal que $v_{a'}, v_{a''} \in C_1$ (chamaremos este de Q_2). Observe que ao podar a aresta $(v_{a'}, v_{a''})$ estaremos dividindo o *cluster* C_1 em 2 sub-cluster isolados um do outro, um deles, com $n_{C_{1b}}$ instâncias, ligado a C_2 por (v_a, v_b) e outro, com $n_{C_{1a}}$ instâncias, sem nenhuma ligação aos demais *clusters* . Seja qual for o valor de n_C, o cálculo de Q_1 se dá da seguinte forma:

$$Q_1 = SSTO - \sqrt{\sum_{q=1}^{2} \sum_{i \in T_q} (y_i - p_q)^2}$$

$$\sum_{q=1}^{2} \sum_{i \in T_q} (y_i - p_q)^2 = (n_{C_1} * 0) + (n_{C_2} + n_{C_4} + ... + n_{C_{n_C}})(1 - \frac{n_{C_2} + n_{C_4} + ... + n_{C_{n_C}}}{n_{C_2} + n_{C_3} + n_{C_4} ... + n_{C_{n_C}}})^2$$

$$+ (n_{C_3} + n_{C_5} + ... + n_{C_{n_C - 1}})(0 - \frac{n_{C_2} + n_{C_4} + ... + n_{C_{n_C}}}{n_{C_2} + n_{C_3} + n_{C_4} ... + n_{C_{n_C}}})^2$$

$$= \frac{(n_{C_2} + n_{C_4} + ... + n_{C_{n_C}})(n_{C_3} + n_{C_5} + ... + n_{C_{n_C - 1}})}{n_{C_2} + n_{C_3} + n_{C_4} ... + n_{C_{n_C}}}$$

Então, $Q_1 = SSTO - \sqrt{\frac{(n_{C_2} + n_{C_4} + ... + n_{C_{n_C}})(n_{C_3} + n_{C_5} + ... + n_{C_{n_C - 1}})}{n_{C_2} + n_{C_3} + n_{C_4} ... + n_{C_{n_C}}}}$.

Para o cálculo de Q_2, ao tirarmos uma aresta qualquer que une dois pontos pertencentes ao mesmo *cluster* C_1, temos o seguinte:

$$Q_2 = SSTO - \sqrt{\sum_{q=1}^{2} \sum_{i \in T_q} (y_i - p_q)^2}$$

$$\sum_{q=1}^{2} \sum_{i \in T_q} (y_i - p_q)^2 = (n_{C_{1a}} * 0) + (n_{C_2} + n_{C_4} + ... + n_{C_{n_C}})(1 - \frac{n_{C_2} + n_{C_4} + ... + n_{C_{n_C}}}{n_{C_{1b}} + n_{C_2} + n_{C_3} ... + n_{C_{n_C}}})^2$$

$$+ (n_{C_{1b}} + n_{C_3} + n_{C_5} + ... + n_{C_{n_C - 1}})(0 - \frac{n_{C_2} + n_{C_4} + ... + n_{C_{n_C}}}{n_{C_2} + n_{C_3} + n_{C_4} ... + n_{C_{n_C}}})^2$$

$$= \frac{(n_{C_2} + n_{C_4} + ... + n_{C_{n_C}})(n_{C_{1b}} + n_{C_3} + n_{C_5} + ... + n_{C_{n_C - 1}})}{n_{C_{1b}} + n_{C_2} + n_{C_3} + n_{C_4} ... + n_{C_{n_C}}}$$

Então, $Q_2 = SSTO - \sqrt{\frac{(n_{C_2} + n_{C_4} + ... + n_{C_{n_C}})(n_{C_{1b}} + n_{C_3} + n_{C_5} + ... + n_{C_{n_C - 1}})}{n_{C_{1b}} + n_{C_2} + n_{C_3} ... + n_{C_{n_C}}}}$.

Temos então os valores de Q_1 e Q_2 e queremos verificar se $Q_1 > Q_2$, como diz o Lema. Então vamos fazer a comparação, considerando as equivalências seguintes:

$$Q_1 > Q_2$$

$$\equiv \frac{(n_{C_2} + n_{C_4} + ... + n_{C_{n_C}})(n_{C_3} + n_{C_5} + ... + n_{C_{n_C - 1}})}{n_{C_2} + n_{C_3} + n_{C_4} ... + n_{C_{n_C}}} > \frac{(n_{C_2} + n_{C_4} + ... + n_{C_{n_C}})(n_{C_{1b}} + n_{C_3} + ... + n_{C_{n_C - 1}})}{n_{C_{1b}} + n_{C_2} + n_{C_3} + n_{C_4} ... + n_{C_{n_C}}}$$

$$\equiv 0 < n_{C_{1b}}(n_{C_2} + n_{C_4} + ... + n_{C_{n_C}})$$

É claro que a última desigualdade é verdadeira, pois $n_{C_{1b}} \geq 1$ e $n_{C_2} \geq 1$, e os demais *clusters* $C_4, ...C_n$ podem ter 0 ou mais elementos. Portanto, a partir das equivalências fica demonstrado o Lema.

É fácil observar que no caso que estamos tratando nesta seção, quando o número de *clusters* n_C é igual a 2, todos os *clusters* do conjunto de dados se encaixam nas suposições do Lema 1, de forma que a aresta a ser retirada na primeira poda da AGM será a aresta de ligação entre os 2 *clusters* C_1 e C_2, particionando o conjunto V da forma ideal. O caso em que $n_C > 2$ será discutido mais atenciosamente na seção 3.2.2.

Agora sim, podemos concluir com o Teorema 2:

Teorema 2. *Seja V um conjunto de dados composto por n_C clusters rotulados compactos e $T(V,E_T)$ a AGM correspondente. Se existe uma aresta $e \in E_T$ com vértices u e v e com peso $Q(e)$ (valor de Q referente ao grafo resultante da poda de e), tal que (u,v) foi a $i_{u,v}$-ésima aresta a ser agragada a E_T durante a execução do algoritmo de Prim e tal que é satisfaz:*

$$\begin{cases} Q(e) > max\{Q(e_j)|1 \leq j < i_{u,v}\} \\ Q(e) > max\{Q(e_j)|i_{u,v} < j \leq |E_T|\} \\ indice(u) < indice(v) \end{cases} \tag{3.2}$$

então existem duas sub-árvores $T_1(V_1,E_1)$ e $T_2(V_2,E_2)$ que são dominadas por u e v separadamente e satisfazem:

$$u \in V_1; v \in V_2; |E_1| > 0; |E_2| > 0.$$

Além disso, os pontos de V_1 e V_2 representam os clusters ótimos em relação ao rótulo que podem ser obtidos ao retirar uma aresta de V.

Prova. Suponha que, durante a execução do algoritmo de Prim, v_S foi o vértice inicial (primeiro a entrar na AGM) e (u,v) foi o $i_{u,v}$-ésimo vértice a ser agregado à AGM. Dadas as condiçoes do Teorema, temos que $Q(e) > max\{Q(e_j)|0 < j < i_{u,v}\}$. Considere o conjunto $V_1 = (\{v_k|0 < k < i_{u,v}\} \cup \{v_s\})$. Suponha, por contradição, que V_1 não é uma árvore. Então, existe pelo menos uma aresta $\{e_k|0 < k < i_{u,v}\}$ em E_T com um vértice que não pertence a V_1. Mas pelo Teorema 1, só existe uma aresta em E_T com um vértice em cada cluster, e essa aresta é a e. Portanto, $V_1 = T_1(V_1,E_1)$ é a sub-árvore de $T(V,E_T)$ que satisfaz $u \in V_1$ e $|E_1| > 0$ e é dominada por u.

De forma análoga, é fácil provar que o subconjunto $T_2(V_2,E_2)$, com $V_2 = \{v_k|i_{u,v} < k < |V|\}$ e $E_2 = \{e_k|i_{u,v} \leq k < |V|\}$ é a sub-árvore de $T(V,E_T)$ que satisfaz $v \in V_2$ e $|E_2| > 0$ e é dominada por v.

Como $Q(e) > max\{Q(e_j)|1 \leq j < i_{u,v}\}$ e $Q(e) > max\{Q(e_j)|i_{u,v} < j \leq |E_T|\}$, fica claro que tirando a aresta e, os *clusters* obtidos são os *clusters* ótimos com relação ao rótulo, segundo a Definição 2.

Caso o número de *clusters* n_C seja igual a 2, pelo Lema 1 podemos concluir que a aresta e é a aresta que faz a ligação entre estes dois *clusters* , já que $Q(e) > max\{Q(e_j)|1 \leq j < i_{u,v}\}$ e $Q(e) > max\{Q(e_j)|i_{u,v} < j \leq |E_T|\}$. Como eles são dois *clusters* rotulados compactos (não há pontos com rótulos trocados em nenhum deles), $SSW = 0$ e, consequentemente, $Q = SSTO$. Portanto, os *clusters* formados são ideais em relação ao rótulo.

A demonstração mostra que, além do resultado geral do Teorema 2, se $n_C = 2$, então os *clusters* obtidos são também os ideais em relação ao rótulo.

Dessa forma, temos definido o método que, baseado apenas nos pesos relativos aos rótulos definidos a partir do cálculo de Q e na AGM construída com base nas distâncias entre os atributos x, é capaz de estabelecer uma partição do espaço que contém os dados que irá determinar as regiões de classificação a serem utilizadas no proximo passo do método, para classificar novos objetos cujo rótulo é desconhecido.

Para um conjunto de dados formado por 2 *clusters* , a prova está completa. Quando o número de *clusters* é maior que 2, a discussão se estende na seção 3.2.2.

3.2.2 Caso Geral: n_C *clusters* rotulados compactos

Podemos perceber que a demonstração apresentada na seção anterior funciona perfeitamente no caso de termos apenas 2 *clusters* rotulados compactos compondo o conjunto de dados V. Caso o número de *clusters* seja maior, precisamos definir algumas condições extras para que os resultados sejam válidos. Como a medida de dissimilaridade Q é uma medida global, ela utiliza todos os vértices de V no cálculo do peso de cada aresta, e não apenas os dois vértices desta aresta. Quando temos 3 ou mais *clusters* formando V, a medida de dissimilaridade será calculada para a retirada de uma aresta de cada vez, com base na homogeneidade dos pontos dentro dos *clusters* formados com esta poda e na heterogeneidade entre eles. Na primeira poda, por exemplo, serão formados dois *clusters* e a medida Q será calculada com base neles. Como existem mais de 2 *clusters* reais nos dados, essas medidas não necessariamente serão maiores nas arestas de ligação entre dois *clusters* rotulados compactos.

Para explicar a solução encontrada nesse caso, precisamos esclarecer um detalhe sobre o número de *clusters* em que o conjunto de dados pode ser dividido. Suponha um conjunto de dados composto por n_C *clusters* rotulados compactos, tal que esses *clusters* foram separados corretamente a partir da poda das $n_C - 1$ arestas que faziam a ligação entre eles. Conforme mostrado anteriormente, neste caso obteremos o conjunto particionado em n_C *clusters* ideais com relação ao rótulo, tal que SSW referente a essa partição é igual a 0 e a medida Q é a maior possível ($Q = SSTO$) entre quaisquer outros n_C *clusters* que pudessem ser formados a partir de $n_C - 1$ podas na AGM original. Suponha que continuemos a podar arestas, dividindo o conjunto de dados em $n_C + 1$ *clusters* no próximo passo. Como os n_C *clusters* ideais já estavam formados, a próxima poda apenas irá dividir um deles em 2 partes. Se calcularmos novamente a medida Q com essa nova divisão, obteremos o mesmo valor que tinhamos anteriormente, ou seja, continuamos tendo *clusters* ideais com relação ao rótulo. A partir dessas observações, é fácil perceber que podemos ter n_C ou mais *clusters* ideais, de forma que a medida Q será sempre a maior possível (igual a $SSTO$). Dessa forma, consideremos o seguinte Lema:

Lema 2. *Considere um conjunto de dados V composto por n_C clusters rotulados compactos e $T(V, E_T)$ a AGM correspondente. Sejam $(u_i, v_i) \in E_T, i = 1, ..., n_C - 1$ as arestas de ligação entre os n_C clusters , ou seja, se $u_i \in C_j$ então $v_i \in C_k, k \neq j$. Considere uma partição de T em m_C sub-árvores, $T_1(V_1, E_{T_1}), ..., T_{m_C}(V_{m_C}, E_{T_{m_C}})$, onde $m_C \geq n_C$. Se $\bigcup_{i=1}^{n_C-1}(u_i, v_i) \notin \bigcup_{i=1}^{m_C} E_{T_i}$, então estas m_C sub-árvores representam m_C clusters ideais em relação ao rótulo.*

Prova. Suponha por absurdo que os m_C *clusters* definidos no enunciado do Lema não são ideais em relação ao rótulo. Então, para essa partição de V, temos que $Q < SSTO$, ou seja

$SSW \neq 0$. Neste caso, existem pelo menos dois vértices com rótulos distintos pertencentes a um mesmo *cluster* (um dos m_C *clusters* resultantes da partição enunciada). Consequentemente, existe uma aresta $w \in \bigcup_{i=1}^{m_C} E_{T_i}$ que faz a ligação entre esses dois vértices com rótulos distintos. Mas pelo Teorema 1, há exatamente $n_C - 1$ arestas que ligam pontos com rótulos distintos em T e, pelo enunciado do Teorema, nenhuma dessas arestas pertence a $\bigcup_{i=1}^{m_C} E_{T_i}$. Por contradição, concluimos que $SSW = 0$, $Q = SSTO$ e os m_C *clusters* definidos no enunciado do Teorema são ideais em relação ao rótulo.

É claro que o melhor é que o algoritmo consiga dividir o conjunto de dados em exatamente n_C *clusters* , evitando complexidade maior que a necessária. Mas no caso dele ser dividido em m_C, com $m_C > n_C$, de forma que cada um dos n_C *clusters* ideais sejam formados pela união de 1 ou mais dos m_C encontrados, isso não causará prejuízo nenhum na Definição das regiões de classificação corretas.

A metodologia LORC, no caso de um conjunto de dados composto por n_C *clusters* rotulados compactos, sempre obterá uma divisão desse conjunto em m_C *clusters* ideais com relação ao rótulo ($m_C \geq n_C$). O valor máximo de m_C necessário para que esteja assegurado que o método alcance $SSW = 0$ (e consequentemente $Q = SSTO$) depende do número de folhas da AGM correspondente. Já vimos na seção anterior, que no caso de apenas 2 *clusters* a aresta a ser podada na primeira iteração do LORC é a aresta de ligação entre os dois *clusters* . Portanto, nesse caso, não é necessário que o conjunto de dados seja particionado em mais de 2 para que as regiões de classificação sejam corretas e os 2 *clusters* ideais com relação ao rótulo, para os quais $Q = SSTO$, sejam encontrados.

Voltando então ao probLema de particionar um conjunto de dados composto por mais de 2 *clusters* rotulados compactos, vimos o comportamento do LORC nos *clusters* que têm apenas um vértice de uma aresta de ligação entre *clusters* no Lema 1. Agora falta verificar os demais *clusters* do conjunto de dados. Primeiramente, vejamos o Lema a seguir:

Lema 3. *Seja V um conjunto de dados composto por n_C clusters rotulados compactos $C_1, ..., C_{n_C}$ e $T(V, E_T)$ a AGM correspondente. Seja C_i um dos n_C clusters rotulados compactos de V, tal que $u_i \in C_i$ e $v_i \in C_i$ são vértices das arestas (u_i, u_j) e (v_i, v_k) que ligam C_i aos clusters rotulados compactos C_j e C_k, respectivamente. Considere uma aresta qualquer (r_i, s_i), com $r_i, s_i \in C_i$, tal que ao podar essa aresta da AGM serão formadas 2 sub-árvores e que cada uma delas possua pelo menos uma aresta de ligação entre 2 clusters compostos por elementos de rótulos diferentes. Então, a medida de dissimilaridade Q é maior ao considerar a poda de alguma das arestas (u_i, u_j) ou (v_i, v_j) do que ao podar qualquer outra aresta $(r_i, s_i) \in C_i$.*

Prova. Sem perda de generalidade, suponha que os vértices pertencentes aos *clusters* nomeados com índices ímpares $(C_1, C_3, ..., C_{n_C-1})$ têm rótulo 1 e os pertencentes aos de índices pares $(C_2, C_4, ..., C_{n_C})$ têm rótulo 0. Seja C_i um dos *clusters* que compõem V, tal que C_i tem pelo menos 2 vértices que são de arestas de ligação aos *clusters* C_{i-1} e C_{i+1}. A aresta (u_i, u_{i-1}) liga os *clusters* C_i e C_{i-1} e a aresta (v_i, v_{i+1}) liga os *clusters* C_i e C_{i+1}, com $u_{i-1} \in C_{i-1}, u_i, v_i \in C_i, v_{i+1} \in C_{i+1}$.

Suponha que o número de vértices em cada *cluster* $C_1, C_2, ..., C_{n_C}$ seja igual a $n_{C_1}, n_{C_2}, ..., n_{C_{n_C}}$, respectivamente. Ao considerar a poda de uma aresta (r_i, s_i) com as caracteristicas dadas no Lema, o conjunto V será dividido em 2 sub-árvores: uma formada pelos $n_{C_1} + n_{C_2} + ... + n_{Ci-1}$

pontos de outros *clusters* mais $n_{C_{ia}}$ pontos de C_i e outra formada pelos $n_{C_{i+1}} + n_{C_{i+2}} + ... + n_{C_{n_C}}$ pontos de outros *clusters* mais $n_{C_{ib}}$ pontos de C_i. Lembrando que $n_{C_{ia}} + n_{C_{ib}} = n_{C_i}$. Observe que, se $n_{C_{ia}} = 0$, a aresta podada é a (u_i, u_{i-1}) (aresta de ligação entre C_{i-1} e C_i) e se $n_{C_{ib}} = 0$, então a aresta podada é a (v_i, v_{i+1}) (aresta de ligação entre C_i e C_{i+1}). Dessa forma, mantendo os valores $n_{C_1}, n_{C_2}, ..., n_{C_{n_C}}$ fixos, podemos variar os valores de $n_{C_{ia}}$ e $n_{C_{ib}}$, sendo condicionados um ao outro, de forma a considerarmos todas as arestas de interesse. Nesse caso, o cálculo de Q é feito da seguinte forma:

$$Q{=}SSTO{-}\sqrt{\sum_{q=1}^{2}\sum_{i\in T_q}(y_i{-}p_q)^2}$$

$$\sum_{q=1}^{2}\sum_{i\in T_q}(y_i{-}p_q)^2{=}(n_{C_1}{+}n_{C_3}{+}...{+}n_{C_{i-1}})(1{-}\frac{{}^{n_{C_1}+n_{C_3}+...+n_{C_{i-1}}}}{{}^{n_{C_1}+n_{C_2}+...n_{C_{ia}}}})^2$$

$$+(n_{C_2}{+}n_{C_4}{+}...{+}n_{C_{ia}})(0{-}\frac{{}^{n_{C_1}+n_{C_3}+...+n_{C_{i-1}}}}{{}^{n_{C_1}+n_{C_2}+...n_{C_{ia}}}})^2$$

$$+(n_{C_{i+1}}{+}n_{C_{i+3}}{+}...{+}n_{C_{n_C-1}})(1{-}\frac{{}^{n_{C_{i+1}}+n_{C_{i+3}}+...+n_{C_{n_C-1}}}}{{}^{n_{C_{ib}}+n_{C_{i+1}}+n_{C_2}+...n_{C_n}}})^2$$

$$+(n_{C_{ib}}{+}n_{C_{i+2}}{+}...{+}n_{C_{n_C}})(0{-}\frac{{}^{n_{C_{i+1}}+n_{C_{i+3}}+...+n_{C_{n_C-1}}}}{{}^{n_{C_{ib}}+n_{C_{i+1}}+n_{C_2}+...n_{C_n}}})^2$$

$$=\frac{({}^{n_{C_1}+n_{C_3}+...+n_{C_{i-1}}})({}^{n_{C_2}+n_{C_4}+...+n_{C_{ia}}})}{{}^{n_{C_1}+n_{C_2}+...n_{C_{ia}}}}+\frac{({}^{n_{C_{i+1}}+n_{C_{i+3}}+...+n_{C_{n_C-1}}})({}^{n_{C_{ib}}+n_{C_{i+2}}+...+n_{C_{n_C}}})}{{}^{n_{C_{ib}}+n_{C_{i+1}}+n_{C_2}+...n_{C_n}}}$$

Como pelo menos 3 *clusters* devem existir para que o conjunto V seja formado por mais de 2 *clusters* rotulados compactos, conforme o enunciado, pelo menos 3 *clusters* devem ter 1 ou mais elementos. Então, temos que as funções

$$f_1(n_{C_{ia}}){=}\frac{({}^{n_{C_1}+n_{C_3}+...+n_{C_{i-1}}})({}^{n_{C_2}+n_{C_4}+...+n_{C_{ia}}})}{{}^{n_{C_1}+n_{C_2}+...n_{C_{ia}}}}{\geq}0$$

e

$$f_2(n_{C_{ib}}){=}\frac{({}^{n_{C_{i+1}}+n_{C_{i+3}}+...+n_{C_{n_C-1}}})({}^{n_{C_{ib}}+n_{C_{i+2}}+...+n_{C_{n_C}}})}{{}^{n_{C_{ib}}+n_{C_{i+1}}+n_{C_2}+...n_{C_n}}}{\geq}0$$

. Precisamos provar que o valor de Q é máximo quando $n_{C_{ia}} = 0$ ou $n_{C_{ib}} = 0$, casos em que a aresta podada é uma aresta de ligação entre *clusters* .

As funções $f_1(n_{C_{2a}})$ e $f_2(n_{C_{2b}})$ são funções crescentes, quando analisadas separadamente, pois suas derivadas são positivas:

$$\frac{\partial(f_1(n_{C_{ia}}))}{\partial n_{C_{ia}}}{=}\frac{({}^{n_{C_1}+n_{C_3}+...+n_{C_{i-1}}})^2}{({}^{n_{C_1}+n_{C_2}+...n_{C_{ia}}})^2}{>}0$$

e

$$\frac{\partial(f_2(n_{C_{ib}}))}{\partial n_{C_{ib}}}{=}\frac{({}^{n_{C_{i+1}}+n_{C_{i+3}}+...+n_{C_{n_C-1}}})^2}{({}^{n_{C_{ib}}+n_{C_{i+1}}+n_{C_2}+...n_{C_n}})^2}{>}0$$

.

Mas é importante lembrar que $n_{C_i} = n_{C_{ia}} + n_{C_{ib}}$ também é fixo, de forma que quando $n_{C_{ia}}$ aumenta, $n_{C_{ib}}$ diminui, e vice-versa. Dessa forma, podemos reescrever Q em função de apenas um dos valores variáveis, como $n_{C_{ia}}$, por exemplo (no caso de escrever em função de $n_{C_{ib}}$, os resultados são idênticos). Assim podemos derivar a função inteira e encontrar o ponto onde

ocorre o valor máximo de $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})(n_{C_2}+n_{C_4}+\ldots+n_{C_{ia}})}{n_{C_1}+n_{C_2}+\ldots n_{C_{ia}}}+\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})(n_{C_{ib}}+n_{C_{i+2}}+\ldots+n_{C_{n_C}})}{n_{C_{ib}}+n_{C_{i+1}}+n_{C_2}+\ldots n_{C_n}}$, que corresponde ao mínimo de Q. Teremos o seguinte:

$$\frac{\partial\left(\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})(n_{C_2}+n_{C_4}+\ldots+n_{C_{ia}})}{n_{C_1}+n_{C_2}+\ldots n_{C_{ia}}}+\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})(n_{C_i}-n_{C_{ia}}+n_{C_{i+2}}+\ldots+n_{C_{n_C}})}{n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+n_{C_2}+\ldots n_{C_n}}\right)}{\partial n_{C_{2a}}}=0$$

$$\frac{n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}}}{n_{C_1}+n_{C_2}+\ldots+n_{C_{ia}}}=\frac{n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C}}+1}{n_{C_{ib}}+n_{C_{i+1}}+\ldots+n_{C_{n_C}}}$$

Portanto, o valor mínimo de Q ocorrerá quando a proporção de rótulos 0's e 1's for a mesma nos 2 subconjuntos formados com a partição de V (caso essa igualdade ocorra para alguma combinação possível dos valores de $n_{C_{ia}}$ e $n_{C_{ib}}$). Caso não ocorra a igualdade citada, o mínimo será no extremo oposto ao máximo, ou seja, o valor de Q será crescente (ou decescente) em relação ao valor de $n_{C_{ia}}$, por exemplo. Já o valor máximo de Q ocorrerá em um dos extremos, ou seja, quando $n_{C_{ia}}=0$ ou $n_{C_{ib}}=0$, dependendo da proporção de rótulos 1's (rótulo distinto do rótulo dos elementos do *cluster* C_i) em cada subconjunto de V formado pela partição da AGM.

Resumindo, Q é máximo quando

$$\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})(n_{C_2}+n_{C_4}+\ldots+n_{C_{ia}})}{n_{C_1}+n_{C_2}+\ldots n_{C_{ia}}}+\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})(n_{C_{ib}}+n_{C_{i+2}}+\ldots+n_{C_{n_C}})}{n_{C_{ib}}+n_{C_{i+1}}+n_{C_2}+\ldots n_{C_n}}$$

é mínimo. Em suma, temos o seguinte:

- Se $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})}{(n_{C_1}+n_{C_2}+\ldots+n_{C_{i-1}})}*\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})}{(n_{C_1}+n_{C_2}+\ldots+n_{C_i})}>\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C}})}{(n_{C_{i+1}}+n_{C_{i+2}}+\ldots+n_{C_{n_C}})}*\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C}})}{(n_{C_i}+n_{C_{i+1}}+\ldots+n_{C_{n_C}})}$, então $\max(Q)$ ocorre quando $n_{C_{i_a}}=0$ e $n_{C_{i_b}}=n_{C_i}$.

- Se $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})}{(n_{C_1}+n_{C_2}+\ldots+n_{C_{i-1}})}*\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})}{(n_{C_1}+n_{C_2}+\ldots+n_{C_i})}<\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C}})}{(n_{C_{i+1}}+n_{C_{i+2}}+\ldots+n_{C_{n_C}})}*\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C}})}{(n_{C_i}+n_{C_{i+1}}+\ldots+n_{C_{n_C}})}$, então $\max(Q)$ ocorre quando $n_{C_{i_b}}=0$ e $n_{C_{i_a}}=n_{C_i}$.

- Se $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})}{(n_{C_1}+n_{C_2}+\ldots+n_{C_{i-1}})}*\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})}{(n_{C_1}+n_{C_2}+\ldots+n_{C_i})}=\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C}})}{(n_{C_{i+1}}+n_{C_{i+2}}+\ldots+n_{C_{n_C}})}*\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C}})}{(n_{C_i}+n_{C_{i+1}}+\ldots+n_{C_{n_C}})}$, então $\max(Q)$ ocorre quando $n_{C_{i_a}}=0$ e $n_{C_{i_b}}=n_{C_i}$ ou $n_{C_{i_b}}=0$ e $n_{C_{i_a}}=n_{C_i}$

Portanto, quaisquer que sejam as quantidades de elementos em cada um dos n_C *clusters* rotulados compactos, pelo menos uma das arestas de ligação terá valor do peso Q maior do que qualquer outra aresta que liga 2 pontos de C_i com as caracteristicas dadas. Logo, o resultado do Lema é válido para qualquer conjunto de dados V formado por n_C *clusters* rotulados compactos.

Podemos perceber então, que a única possibilidade de uma aresta que não é de ligação entre 2 *clusters* ser podada antes que todas as arestas de ligação o sejam, é se esta aresta dividir o conjunto de dados em 2 partes, da seguinte forma: um subconjunto é composto por uma parcela dos vértices de um *cluster* C_i (digamos n_{ia}) e a outra parcela é composta pelo restante dos vértices de C_i ($n_{ib}=n_i-n_{ia}$ elementos) e de todos os outros vértices do conjunto de dados. Para que isso ocorra, é necessário que o *cluster* C_i tenha um vértice cujo grau seja maior que 3, ou seja, é necessário que o *cluster* C_i tenha um vértice que seja uma folha da AGM.

Portanto, podemos concluir com o seguinte Teorema:

Teorema 3. *Considere um conjunto de dados V composto por n_C clusters rotulados compactos $C_1,...,C_{n_C}$ e $T(V,E_T)$ a AGM correspondente. Seja C_i um cluster que contém pelo menos 2 vértices u e v que são vértices de arestas de ligação a outros clusters , ou seja, C_i é unido na AGM a pelo menos outros 2 clusters rotulados compactos (ambos com rótulos distintos do rótulo de C_i). Seja n_i o número de elementos de C_i e n_{fi} o número de folhas da AGM contidas no cluster C_i. Então $(n_C-1)+\sum_{i=2}^{n_C-1}(n_{fi})$ é o maior número possível de arestas a serem podadas pelo LORC para que seja obtida uma partição de V composta por clusters ideais em relação ao rótulo (onde $Q = SSTO$).*

Prova. A prova do Teorema é direta, a partir dos resultados dos Lemas 1, 2, 3.

Na prática, os *clusters* reais que compõem o conjunto de dados são desconhecidos, sendo necessário considerar todas as folhas da árvore para obtér o número de podas necessárias. Ou seja,

$$m_C = (n_C - 1) + n_f,$$

onde n_f é o número total de folhas da AGM.

3.2.2.1 Solução Prática no Algoritmo

Já sabemos qual o número máximo de *clusters* em que o conjunto de dados deve ser dividido para que certamente sejam obtidos *clusters* ótimos com relação ao rótulo (neste caso de *clusters* rotulados compactos, são os *clusters* tais que $Q = SSTO$). Agora precisamos responder a seguinte pergunta: Qual o número mínimo de *clusters* em que o conjunto de dados deve ser dividido para que sejam obtidos *clusters* ótimos em relação ao rótulo? Com o objetivo de obter este número, observe que se $Q = SSTO$, então $SSW = 0$. Então, propomos um algoritmo simples (2) cujo resultado é o número que buscamos.

Algorithm 2 Obtenção do Número de *cluster* Necessários

1: **Entrada:** Todos os pontos do conjunto de treinamento
2: Constrói a AGM (algoritmo de Prim) a partir dos atributos (x) dos dados
3: Define o número máximo de *clusters* (NMC) = número de folhas da AGM
4: Executa o LORC com número de grupos igual a NMC
5: Encontra o menor número de grupos para o qual $SSW = 0$
6: **Saída:** Número de *clusters* mínimo necessário para que o LORC pode todas as arestas que ligam *clusters* distintos

3.2.3 Outros tipos de *clusters*

Quando o conjunto de dados é formado por n_C *clusters* rotulados que não são *clusters* rotulados compactos, o resultado do Teorema 1 não é válido, ou seja, pode haver mais que uma aresta de ligação entre dois *clusters* compostos por pontos com rótulos distintos. Portanto, para encontrar a partição ideal da AGM correspondente é necessário que mais de $n_C - 1$ arestas sejam podadas. Com o objetivo de definir o menor número de *clusters* possível para o qual o valor de Q seja

máximo (igual a $SSTO$), utilizamos novamente o algoritmo (2). Porém, há necessidade de modificarmos o valor do número máximo de *clusters* (NMC), que nesse caso não é igual ao número de folhas da AGM. Na realidade, o NMC é igual a n, ou seja, o número total de instâncias do conjunto de dados. Como podemos nos deparar com os mais diversos formatos de *clusters* , há possibilidades de conjuntos de dados nos quais o desempenho do método é ótimo (como no caso dos *clusters* rotulados compactos) assim como de outros em que, para encontrar os *clusters* ideiais em relação ao rótulo ($Q = SSTO$), seria necessário podar todas as arestas da AGM, particionando em n *clusters* , cada um composto por um único ponto. A Figura 3.2 representa um exemplo de formato de *cluster* em que o algoritmo apresentado teria este probLema. Na Seção 3.3, estes casos serão abordados mais detalhadamente. Para o caso específico apresentado na Figura 3.2, por exemplo, uma variação do método mais eficiente para este tipo de conjuntos de dados será apresentada na Seção 3.3.1.

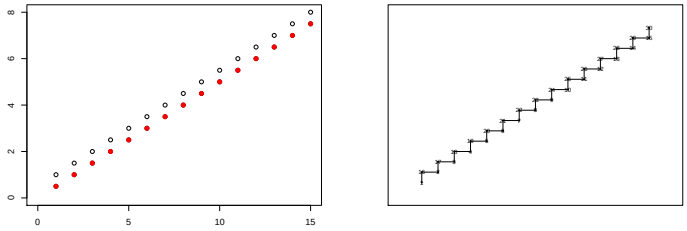


Figura 3.2 Exemplo de conjunto de dados no qual o método LORC não apresenta bom resultado. A partição teria que ser em n subconjuntos para alcançar $Q = SSTO$. À esquerda, os pontos em vermelho representam um rótulo e os pontos em preto representam o outro rótulo. À direita, temos a AGM correspondente.

3.3 Variações do LORC

Nesta seção apresentaremos variações da metodologia proposta que podem ser utilizadas em alguns casos de conjuntos de dados nos quais o método LORC tradicional, conforme apresentado, não tem bons resultados na etapa de particionar o grafo, definindo as regiões de classificação de forma incorreta. Nesses casos, algumas das variações podem solucionar o possível probLema, resultando em regiões de classificação mais próximas às ideais.

3.3.1 LORCy

O foco principal desta variação do LORC é lidar com conjuntos de dados cuja separação entre os *clusters* rotulados não é tão bem definida como ocorria com os *clusters* rotulados compactos. Conforme comentado anteriormente, em alguns conjuntos de dados com formatos de *clusters* diferentes dos rotulados compactos, o método LORC não mostra um bom desempenho. Isso ocorre principalmente quando existem muitos pontos de um *cluster* com determinado rótulo que são mais próximos a algum ponto de rótulo diferente do que dos demais pontos de mesmo

rótulo que pertencem ao seu cluster, como no exemplo da Figura 3.2. Este representa o pior cenário possível, visto que o LORC só consegue atingir uma partição ótima ao dividir o conjunto de dados em *n clusters* (*n* é o número de pontos do conjunto de dados). Portanto, vamos focar em casos desse tipo. Para isso, vamos definir os *clusters* rotulados complexos.

Definição 4 (*cluster* rotulados complexos)**.** Para uma dada métrica de distância, um *cluster* rotulado complexo é um conjunto de pontos V tal que para qualquer ponto v_i com um determinado rótulo y_i, existe um ponto v_k, com rótulo diferente de y_i, tal que $dist(v_i, v_j) > dist(v_i, v_k)$, para todo ponto v_j com rótulo igual a y_i.

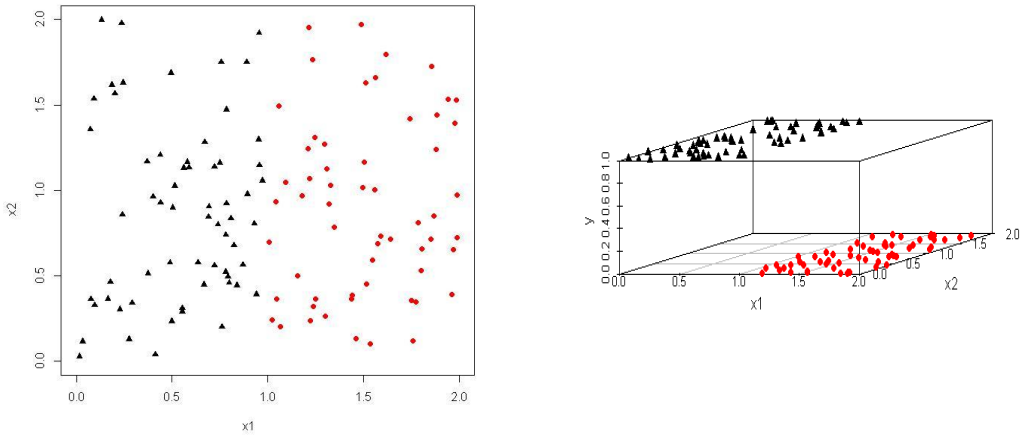
Buscando uma forma de particionar corretamente o conjunto de dados formado por *clusters* rotulados complexos, apresentamos uma variação do método LORC, que será designada pela sigla LORCy.

O LORCy segue exatamente os mesmo passos do LORC, a diferença se encontra apenas no cálculo dos pesos das arestas na etapa de construção da AGM. Novamente, utilizaremos a distância euclidiana para atribuição dos pesos, ou seja, o custo associado à aresta $(v_i; v_j)$ é $(dist(v_i, v_j))^{-1}$. Porém agora a distância não é calculada com base apenas nos vetores de atributos $(x_{i1}; ...; x_{ik})$, mas nos vetores completos que caracterizam cada elemento da nossa base de dados $(x_{i1}; ...; x_{ik}; y_i)$. Dessa forma, a diferença desses métodos é que o rótulo y_i de cada elemento do conjunto de treinamento do modelo será levado em conta no momento de calcular os pesos das arestas do grafo e, consequentemente, no momento da construção da árvore geradora mínima (AGM).

É importante ressaltar que a etapa de poda da AGM para obtenção dos *clusters* não é alterada, ou seja, a medida de heterogeneidade utilizada continua se baseando apenas nas respostas y_i's.

Fica claro que o LORCy obterá bons resultados ao particionar um conjunto de dados formado por *clusters* rotulados complexos. Mas existem situações menos extremas em que ele também é mais adequado do que o LORC. Suponha que temos um conjunto de dados em 2 dimensões, tal que a conFiguração do conjunto de treinamento do modelo é a exibida na Figura 3.3(a). É importante observar que os dois grupos não estão "misturados", ou seja, os *clusters* são bem definidos. Eles estão claramente divididos se observarmos os rótulos, porém não há uma distância razoável entre eles, ao considerar apenas os atribitos $(x_1; x_2)$. Podemos observar que esse conjunto de dados não atende as regras de um conjunto composto por *clusters* rotulados compactos. Nesse caso, se utilizarmos o método clássico LORC, não conseguiremos identificar os 2 *clusters* corretos ao particionar o conjunto de dados inicial em 2 subconjuntos. Já no caso de utilizarmos o LORCy, fica claro que obteremos maior exito na Definição da partição correta. Na Figura 3.3(b) temos uma visualização da conFiguração dos dados ao considerar a resposta.

Não é difícil perceber que LORCy também tem resultado ótimo para conjuntos de dados formados por clusters rotulados compactos. Nesse caso, pode surgir a questão: Se o método tem, comprovadamente, solução ótima em uma gama mais ampla de conjuntos de dados que o LORC, porque não utilizarmos sempre ele? A resposta é que, quando o conjunto de dados tem instâncias mal rotuladas (ruído no rótulo), o LORC será capaz de fazer partições do espaço melhores que o LORCy, gerando melhores resultados para a classificação. Veremos mais detalhadamente esse caso no próximo capítulo.



(a) Visão sem levar em conta a dimensão da resposta

(b) Visão levando em conta a dimensão da resposta

Figura 3.3 Exemplo de cenário que inspirou a modificação do método

3.3.2 Random LORC e Random LORCy

Com a intenção de minimizar efeitos no treinamento do algoritmo de minorias de instâncias mal rotuladas ou anômalas existentes no conjunto de dados, formulamos uma segunda variação do método. A modifição proposta pode ser utilizada tanto no LORC quanto no LORCy de forma análoga. A implementação desta variação no LORC será denominada "*Random* LORC"e no LORCy será chamada de "*Random* LORCy". Ela é baseada em um conceito muito interessante e bastante utilizado atualmente, que é a técnica de reamostragem *bootstrap*. Vários esquemas diferentes de simulação *Bootstrap* têm sido propostos na literatura e muitos deles apresentam bom desempenho em uma ampla variedade de situações.

O método de simulação *Bootstrap* foi originalmente proposto por [Efron, 1979]. O método tem por base a idéia de que podemos tratar nossa amostra como se ela fosse a população que deu origem aos dados e usar amostragem com reposição da amostra original para gerar pseudoamostras. A partir destas pseudoamostras, é possível estimar características da população, tais como média, variância, percentis, etc.

No nosso caso, a idéia é utilizar a técnica de reamostragem *Bootstrap* da seguinte forma: geramos j pseudoamostras de nossa amostra original (das n instâncias que compõem o nosso conjunto de dados de treinamento do modelo V) e executamos o método LORC (ou o LORCy) para cada uma dessas amostras. A cada uma dessas j iterações, teremos uma região de classificação formada pela partição resultante da aplicação do método na amostra selecionada. Para uma nova instância a ser classificada na próxima etapa, o rótulo atribuído em cada um dos j cenários obtidos será registrado. Finalmente, observamos qual foi a classifacação mais frequente desse novo elemento e esse será o rótulo atribuído a ele.

A principal vantagem dessa variação ocorre quando o conjunto de dados de treinamento V tem um pequeno percentual de ruído no rótulo. Nesse caso, em muitas das amostras *Bootstrap* a maior parte desses pontos que são ruído podem ficar de fora. Assim elas têm menor possibilidade de atrapalhar a classificação de novos pontos, gerando resultados ruins. Este tipo de conjunto de dados será analisado detalhadamente do capítulo 4.

CAPÍTULO 4

O Método LORC em Conjuntos de Dados com Ruído no Rótulo

O probLema de ruído no rótulo em classificação supervisionada é bastante complexo. Na Seção 2.4, o conceito foi explicado. Nesta seção vamos verificar como a metodologia LORC e suas variações se comportam em conjuntos de dados com essa característica. As demonstrações vistas na seção anterior, quando tratamos de conjuntos de dados sem ruído no rótulo, serão reformuladas com o objetivo de demonstrar matematicamente a eficiência da metodologia LORC também para o tipo de dados definido nesta seção.

4.1 A Metodologia do LORC em Conjuntos de Dados com Ruído no Rótulo

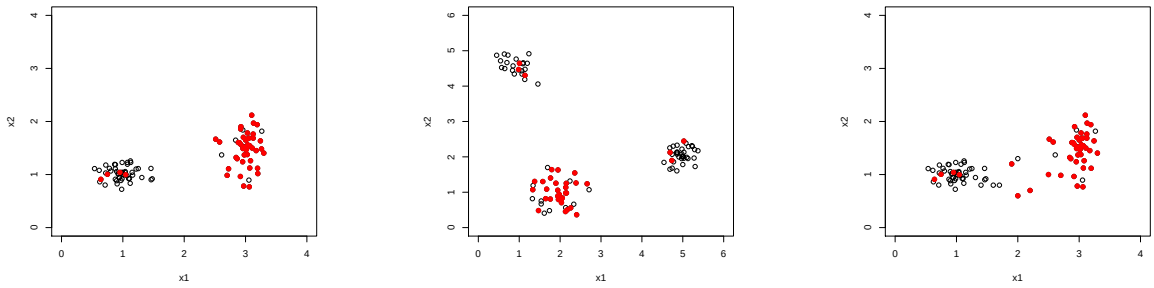
Nesta seção faremos as demonstrações matemáticas de eficiência da metologia LORC ao tratar de conjuntos de dados com ruído no rótulo para o treinamento do algoritmo de aprendizagem de máquina. O caminho percorrido é o mesmo das demonstrações apresentadas para conjuntos de dados sem ruído no rótulo. Serão refeitas definições e reformulados Lemas, Teoremas e as respectivas demontrações para os novos tipos de conjuntos de dados a serem tratados. Para começar, vamos definir o tipo de conjunto de dados que iremos considerar nesta parte.

Definição 5 (*Clusters* Compactos)**.** Considere um conjunto de pontos rotulados V. Para uma dada métrica de distância, um *cluster* compacto C é um sub-conjunto de V tal que para qualquer ponto $v_i \in C$, $dist(v_i, v_j) < dist(v_i, v_k)$, para todo ponto $v_j \in C$ e todo ponto $v_k \notin C$.

Observe que a Definição 5 é semelhante à Definição 1, de *clusters* rotulados compactos. A diferença é que na Definição 1 os *clusters* compactos têm todos os elementos com o mesmo rótulo. Já a Definição 5 permite elementos com rótulos diferentes dentro de um mesmo *cluster* compacto, de forma a possibilitar *clusters* compactos com ruído no rótulo de seus componentes.

A Figura 4.1 mostra exemplos de dois conjuntos de dados com 10% de pontos mal rotulados com ruído do tipo NCAR, sendo que o representado em 4.1(a) é composto por 2 *clusters* compactos e o representado em 4.1(b) por 3 *clusters* compactos. Na Figura 4.1(c) o conjunto de dados é formado por 2 *clusters* que não atendem a Definição 5, ou seja, não são *clusters* compactos. Os pontos em vermelho têm rótulo 1 e os pretos têm rótulo 0.

A principal diferença que encontraremos nas demonstrações da eficiência do LORC para *clusters* compactos com ruído no rótulo (diferentemente dos *clusters* rotulados compactos analisados no Capítulo 3), é que nem sempre será possível atingir a partição ideal (na qual $SSW = 0$



(a) (b) (c)

Figura 4.1 Exemplos de Conjuntos de Dados Formados por *Clusters* que atendem a Definição de Rotulados Compactos (em 4.1(a) e 4.1(b)) e que não a atendem (em 4.1(c)). As cores representam os rótulos de cada instância.

e $Q = SSTO$), pois essa medida é calculada com base nos rótulos observados. Nesse caso, quando as regiões de classificação corretas são obtidas pelo método, o valor de Q não alcança o de $SSTO$. Portanto, o objetivo agora é maximizar o valor de Q, mas sabendo que mesmo que os *clusters* encontrados sejam os ideais, esse valor será menor que $SSTO$. Precisamos então, reescrever a Definição 2, de forma que os *clusters* ótimos com relação ao rótulo são idênticamente definidos, porém a parte referente aos *clusters* ideais é modificada:

Definição 6 (*Clusters* ótimos em relação ao rótulo e *Clusters* ideais em relação ao rótulo)**.** Ao particionar um conjunto de dados rotulados em C *clusters* a partir da poda de $C-1$ arestas da AGM correspondente, os C *clusters* ótimos com relação ao rótulo são os que resultam no valor máximo de Q, definido em (3.1). Caso os *clusters* ótimos em relação ao rótulo obtidos sejam exatamente os representados nos dados (ou subconjuntos deles), diremos que além de ótimos com relação ao rótulo, eles são os *clusters* ideais em relação ao rótulo.

Com base nas novas definições, o Teorema 1 fica da seguinte forma:

Teorema 4. *Seja um conjunto de dados rotulados V com a respectiva AGM $T(V, E_T)$. Se V é formado por n_C clusters compactos, então existem exatamente $n_C - 1$ arestas em E_T que ligam pontos pertencentes a clusters distintos. Isso significa que se existe ligação entre pontos de dois clusters distintos de V, essa ligação é feita por uma única aresta $e \in E_T$.*

Prova. Observe que a demonstração do Teorema 1 não utiliza o fato dos dados não apresentarem ruído no rótulo (*clusters* compactos formados por elementos de mesmo rótulo). Dessa forma, a demonstração deste Teorema 4 é análoga à que foi desenvolvida para o Teorema 1, apenas substituindo as definições adequadamente.

Podemos perceber então, que para conjuntos de dados compostos por *clusters* compactos, se existe ligação entre quaisquer dois *clusters*, essa ligação também será sempre realizada por apenas uma aresta na AGM. As mesmas implicações vistas para os conjuntos de dados sem ruído no rótulo poderão ser verificadas neste tipo de situação, como veremos a seguir.

Antes de verificarmos o Teorema principal (Teorema 2) para este caso, precisamos mostrar que outros resultados são válidos, a começar pelo Lema que diz que a aresta com maior valor correspondente de Q é a aresta correta a ser retirada. É claro que alguns dos resultados não são verdadeiros para qualquer conjunto de dados com ruído no rótulo. Dessa forma, as demonstrações daqui em diante se limitarão a conjuntos de dados com ruído no rótulo segundo o modelo de ruído completamente aleatório (NCAR) e o modelo de ruído aleatório (NAR). Como estamos tratando de dados binários, cada *cluster* cujo rótulo original (sem troca de rótulos) é 0 terá o mesmo percentual de troca de rótulos, assim como o percentual de troca de rótulos para todos os *clusters* cujo rótulo original é 1 também será o mesmo. Usaremos também a suposição de que cada subconjunto de algum desses *clusters* (seja do *cluster* C_i, por exemplo) que venha a ser formado a partir da poda de uma aresta da AGM, será composto pelo mesmo percentual de rótulos 0's e 1's do *cluster* original. Ou seja, subconjuntos formados a partir de uma partição de C_i terão o mesmo percentual de pontos com rótulos 0's e de 1's que C_i.

Focando nos modelos NCAR e NAR, vamos verificar que o Lema 4 (equivalente ao Lema 1 que foi definido para dados sem ruído no rótulo) é válido, para posteriormente provarmos que o resultado do Teorema 2 vale também ao tratarmos de conjuntos de dados compostos por 2 *clusters* compactos se torna relativamente fácil. Então, considere o Lema 4:

Lema 4. *Dado um conjunto de dados V composto por n_C clusters compactos, $T(V,E_T)$ é a AGM correspondente. Considere que o conjunto V tem dados com ruído no rótulo segundo o modelo NAR. Seja C_1 um dos n_C clusters rotulados compactos de V, tal que só existe um vértice $v_a \in C_1$ que seja vértice de uma aresta (v_a, v_b) de ligação de C_1 com outro cluster rotulado compacto C_2 de V, onde $(v_a, v_b) \in E_T$. Então, a medida de dissimilaridade $Q(v_a, v_b)$ é maior do para qualquer outra aresta $Q(v_{a'}, v_{a''})$ tal que $v_{a'}, v_{a''} \in C_1$.*

Prova. Sem perda de generalidade, vamos fazer algumas suposições:

- Suponha que o número de instâncias em cada um dos *clusters* seja representado da seguinte forma: cada *cluster* C_i é composto por n_{C_i} vértices, com $i = 1, 2, ..., n_C$.
- Suponha que um *cluster* C_i qualquer entre os n_C *clusters* compactos seja ligado aos *clusters* C_{i-1} e C_{i+1} por duas arestas distintas em E_T.
- Suponha também que os *clusters* C_i's tais que i é ímpar são originalmente formados por intâncias com rótulo 0 e os C_i's com i par por instâncias com rótutlo 1.
- Suponha que em cada *cluster* C_i originalmente formado por intâncias com rótulo 0, $x\%$ dos rótulos sejam iguais a 1 e em cada *cluster* C_i originalmente formado por intâncias com rótulo 1, $y\%$ dos rótulos sejam iguais a 0. Dessa forma, há $x\%$ de troca de rótulo nos grupos onde o rótulo original é 0 e $y\%$ nos que o rótulo original é 1. Observe que se $x \neq y$, temos um conjunto de dados com ruído do tipo NAR. Se $x = y$, temos um conjunto de dados com ruído do tipo NCAR.

Vamos calcular o valor de Q ao podar a aresta (v_a, v_b) (chamaremos este de Q_1) e também ao podar uma outra aresta $(v_{a'}, v_{a''})$ qualquer, tal que $v_{a'}, v_{a''} \in C_1$ (chamaremos este de Q_2). Observe que ao podar a aresta $(v_{a'}, v_{a''})$ estaremos dividindo o *cluster* C_1 em 2 sub-clusters

isolados um do outro, um deles, com $n_{C_{1b}}$ instâncias, ligado a C_2 por (v_a, v_b) e outro, com $n_{C_{1a}}$ instâncias, sem nenhuma ligação aos demais *clusters*. Seja qual for o valor de n_C, o cálculo de Q_1 se dá da seguinte forma:

$$Q_1 = SSTO - \sqrt{\sum_{q=1}^{2} \sum_{i \in T_q} (y_i - p_q)^2}$$

$$\sum_{q=1}^{2} \sum_{i \in T_q} (y_i - p_q)^2 = (1-x) n_{C_1} (0 - \tfrac{x}{n_{C_1}} n_{C_1})^2 + x n_{C_1} (1 - \tfrac{x}{n_{C_1}} n_{C_1})^2$$

$$+ [y(n_{C_2} + n_{C_4} + ...) + (1-x)(n_{C_3} + n_{C_5} + ...)](0 - \frac{(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_3} + n_{C_5} + ...)}{n_{C_2} + n_{C_3} + n_{C_4} + ...})^2$$

$$+ [(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_3} + n_{C_5} + ...)](1 - \frac{(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_3} + n_{C_5} + ...)}{n_{C_2} + n_{C_3} + n_{C_4} + ...})^2$$

$$= x(1-x) n_{C_1} + \frac{[y(n_{C_2} + n_{C_4} + ...) + (1-x)(n_{C_3} + n_{C_5} + ...)][(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_3} + n_{C_5} + ...)]}{n_{C_2} + n_{C_3} + n_{C_4} + ...}$$

Então, $Q_1 = SSTO - \sqrt{x(1-x) n_{C_1} + \frac{[y(n_{C_2} + n_{C_4} + ...) + (1-x)(n_{C_3} + n_{C_5} + ...)][(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_3} + n_{C_5} + ...)]}{n_{C_2} + n_{C_3} + n_{C_4} + ...}}$.

Para o cálculo de Q_2, ao tirarmos uma aresta qualquer que une dois pontos pertencentes ao mesmo *cluster* C_1, temos o seguinte:

$$Q_2 = SSTO - \sqrt{\sum_{q=1}^{2} \sum_{i \in T_q} (y_i - p_q)^2}$$

$$\sum_{q=1}^{2} \sum_{i \in T_q} (y_i - p_q)^2 = (1-x) n_{C_{1a}} (0 - \tfrac{x}{n_{C_{1a}}} n_{C_{1a}})^2 + x n_{C_{1a}} (1 - \tfrac{x}{n_{C_{1a}}} n_{C_{1a}})^2$$

$$+ [y(n_{C_2} + n_{C_4} + ...) + (1-x)(n_{C_{1b}} + n_{C_3} + ...)](0 - \frac{(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_{1b}} + n_{C_3} + ...)}{n_{C_{1b}} + n_{C_2} + n_{C_3} + ...})^2$$

$$+ [(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_{1b}} + n_{C_3} + ...)](1 - \frac{(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_{1b}} + n_{C_3} + ...)}{n_{C_{1b}} + n_{C_2} + n_{C_3} + + ...})^2$$

$$= x(1-x) n_{C_{1a}} + \frac{[y(n_{C_2} + n_{C_4} + ...) + (1-x)(n_{C_{1b}} + n_{C_3} + ...)][(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_{1b}} + n_{C_3} + ...)]}{n_{C_{1b}} + n_{C_2} + n_{C_3} + n_{C_4} + ...}$$

Então, $Q_2 = SSTO - \sqrt{x(1-x) n_{C_{1a}} + \frac{[y(n_{C_2} + n_{C_4} + ...) + (1-x)(n_{C_{1b}} + n_{C_3} + ...)][(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_{1b}} + n_{C_3} + ...)]}{n_{C_{1b}} + n_{C_2} + n_{C_3} + n_{C_4} + ...}}$.

Temos então os valores de Q_1 e Q_2 e queremos verificar se $Q_1 > Q_2$, como diz o Lema. Então vamos fazer a comparação procedendo com as devidas manipulações algébricas para simplificar as expressões, e considerando as equivalências seguintes:

$$Q_1 > Q_2$$

$$\equiv x(1-x) n_{C_1} + \frac{[y(n_{C_2} + n_{C_4} + ...) + (1-x)(n_{C_3} + n_{C_5} + ...)][(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_3} + n_{C_5} + ...)]}{n_{C_2} + n_{C_3} + n_{C_4} + ...}$$

$$< x(1-x) n_{C_{1a}} + \frac{[y(n_{C_2} + n_{C_4} + ...) + (1-x)(n_{C_{1b}} + n_{C_3} + ...)][(1-y)(n_{C_2} + n_{C_4} + ...) + x(n_{C_{1b}} + n_{C_3} + ...)]}{n_{C_{1b}} + n_{C_2} + n_{C_3} + n_{C_4} + ...}$$

$$\equiv 0 < (x+y)^2 - 2(x+y) + 1$$

Como $0 \leq x \leq 1$ e $0 \leq y \leq 1$, temos uma inequação de segundo grau, representada por uma parábola voltada para cima. Para qualquer valor possível de $x + y$, que pode variar entre 0 e 2, essa inequação é válida, exceto quando $x + y = 1$, caso em que o parábola em questão

encosta o eixo horizontal (é igual a 0). Na prática do probLema que estamos analisando, se $x+y=1$, temos que o percentual mínimo de troca de rótulo em alguma das classes é 50%. Isso significa que há mais de metade de rótulos trocados em uma das classes (e menos de 50% na outra classe) ou então que há 50% de troca em ambas categorias. Nestes casos, é esperado que nenhum algoritmo particione corretamente o conjunto de dados de forma a encontrar as regiões corretas. Portanto, desde que haja menos de metade dos rótulos trocados em ambas as categorias, a última desigualdade é verdadeira e, a partir das equivalências, fica demonstrado o Lema.

Agora vamos considerar o caso geral, quando o conjunto de dados V é formado por n_C *clusters* compactos, permitindo que haja ruído no rótulo do tipo NAR (consequentemente do tipo NCAR também). Primeiramente veremos que no caso dos conjuntos de dados formados por *clusters* compactos (considerando a possibilidade de ruído no rótulo do tipo NAR ou NCAR), também é válido o resultado do Lema 2. Para expicar mais claramente o próximo Lema a ser apresentado, suponha um conjunto de dados composto por n_C *clusters* compactos, tal que esses *clusters* foram separados corretamente a partir da poda das n_C-1 arestas que faziam a ligação entre eles. Conforme mostrado anteriormente, neste caso obteremos o conjunto particionado em n_C *clusters* ideais com relação ao rótulo, tal que o valor da medida Q para esta partição é a maior possível (e, consequentemente, o valor de SSW o menor possível) entre qualquer outra partição de V em n_C *clusters* que pudessem ser formados a partir de n_C-1 podas na AGM original. Suponha que continuemos a podar arestas, dividindo o conjunto de dados em n_C+1 *clusters* no próximo passo. Como os n_C *clusters* ideais já estavam formados, a próxima poda apenas irá dividir um deles em 2 partes, lembrando que os subconjuntos formados terão a mesma proporção de elementos de cada rótulo do *cluster* original. Se calcularmos novamente a medida Q com essa nova divisão, obteremos o mesmo valor que tinhamos anteriormente. A partir dessas observações, é fácil perceber que podemos ter n_C ou mais *clusters* ideais, de forma que a medida Q será sempre a maior possível (igual a $SSTO$). Dessa forma, consideremos o seguinte Lema:

Lema 5. *Considere um conjunto de dados V composto por n_C clusters compactos e $T(V,E_T)$ a AGM correspondente. Sejam $(u_i,v_i)\in E_T, i=1,...,n_C-1$ as arestas de ligação entre os n_C clusters, ou seja, se $u_i\in C_j$ então $v_i\in C_k, k\neq j$. Considere uma partição de T em m_C sub-árvores, $T_1(V_1,E_{T_1}),...,T_{m_C}(V_{m_C},E_{T_{m_C}})$, onde $m_C\geq n_C$. Se $\bigcup_{i=1}^{n_C-1}(u_i,v_i)\notin\bigcup_{i=1}^{m_C}E_{T_i}$, então estas m_C sub-árvores representam uma partição de V em m_C clusters cujo valor de Q é igual ao valor de Q referente à partição de V nos n_C clusters ideais com relação ao rótulo.*

Prova.

Suponha que em cada *cluster* C_i originalmente formado por instâncias com rótulo 0, x% dos rótulos sejam iguais a 1 (rótulos trocados) e em cada *cluster* C_i originalmente formado por intâncias com rótulo 1, z% dos rótulos sejam iguais a 0. Dessa forma, há x% de troca de rótulo nos grupos onde o rótulo original é 0 e z% nos que o rótulo original é 1. Observe que se $x\neq z$, temos um conjunto de dados com ruído do tipo NAR. Se $x=z$, temos um conjunto de dados com ruído do tipo NCAR.

Em cada um dos n_C *clusters* compactos que foram obtidos a partir da partição de V nos n_C *clusters* descritos no enunciado do Lema, a proporção de instâncias com rótulo 1 é igual a x

(nos *clusters* cujo rótulo original é 0) ou $1-z$ (nos *clusters* cujo rótulo original é 1). Vamos calcular o valor de Q referente a essa partição (denominado Q_1):

$$Q_1 = SSTO - SSW = SSTO - \sum_{q=1}^{n_C} \sum_{i \in T_q} (y_i - p_q)^2$$

$$\sum_{q=1}^{n_C} \sum_{i \in T_q} (y_i - p_q)^2 = \sum_{qa=1}^{n_{C_1}} \sum_{i \in T_{qa}} (y_i - (1-z))^2 + \sum_{q_b=1}^{n_{C_0}} \sum_{i \in T_{q_b}} (y_i - x)^2$$

$$\sum_{q=1}^{n_C} \sum_{i \in T_q} (y_i - p_q)^2 = n_1[y(1-y)] + n_0[x(1-x)]$$

onde n_{C_1} é o número de *clusters* compactos de V compostos originalmente por instâncias de rótulo 1 e n_{C_0} o número de *clusters* compostos originalmente por instâncias de rótulo 0, n_1 é o número total de pontos que compõem os *clusters* cujo rótulo original era 1 e n_0 é o número total de pontos que compõem os *clusters* cujo rótulo original era 0.

Agora, suponha que um desses n_C *clusters* seja dividido em 2 subgrupos: C_a e C_b. Nesse caso, temos que $m_C = n_C + 1$. Sem perda de generalidade, suponha que o *cluster* que foi particionado gerando C_a e C_b era originalmente composto por 0, de forma que tanto o *cluster* original quanto os dois subclusters resultantes têm $x\%$ de seus elementos com rótulos 1's e $(1-x)\%$ com rótulos 0's. Fica fácil perceber que o valor de Q calculado para a nova partição de V não será alterado. E o mesmo ocorrerá caso sejam podadas novas arestas, desde que $\bigcup_{i=1}^{n_C-1}(u_i, v_i) \notin \bigcup_{i=1}^{m_C} E_{T_i}$. Dessa forma, temos que $Q = Q_1$, e fica demonstrado o Lema 5.

Finalmente, para mostrar que a quantidade de arestas a serem podadas antes das arestas corretas (as de ligação entre *clusters*) é limitada, precisamos redefinir e provar o análogo ao Lema 3, o que está sendo apresentado a seguir, no Lema 6:

Lema 6. *Seja V um conjunto de dados composto por n_C clusters compactos $C_1, ..., C_{n_C}$ e seja $T(V, E_T)$ a AGM correspondente. Considere que o conjunto V tem dados com ruído no rótulo segundo o modelo NAR. Seja C_i um dos n_C clusters compactos de V, tal que $u_i, v_i \in C_i$ são vértices das arestas (u_i, u_j) e (v_i, v_k) que ligam C_i aos clusters compactos C_j e C_k, respectivamente. Considere uma aresta qualquer (r_i, s_i), com $r_i, s_i \in C_i$, tal que ao podar essa aresta da AGM serão formadas 2 sub-árvores e que cada uma delas possua pelo menos uma aresta de ligação entre 2 clusters compostos por elementos de rótulos diferentes. Então, a medida de dissimilaridade Q é maior ao considerar a poda de alguma das arestas (u_i, u_j) ou (v_i, v_j) do que ao podar qualquer outra aresta $(r_i, s_i) \in C_i$.*

Prova. Sem perda de generalidade, suponha que os vértices pertencentes aos *clusters* nomeados com índices ímpares $(C_1, C_3, ..., C_{n_C-1})$ têm rótulo originalmente igual a 1 e os pertencentes aos de índices pares $(C_2, C_4, ..., C_{n_C})$ têm rótulo originalmente igual a 0, exceto os vértices que estão com os rótulos trocados. Suponha que em cada *cluster* originalmente formado por instâncias com rótulo 0, $x\%$ dos rótulos sejam trocados (iguais a 1) e em cada *cluster* originalmente formado por intâncias com rótulo 1, $y\%$ dos rótulos sejam trocados (iguais a 0). Dessa forma, há $x\%$ de troca de rótulo nos grupos onde o rótulo original é 0 e $y\%$ nos que o rótulo original é 1. Observe que se $x \neq y$, temos um conjunto de dados com ruído do tipo NAR. Se $x = y$, temos um conjunto de dados com ruído do tipo NCAR.

Seja C_i um dos *clusters* que compõem V, tal que C_i tem 2 vértices que são de arestas de ligação aos *clusters* C_{i-1} e C_{i+1}. A aresta (u_i, u_{i-1}) liga os *clusters* C_i e C_{i-1} e a aresta (v_i, v_{i+1}) liga os *clusters* C_i e C_{i+1}, com $u_{i-1} \in C_{i-1}, u_i, v_i \in C_i, v_{i+1} \in C_{i+1}$.

Suponha que o número de vértices em cada *cluster* $C_1, C_2, ..., C_{n_C}$ seja igual a $n_{C_1}, n_{C_2}, ..., n_{C_{n_C}}$, respectivamente. Ao considerar a poda de uma aresta (r_i, s_i) com as caracteristicas dadas no Lema, o conjunto V será dividido em 2 sub-árvores: uma formada pelos $n_{C_1} + n_{C_2} + ... + n_{Ci-1}$ pontos de outros *clusters* mais $n_{C_{ia}}$ pontos de C_i e outra formada pelos $n_{C_{i+1}} + n_{C_{i+2}} + ... + n_{C_{n_C}}$ pontos de outros *clusters* mais $n_{C_{ib}}$ pontos de C_i. Lembrando que $n_{C_{ia}} + n_{C_{ib}} = n_{C_i}$. Observe que, se $n_{C_{ia}} = 0$, a aresta podada é a (u_i, u_{i-1}) (aresta de ligação entre C_{i-1} e C_i) e se $n_{C_{ib}} = 0$, então a aresta podada é a (v_i, v_{i+1}) (aresta de ligação entre C_i e C_+i). Dessa forma, mantendo os valores $n_{C_1}, n_{C_2}, ..., n_{C_{n_C}}$ fixos, podemos variar os valores de $n_{C_{ia}}$ e $n_{C_{ib}}$, sendo condicionados um ao outro, de forma a considerarmos todas as arestas de interesse. Nesse caso, o cálculo de Q é feito da seguinte forma:

$$Q = SSTO - \sqrt{\sum_{q=1}^{2} \sum_{i \in Tq} (y_i - p_q)^2}$$

$$\sum_{q=1}^{2} \sum_{i \in Tq} (y_i - p_q)^2 = [(1-x)(n_{C_1} + n_{C_3} + ... + n_{C_{i-1}}) + y(n_{C_2} + n_{C_4} + ... n_{C_{ia}})](1 - \frac{(1-x)(n_{C_1} + n_{C_3} + ... + n_{C_{i-1}}) + y(n_{C_2} + n_{C_4} + ... n_{C_{ia}})}{n_{C_1} + n_{C_2} + ... n_{C_{ia}}})^2$$

$$+ [x(n_{C_1} + n_{C_3} + ... + n_{C_{i-1}}) + (1-y)(n_{C_2} + n_{C_4} + ... n_{C_{ia}})](0 - \frac{(1-x)(n_{C_1} + n_{C_3} + ... + n_{C_{i-1}}) + y(n_{C_1} + n_{C_2} + ... n_{C_{ia}})}{n_{C_1} + n_{C_2} + ... n_{C_{ia}}})^2$$

$$+ ([(1-x)(n_{C_{i+1}} + n_{C_{i+3}} + ... + n_{C_{n_C-1}}) + y(n_{C_{ib}} + n_{C_{i+2}} + ... + n_{C_{n_C}})](1 - \frac{(1-x)(n_{C_{i+1}} + n_{C_{i+3}} + ... + n_{C_{n_C-1}}) + y(n_{C_{ib}} + n_{C_{i+2}} + ... + n_{C_{n_C}})}{n_{C_{ib}} + n_{C_{i+1}} + ... n_{C_n}})^2$$

$$+ ([x(n_{C_{i+1}} + n_{C_{i+3}} + ... + n_{C_{n_C-1}}) + (1-y)(n_{C_{ib}} + n_{C_{i+2}} + ... + n_{C_{n_C}})](0 - \frac{x(n_{C_{i+1}} + n_{C_{i+3}} + ... + n_{C_{n_C-1}}) + (1-y)(n_{C_{ib}} + n_{C_{i+2}} + ... + n_{C_{n_C}})}{n_{C_{ib}} + n_{C_{i+1}} + ... n_{C_n}})^2$$

$$= (y(1-y))(n_{C_2} + n_{C_4} + ... n_{C_i} + n_{C_{i+2}} + ... + n_{C_{n_C}}) + ((1-x)(1-y) + xy - y(1-y))(n_{C_1} + n_{C_3} + ... + n_{C_{i-1}} + n_{C_{i+1}} + n_{C_{i+3}} + ... + n_{C_{n_C-1}})$$

$$- ((x+y)^2 - 2(x+y) + 1)(\frac{(n_{C_1} + n_{C_3} + ... + n_{C_{i-1}})^2}{n_{C_1} + n_{C_2} + ... n_{C_{ia}}} + \frac{(n_{C_{i+1}} + n_{C_{i+3}} + ... + n_{C_{n_C-1}})^2}{n_{C_{ib}} + n_{C_{i+1}} + ... n_{C_n}})$$

Precisamos provar que o valor de Q é máximo quando $n_{C_{ia}} = 0$ ou $n_{C_{ib}} = 0$, casos em que considerada para a poda é uma aresta de ligação entre *clusters*. Observe que a parcela que depende dos valores de $n_{C_{ia}}$ e $n_{C_{ib}}$ é apenas a da última linha do resultado acima, cujo sinal que a antecede é negativo. O coeficiente $(1-x)(1-y) + xy - x(1-x) - y(1-y) \geq 0$ para quaisquer valores de x e y. Dessa forma, o valor de Q será máximo quando o valor de $\frac{(n_{C_1} + n_{C_3} + ... + n_{C_{i-1}})^2}{n_{C_1} + n_{C_2} + ... n_{C_{ia}}} + \frac{(n_{C_{i+1}} + n_{C_{i+3}} + ... + n_{C_{n_C-1}})^2}{n_{C_{ib}} + n_{C_{i+1}} + ... n_{C_n}}$ for máximo.

É claro que as funções $\frac{(n_{C_1} + n_{C_3} + ... + n_{C_{i-1}})^2}{n_{C_1} + n_{C_2} + ... n_{C_{ia}}}$ e $\frac{(n_{C_{i+1}} + n_{C_{i+3}} + ... + n_{C_{n_C-1}})^2}{n_{C_{ib}} + n_{C_{i+1}} + ... n_{C_n}}$ são funções decrescentes em relação a $n_{C_{ia}}$ e $n_{C_{ib}}$, respectivamente, quando analisadas separadamente. Mas é importante lembrar que $n_{C_i} = n_{C_{ia}} + n_{C_{ib}}$ também é fixo, de forma que quando $n_{C_{ia}}$ aumenta, $n_{C_{ib}}$ diminui, e vice-versa. Dessa forma, podemos reescrever Q em função de apenas uma das variáveis, como $n_{C_{ia}}$, por exemplo (no caso de escrever em função de $n_{C_{ib}}$, os resultados são idênticos). Assim podemos derivar a função como um todo para observar as tendências de crescimentos e/ou decrescimento e encontrar possíveis pontos de máximo e mínimo. Teremos o seguinte:

$$\frac{\partial(\frac{(n_{C_1} + n_{C_3} + ... + n_{C_{i-1}})^2}{n_{C_1} + n_{C_2} + ... n_{C_{ia}}} + \frac{(n_{C_{i+1}} + n_{C_{i+3}} + ... + n_{C_{n_C-1}})^2}{n_{C_i} - n_{C_{ia}} + n_{C_{i+1}} + ... n_{C_n}})}{\partial n_{C_{2a}}}$$

$$=-\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2}+\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2}$$

$$\begin{cases} =0, & \text{se} \quad \frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2} = \frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2} \\ <0, & \text{se} \quad \frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2} > \frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2} \\ >0, & \text{se} \quad \frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2} < \frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2} \end{cases}$$

Logo, se para alguma combinação possível dos valores de $n_{C_{ia}}$ e $n_{C_{ib}}$ ocorrer $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2}=\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2}$, estes serão os valores correspondentes ao mínimo de Q. Nesse caso, o máximo de Q irá ocorrer quando $n_{C_{ia}}=0$ e $n_{C_{ib}}=n_{C_i}$ ou quando $n_{C_{ia}}=n_{C_i}$ e $n_{C_{ib}}=0$. Caso não ocorra essa igualdade para nenhuma combinação possível de $n_{C_{ia}}$ e $n_{C_{ib}}$, teremos duas possibilidades: a derivada será sempre negativa, caso $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2}>\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2}$ para todo valor de $n_{C_{ia}}$ e $n_{C_{ib}}$; ou a derivada sempre será positiva, caso $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2}<\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2}$ para todo valor de $n_{C_{ia}}$ e $n_{C_{ib}}$. Na hipótese da derivada ser sempre negativa, o mínimo da função (correspondente ao mínimo de Q) ocorerá quando $n_{C_{ia}}=n_{C_i}$ e $n_{C_{ib}}=0$ e o máximo da função (correspondente ao mínimo de Q) quando $n_{C_{ia}}=0$ e $n_{C_{ib}}=n_{C_i}$. Na hipótese da derivada ser sempre positiva, o mínimo da função (correspondente ao mínimo de Q) ocorerá quando $n_{C_{ia}}=0$ e $n_{C_{ib}}=n_{C_i}$ e o máximo da função (correspondente ao máximo de Q) quando $n_{C_{ia}}=n_{C_i}$ e $n_{C_{ib}}=0$.

Em suma, temos o seguinte:

- Se $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2}>\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2}$ para todo valor de $n_{C_{ia}}$, tal que $0\leq n_{C_{ia}}\leq n_{C_i}$, então $\max(Q)$ ocorre quando $n_{C_{ia}}=n_{C_i}$ e $n_{C_{ib}}=0$.

- Se $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2}<\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2}$ para todo valor de $n_{C_{ia}}$, tal que $0\leq n_{C_{ia}}\leq n_{C_i}$, então $\max(Q)$ ocorre quando $n_{C_{ia}}=0$ e $n_{C_{ib}}=n_{C_i}$.

- Se $\frac{(n_{C_1}+n_{C_3}+\ldots+n_{C_{i-1}})^2}{(n_{C_1}+n_{C_2}+\ldots n_{C_{ia}})^2}=\frac{(n_{C_{i+1}}+n_{C_{i+3}}+\ldots+n_{C_{n_C-1}})^2}{(n_{C_i}-n_{C_{ia}}+n_{C_{i+1}}+\ldots n_{C_n})^2}$ para algum valor de $n_{C_{ia}}$, tal que $0\leq n_{C_{ia}}\leq n_{C_i}$, então $\max(Q)$ ocorre quando $n_{C_{ia}}=0$ e $n_{C_{i_b}}=n_{C_i}$ ou $n_{C_{i_a}}=n_{C_i}$ e $n_{C_{i_b}}=0$

Portanto, quaisquer que sejam as quantidades de elementos em cada um dos n_C *clusters* compactos, pelo menos uma das arestas de ligação terá valor do peso Q maior do que qualquer outra aresta que liga 2 pontos de C_i com as caracteristicas dadas. Logo, o resultado do Lema é válido para qualquer conjunto de dados V formado por n_C *clusters* compactos.

Temos agora todas as ferramentas necessárias para verficarmos a validade do Teorema final, no caso dos conjuntos de dados com ruído no rótulo definidos neste Capítulo:

Teorema 5. *Considere um conjunto de dados V composto por n_C clusters compactos $C_1, ..., C_{n_C}$ e $T(V, E_T)$ a AGM correspondente. Seja C_i um cluster que contém pelo menos 2 vértices u e v que são vértices de arestas de ligação a outros clusters, ou seja, C_i é unido na AGM a pelo menos outros 2 clusters compactos. Seja n_i o número de elementos de C_i e n_{fi} o número de folhas da AGM contidas no cluster C_i. Então $(n_C - 1) + \sum_{i=2}^{n_C-1}(n_{fi})$ é o maior número possível de arestas a serem podadas pelo LORC para que seja obtida uma partição de V cujo valor de Q é o mesmo calculado a partir da partição de V composta pelos n_C clusters ideais em relação ao rótulo.*

Prova. A prova do Teorema é direta, a partir dos resultados dos Lemas 4, 5, 6.

Portanto, temos a completa demonstração de eficiência do LORC para conjuntos de dados formados por *clusters* compactos (que por Definição podem ter ruído no rótulo). Observe que todas as demonstrações desta seção também servem para dados sem ruído no rótulo (conjunto de dados formado por *clusters* rotulados compactos), já que estes seriam um caso específico daqueles, quando o percentual de dados com rótulo trocado é igual a 0. Dessa forma, a teoria desenvolvida neste Capítulo é suficiente para englobar todos os tipos de conjuntos de dados tratados também no Capítulo anterior, servindo como uma demonstração matemática geral da eficiência da metodologia LORC.

4.1.1 Definição do número de *clusters*

No caso de conjuntos de dados com ruído no rótulo, nem sempre será possível atingir a partição ideal (na qual $SSW = 0$ e, consequentemente $Q = SSTO$), pois essa medida é calculada com base nos rótulos observados. Dessa forma, caso as regiões de classificação corretas sejam obtidas pelo método, mesmo assim o valor de Q não alcançará o de $SSTO$, pois SSW não será igual a 0. Portanto, o algoritmo que define o melhor número de *clusters* a ser utilizado precisa considerar essa característica, como mostrado no Algoritmo 3.

Algorithm 3 Obtenção do Melhor Número de *Cluster*s

1: **Entrada:** Todos os n pontos do conjunto de treinamento
2: Constrói a AGM (algoritmo de Prim) a partir dos atributos (x) dos dados
3: Define o número máximo de *clusters* (NMC)
4: Executa o LORC com número de grupos igual a NMC
5: Encontra o menor número de grupos n_C para o qual $SSW(n_C) - SSW(n_C - 1) < \frac{SSTO}{n}$. Caso não ocorra $SSW(n_C) - SSW(n_C - 1) < \frac{SSTO}{n}$ para nenhum valor possível de $n_C > 1$, então define $n_C = \frac{n}{5}$.
6: **Saída:** Número de grupos n_C

À medida que a o número de *clusters* n_C considerado vai aumentando, o valor de SSW vai diminuindo. Estabelecemos então um limite para a diferença entre dois valores de SSW referentes a valores subsequentes de n_C. Dessa forma, ao avaliarmos que ao dividir o conjunto de dados em mais de n_C *clusters* o ganho não seria grande em relação a Q, definimos o valor de n_C.

CAPÍTULO 5

Aplicações a Dados Simulados

Este capítulo apresenta os resultados da aplicação do algoritmo proposto e suas variações em diversos conjuntos de dados simulados. Todos os testes foram executados utilizando o programa R [R Core Team, 2014]. Primeiramente, foram simulados conjuntos de dados com 2 atributos para cada instância, sem nenhum tipo de ruído. Posteriormente, o desempenho do método foi avaliado em alguns desses mesmos conjuntos, introduzindo diferentes tipos e intensidades de ruído no rótulo. Além disso, também foi simulado um conjunto de dados com ruído nos atributos. Os desenhos dos experimentos e os resultados obtidos serão apresentados detalhadamente neste capítulo.

5.1 Descrição Geral

Para avaliar o método, foram simulados oito conjuntos de dados construídos da seguinte forma:

- Sete conjuntos de dados de formatos distintos com atributos em 2 dimensões. Essa escolha dos dados em 2 dimensões se justifica pela facilidade em observar a disposição dos pontos que correspondem aos valores dos atributos no plano cartesiano, caracterizando com facilidade a real partição do espaço.

- Um conjunto de dados com atributos em 20 dimensões, no qual somente 2 das 20 variáveis são relevantes para a classificação, sendo as demais variáveis de ruído.

Para cada conjunto de dados foram gerados dois sub-conjuntos, um para construção do modelo e outro para teste da qualidade do ajuste. O conjunto de construção do modelo é a base da regra de decisão que será utilizada na classificação dos novos pontos. Na prática, este conjunto contém os dados dos atributos e também da resposta, enquanto o conjunto de teste, que consiste dos pontos a serem classificados, não tem informação da resposta. No caso dos dados simulados, os pontos do conjunto de teste são gerados com a resposta, porém na hora da classificação essa informação é omitida. Ela será utilizada após a classificação ser concluída, com o objetivo de verificar a acurácia do método.

Para efeitos de comparação e avaliação da eficiência do método desenvolvido, foram aplicadas aos dados as quatro variações da metodologia proposta (LORC, Random LORC, LORCy e Random LORCy) e outros cinco métodos populares de classificação supervisionada: Regressão Logística, Árvores de Classificação e Regressão(CART), Florestas Aleatórias, Máquinas de Vetores de Suporte (SVM) e k Vizinhos Mais Próximos (kNN). A principal medida de avaliação considerara foi percetual de acertos na classificação dos pontos do conjunto de teste. Além dessa, também foram avaliadas as medidas de Sensibilidade (Revocação), Especificidade

e Precisão. A descrição dos métodos de classificação supervisionada aplicados e das medidas utilizadas já foram descritas detalhadamente no Capítulo 2.

5.2 Definição dos Parâmetros

Alguns dos métodos utilizados neste trabalho, inclusive o LORC e sua variações, exigem que parâmetros sejam informados para que o algoritmo possa ser executado. No Capítulo 2, foram discutidos os métodos e a função de cada parâmetro. Para obtermos um bom desempenho na classificação, procuramos obtér valores para os parâmetros que fossem mais adequados aos conjuntos de dados simulados. Os métodos cujos parâmetros foram analisados são os seguintes:

- SVM: O SVM tem dois parâmetros a serem definidos (C e γ). No R, os valores default desses parâmetros são $C = 1$ e $\gamma = \frac{1}{n}$. Para verificar os melhores valores dos parâmetros, a função *tune()* (implementada no pacote *e1071* do R) foi aplicada a cada conjunto de dados variando os dois ($C = 1, 10$ e $\gamma = 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1$). Para cada combinação possível entre um valor de cada parâmetro, verificou-se a performance do método no conjunto de dados em questão. A dupla de valores com melhor desempenho foi selecionada e utilizada como parâmetros para os próximos testes com o SVM.

- kNN: No kNN é necessário definir o número de vizinhos k que será utilizado na classificação de um novo ponto. Para cada valor de k variando de 1 a $\frac{n}{10}$, onde n é o número de elementos do conjunto de dados de treinamento do modelo, a função *tune()* avaliou o desempenho do método. O valor k referente ao melhor desempenho foi selecionado e utilizado como parâmetro para os próximos testes com o kNN.

- LORC e variações: No LORC em suas variações são necessários 2 parâmetros: o número de arestas que serão podadas da AGM (ou o número de *cluster* em que o conjunto de dados será dividido) e o número de vizinhos k que será utilizado na classificação de um novo ponto. O primeiro parâmetro é definido durante a execução do método, conforme exibido anteriormente. Para estabelecer o valor de k, o procedimento é o mesmo que utilizamos para o kNN: para cada valor de k variando de 1 a $\frac{n}{10}$, onde n é o número de elementos do conjunto de dados de treinamento do modelo, a função *tune()* avaliou o desempenho do método. O valor k referente ao melhor desempenho foi selecionado e utilizado como parâmetro para os próximos testes com o LORC, assim como para cada uma das variações. É importante observar que o valor do parâmetro é selecionado para cada variação, podendo ser diferente em cada uma delas.

5.3 Conjuntos de Dados Sem Ruído no Rótulo

Inicialmente fizemos alguns testes para conjuntos de dados simulados, sem introduzir nenhum ruído no rótulo, afim de observar o comportamento das variações da metodologia LORC e dos demais algoritmos considerados em determinadas situações de interesse. A seguir estão descritos os conjuntos de dados simulados que foram utilizados nesta seção (e que também

serviram de base para os testes da próxima seção, na qual será introduzido ruído no rótulo) e os testes realizados, seguidos pelos resultados observados e alguns comentários a respeito.

5.3.1 Os Conjuntos de Dados Simulados

Foram gerados ao todo 8 cenários distintos de configurações dos pontos. As figuras a seguir mostram um exemplo de configuração de cada cenário, sendo que os pontos vermelhos representam um rótulo (0) e os pontos pretos representam outro (1). A Figura 5.1 mostra exemplos de conjuntos de dados de construção do modelo em 5 dos cenários propostos. No cenário 1 (Figura 5.1(a)), foram formados 3 *clusters* dispostos em forma "diagonal". No cenário 2 (Figura 5.1(b)), foram formados 2 *clusters* circulares, um dentro do outro. No cenário 3 (Figura 5.1(c)), foram formados 4 *clusters* dispostos de forma adequada a um bom desempenho do algoritmo do CART (e do Random Forest, consequentemente), ou seja, separáveis através de divisões realizadas ao longo dos eixos cartesianos. No cenário 4 (Figura 5.1(d)) é a junção dos cenários 1 e 2, formando 4 *clusters*. E finalmente, o cenário 5 (Figura 5.1(e)) é a junção dos cenários 1 e 3, formando 5 *clusters*. Para cada um desses 5 cenários iniciais, o conjunto de dados apresentado nas figuras consiste em 200 pontos.

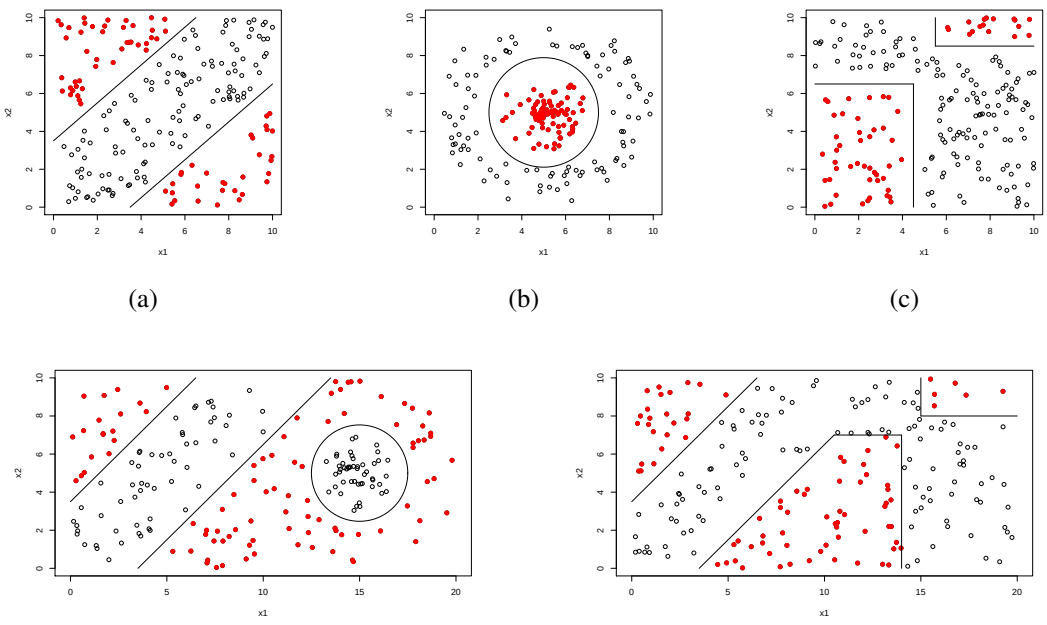


Figura 5.1 Configurações de pontos de 5 cenários simulados para teste dos algoritmos

Observe que nenhum dos 5 cenários apresentados é composto por *clusters* rotulados compactos, de forma que a correta definição das regiões de classificação fique menos óbvia. Dessa forma, se o número de *clusters* reais que forma um determinado conjunto de dados é n_C, pode haver mais de $n_C - 1$ arestas unindo *clusters* com rótulos distintos. Conforme discutido no

capítulo 3, para estes tipos de conjuntos de dados, nos quais também não há misturas entre as instâncias de grupos diferentes, supomos que as variações do LORC com melhor desempenho seriam as que usam o rótulo na construção da AGM, ou seja, a LORCy e a Random LORCy.

O Cenário 6 consiste em um conjunto de dados de 20 dimensões, no qual apenas 2 são relevantes para a classificação. O conjunto de dados é formado por 2 *clusters*. Cada um dos 18 atributos foi gerado de uma distribuição normal, com mesma média e variância para todos os pontos. Os outros 2 atributos (que são importantes para a classificação) foram gerados de distribuições normais com média e variância distintas para os diferentes rótulos. As duas dimensões cujos parâmetros são diferentes (relevantes) foram plotadas na Figura 5.2.

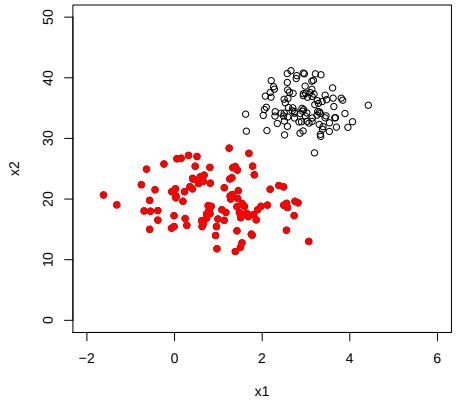


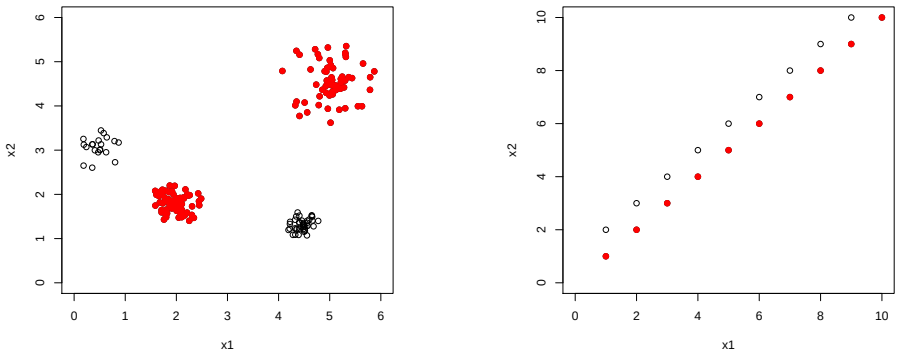
Figura 5.2 Configuração de pontos das 2 variáveis relevantes no Cenário 6

Para lidar com os casos extremos (seja de sucesso ou de fracasso do método), foram simulados outros cenários, conforme mostra a Figura 5.3. No cenário 7 (Figura 5.3(a)) temos um conjunto de dados composto por 4 *clusters* rotulados compactos, todos gerados de distribuições normais com médias e variâncias distintas. Já o cenário 8 (Figura 5.3(b)) representa o caso de fracasso do LORC, sendo um conjunto de dados composto por 2 *clusters* rotulados complexos. Ambos os conjuntos apresentados na Figura 5.3 foram construidos com 200 pontos, porém apenas uma parte do gráfico está exibida em 5.3(b), para melhor vizualização do formato do conjunto.

A Tabela 5.1 apresenta um resumo dos conjuntos de dados simulados que foram utilizados nos testes de desempenho dos algoritmos de classificação supervisionada.

5.3.2 Aplicações e Resultados

O foco principal da metodologia LORC desenvolvida neste trabalho é lidar com conjuntos de dados que têm o problema de ruído no rótulo. Portanto, os testes em conjuntos de dados simulados nesta seção terão como objetivo principal avaliar o desempenho do método neste tipo de conjunto de dados.



(a) (b)

Figura 5.3 Configurações de pontos dos 2 cenários simulados que representam casos de sucesso e fracasso do LORC

Conjunto	Clusters	Atributos	Características
C1	3	2	Não apresenta *clusters* rotulados compactos
C2	2	2	Não apresenta *clusters* rotulados compactos
C3	3	2	Não apresenta *clusters* rotulados compactos
C4	4	2	Não apresenta *clusters* rotulados compactos
C5	4	2	Não apresenta *clusters* rotulados compactos
C6	2	20	Tem variáveis de ruído e não apresenta *clusters* rotulados compactos
C7	4	2	Apresenta *clusters* rotulados compactos
C8	2	2	Apresenta *clusters* rotulados complexos

Tabela 5.1 Resumo dos conjuntos de dados simulados utilizados para avaliação dos métodos de classificação supervisionada

Inicialmente, utilizamos os conjuntos de dados simulados originais, sem que fosse introduzido nenhum ruído no rótulo, para avaliar algumas características interessantes que pudessem resultar em melhores desempenhos de classificação. Esses testes iniciais foram realizados para definirmos parâmetros a serem utilizados nos testes seguintes, nos quais o ruído no rótulo estará presente. Dessa forma, as seções 5.3.2.1 e 5.3.2.2 apresentam, respectivamente, avaliações da influência da quantidade de elementos e do percentual de elementos em cada classe de rótulo nos conjuntos de dados de treinamento dos modelos e a seção 5.3.2.3 mostra uma análise da variação média dos resultados de classificação obtidos por um mesmo método ao variar os conjuntos de dados utilizados. Além disso, também utilizamos os conjuntos de dados sem ruído para avaliar a complexidade dos algoritmos, medida a partir do tempo de processamento, o que será apresentado na seção 5.3.2.4.

5.3.2.1 Número de elementos nos conjuntos de dados

A primeira parte dos testes foi realizada com o objetivo de avaliar se o número de instâncias presentes no conjunto de dados de treinamento do modelo em relação ao número de instâncias no conjunto de teste tem influência no desempenho dos algoritmos de classificação supervisionada.

Para esta análise, os métodos foram avaliados em todos os conjuntos simulados sem ruído no rótulo, variando o tamanho do conjunto de treinamento (começando com 50 instâncias e terminando com 250, aumentando de 50 em 50) e mantendo fixo o número de instâncias do conjunto de teste (igual a 100). Para cada um dos tamanhos estabelecidos para o conjunto de treinamento, cada método foi aplicado 10 vezes, sorteando aleatoriamente as instâncias do conjunto de treinamento (dentro dos desenhos estabelecidos para cada conjunto de dados) e mantendo fixo o conjunto de teste. O desempenho foi avaliado pela média do percentual de acertos na classificação das 50 instâncias do conjunto de teste obtida nas 10 aplicações de cada algoritmo. Como o interesse nesta etapa de testes é avaliar apenas a influência do número de instâncias do conjunto de treinamento, os resultados apresentados na Tabela 5.2 mostram a média de acertos de cada método, já considerando todos os 8 conjuntos de dados avaliados. Os resultados obtidos para cada conjunto de dados, individualmente, estão no apêndice.

	50	100	150	200	250
LORC	0.8038	0.8338	0.8579	**0.8717**	0.8711
LORCy	0.8713	0.8929	0.9065	**0.9133**	0.9073
Random LORC	0.8349	0.8572	0.8743	**0.8856**	0.8799
Random LORCy	0.8766	0.9093	0.92	**0.9266**	0.922
Reg. Logística	0.7047	0.7024	0.7142	**0.7252**	0.7228
CART	0.7261	0.8014	0.8451	0.863	**0.8631**
Flor. Aleatórias	0.8749	0.9486	0.971	**0.9774**	0.9738
SVM	0.9218	0.9778	0.9854	**0.9862**	0.9802
kNN	0.8673	0.8911	0.9008	**0.9078**	0.8989

Tabela 5.2 Percentual médio de acertos dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo. A média foi obtida a partir dos resultados dos 8 conjuntos de dados avaliados.

A partir dos resultados apresentados podemos observar que, em geral, para todos os métodos aplicados, o aumento do número de elementos do conjunto de construção de modelo tende a melhorar o desempenho do algoritmo na classificação de novos pontos. Este resultado era esperado, visto que a medida que a região na qual os pontos estão distribuídos tende a ficar melhor definida quando o número de pontos que a preenche é maior. Dessa forma, é importante utilizarmos o maior número possível de instâncias disponíveis para realizar o treinamento de um algoritmo de classificação supervisionada, afim de obtér bons resultados na classificação de novas instâncias.

Podemos observar ainda que ao aumentarmos o número de elementos no conjunto de dados de treinamento do modelo de 200 para 250, não há mais aumento médio na acurácia (percentual

de acertos) média verificada para os métodos aplicados. Dessa forma, para os conjuntos de dados simulados utilizados nesta seção de testes, ficou decidido que o tamanho a ser utilizado para os conjuntos de dados de treinamento dos modelos de classificação será de 200 instâncias, enquando os conjuntos de dados de teste serão composto por 100 instâncias cada um.

Definidos os tamanhos dos conjuntos de dados de treinamento dos algoritmos para os dados simulados, vamos observar uma outra característica que pode influenciar na acurácia das classificações: o percentual de elementos em cada classe de rótulo no conjunto de dados de treinamento utilizados. Para os testes que foram feitos nesta seção, o número de elementos em cada classe de rótulo havia sido definido de forma aleatória, ou seja, os valores referentes aos atributos de cada ponto foram sorteados uniformemente dentro do intervalo de valores pré-definidos para cada um deles e, posteriormente, observou-se a região onde ele estava localizado para atribuir o rótulo adequado. Dessa maneira, o percentual de elementos em cada classe de rótulo é aproximadamente proporcional à área (ao tratar de 2 dimensões) teórica delimitada para os *clusters* relativos a cada classe.

Porém, pode ser interessante observar se alterações nessas proporções de elementos em cada classe de rótulo podem influenciar no desempenho dos algoritmos. A próxima seção de testes analisará este tópico.

5.3.2.2 Percentual de elementos em cada classe de rótulo

Esta etapa de testes foi executada com o objetivo de observar se diferentes quantidades relativas de elementos em cada classe de rótulos no conjunto de treinamento influenciam no desempenho dos métodos de classificação supervisionada. A intenção é verificar qual o percentual percentual de pontos em cada classe que resulta, em geral, no melhor desempenho dos métodos. A partir dos resultados obtidos, poderemos ecolher os valores a serem utilizados nos testes posteriores com os conjuntos de dados simulados.

Para esta análise, os métodos foram avaliados em todos os conjuntos simulados sem ruído no rótulo, exceto o Conjunto de Dados 8 (composto por *clusters* rotulados complexos) que, por sua construção, não permite uma composição diferente de 50% de elementos em cada classe de rótulos. O número total n de instâncias nos conjuntos de dados de treinamento dos modelos ($n = 200$) foi mantido fixo e o percentual de instâncias em cada classe de rótulo foi variado (classe 0 com 10%, 25%, 50%, 75% e 90% e classe 1 com os percentuais complementares). Para cada um dos percentuais estabelecidos para cada classe de rótulos do conjunto de treinamento, cada método foi aplicado 10 vezes, sorteando aleatoriamente as instâncias do conjunto de treinamento (dentro dos desenhos estabelecidos para cada conjunto de dados) e mantendo fixo o conjunto de teste. O desempenho foi avaliado pelo percentual médio de acertos na classificação das 100 instâncias do conjunto de teste obtida nas 10 aplicações de cada algoritmo. Como o interesse nesta etapa de testes é avaliar os melhores resultados, em geral, para cada conjunto de dados, é interessante apresentar os resultados de cada um deles separadamente. O percentual médio de acertos observado está apresentado nas Tabelas a seguir.

Para o Cenário 1, obtivemos que a categoria com o maior percentual médio de acertos foi a categoria na qual cada classe contém 50% dos pontos.

Para o Cenário 2, obtivemos que a categoria com o maior percentual médio de acertos foi a categoria na qual a classe de pontos com rótulo 0 tem 75% dos pontos e a classe com rótulo 1

	10%/90%	25%/75%	50%/50%	75%/25%	90%/10%
LORC	0.799	0.858	0.91	0.918	0.874
LORCy	0.926	0.984	0.98	0.949	0.912
Random LORC	0.731	0.858	0.919	0.86	0.818
Random LORCy	0.861	0.965	0.975	0.956	0.936
Reg. Logística	0.4	0.5	0.491	0.57	0.61
CART	0.7	0.865	0.887	0.843	0.719
Flor. Aleatórias	0.737	0.938	0.962	0.932	0.853
SVM	0.945	0.989	0.987	0.965	0.969
kNN	0.887	0.978	0.975	0.95	0.914
Média	0.7762	0.8817	**0.8984**	0.8826	0.845

Tabela 5.3 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de elementos em cada classe de rótulos (o primeiro valor é o percentual de dados com rótulo 0 e o último é o percentual de dados com rótulo 1) no conjunto de treinamento, mantendo o conjunto de teste fixo. Resultado para o Conjunto de Dados 1

	10%/90%	25%/75%	50%/50%	75%/25%	90%/10%
LORC	0.791	0.876	0.906	0.968	0.988
LORCy	0.88	0.975	0.994	0.997	0.994
Random LORC	0.705	0.803	0.949	0.974	0.983
Random LORCy	0.786	0.916	0.986	0.985	0.99
Reg. Logística	0.49	0.536	0.548	0.49	0.46
CART	0.84	0.947	0.968	0.98	0.912
Flor. Aleatórias	0.891	0.943	0.983	0.989	0.981
SVM	0.89	0.951	0.994	0.997	0.987
kNN	0.856	0.948	0.987	0.993	0.994
Média	0.7031	0.8772	0.9239	**0.9303**	0.921

Tabela 5.4 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de elementos em cada classe de rótulos (o primeiro valor é o percentual de dados com rótulo 0 e o último é o percentual de dados com rótulo 1) no conjunto de treinamento, mantendo o conjunto de teste fixo. Resultado para o Conjunto de Dados 2

contém 25% dos pontos.

Para o Cenário 3, obtivemos que a categoria com o maior percentual médio de acertos foi a categoria na qual a classe de pontos com rótulo 0 tem 75% dos pontos e a classe com rótulo 1 contém 25% dos pontos.

Para o Cenário 4, obtivemos que a categoria com o maior percentual médio de acertos foi a categoria na qual cada classe contém 50% dos pontos.

Para o Cenário 5, obtivemos que a categoria com o maior percentual médio de acertos foi a categoria na qual a classe de pontos com rótulo 0 tem 75% dos pontos e a classe com rótulo 1 contém 25% dos pontos.

Para o Cenário 6, obtivemos que a categoria com o maior percentual médio de acertos foi a categoria na qual cada classe contém 50% dos pontos.

Para o Cenário 7, obtivemos que a categoria com o maior percentual médio de acertos foi a

	10%/90%	25%/75%	50%/50%	75%/25%	90%/10%
LORC	0.848	0.874	0.934	0.964	0.909
LORCy	0.917	0.923	0.95	0.97	0.913
Random LORC	0.798	0.849	0.924	0.962	0.883
Random LORCy	0.856	0.903	0.946	0.963	0.916
Reg. Logística	0.39	0.307	0.578	0.74	0.69
CART	0.829	0.953	0.975	0.985	0.945
Flor. Aleatórias	0.93	0.952	0.971	0.99	0.911
SVM	0.878	0.916	0.958	0.982	0.95
kNN	0.908	0.904	0.953	0.972	0.918
Média	0.8171	0.8423	0.9099	**0.9476**	0.8928

Tabela 5.5 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de elementos em cada classe de rótulos (o primeiro valor é o percentual de dados com rótulo 0 e o último é o percentual de dados com rótulo 1) no conjunto de treinamento, mantendo o conjunto de teste fixo. Resultado para o Conjunto de Dados 3

	10%/90%	25%/75%	50%/50%	75%/25%	90%/10%
LORC	0.789	0.782	0.732	0.685	0.555
LORCy	0.867	0.914	0.901	0.83	0.613
Random LORC	0.737	0.724	0.705	0.591	0.493
Random LORCy	0.796	0.892	0.93	0.871	0.679
Reg. Logística	0.61	0.56	0.539	0.44	0.46
CART	0.75	0.8	0.86	0.74	0.545
Flor. Aleatórias	0.712	0.855	0.928	0.874	0.664
SVM	0.851	0.9	0.949	0.89	0.715
kNN	0.83	0.876	0.915	0.844	0.632
Média	0.7713	0.8114	**0.8288**	0.7517	0.5951

Tabela 5.6 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de elementos em cada classe de rótulos (o primeiro valor é o percentual de dados com rótulo 0 e o último é o percentual de dados com rótulo 1) no conjunto de treinamento, mantendo o conjunto de teste fixo. Resultado para o Conjunto de Dados 4

categoria na qual a classe de pontos com rótulo 0 tem 25% dos pontos e a classe com rótulo 1 contém 75% dos pontos.

A partir dos resultados apresentados, temos os percentuais de elementos em cada rótulo que proporciona o melhor desempenho médio na classificação de novos pontos para a maioria dos algoritmos testados, para cada conjunto de dados. Os resultados resumidos que serão utilizados nos próximos testes estão apresentados na Tabela 5.10.

5.3.2.3 Desvio-padrão nos resultados de classificação

Para uma melhor comparação dos métodos de classificação nas próximas seções, achamos interessante observar o quanto é "normal"a acurácia da classificação variar em função de diferentes conjuntos de dados simulados (dentro do mesmo formato) para cada método testado. Para realizar essa avaliação utilizamos os mesmos 8 conjuntos de dados apresentados, sem ruído no

	10%/90%	25%/75%	50%/50%	75%/25%	90%/10%
LORC	0.71	0.807	0.783	0.788	0.697
LORCy	0.806	0.929	0.861	0.854	0.751
Random LORC	0.642	0.721	0.788	0.747	0.682
Random LORCy	0.729	0.897	0.832	0.847	0.751
Reg. Logística	0.43	0.393	0.562	0.688	0.596
CART	0.524	0.788	0.807	0.792	0.686
Flor. Aleatórias	0.614	0.891	0.843	0.855	0.717
SVM	0.786	0.93	0.864	0.874	0.768
kNN	0.753	0.899	0.841	0.839	0.738
Média	0.666	0.8061	0.7979	**0.8093**	0.7096

Tabela 5.7 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de elementos em cada classe de rótulos (o primeiro valor é o percentual de dados com rótulo 0 e o último é o percentual de dados com rótulo 1) no conjunto de treinamento, mantendo o conjunto de teste fixo. Resultado para o Conjunto de Dados 5

	10%/90%	25%/75%	50%/50%	75%/25%	90%/10%
LORC	0.799	0.729	0.796	0.686	0.491
LORCy	0.824	0.831	0.876	0.793	0.546
Random LORC	0.765	0.712	0.819	0.509	0.543
Random LORCy	0.805	0.834	0.65	0.571	0.499
Reg. Logística	0.585	0.767	0.78	0.774	0.792
CART	0.957	0.981	0.981	0.975	0.954
Flor. Aleatórias	0.965	0.988	0.992	0.985	0.974
SVM	0.773	0.614	0.683	0.545	0.70
kNN	0.834	0.875	0.784	0.545	0.499
Média	0.8119	0.8146	**0.8179**	0.7092	0.6664

Tabela 5.8 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de elementos em cada classe de rótulos (o primeiro valor é o percentual de dados com rótulo 0 e o último é o percentual de dados com rótulo 1) no conjunto de treinamento, mantendo o conjunto de teste fixo. Resultado para o Conjunto de Dados 6

rótulo. Seguindo os resultados das seções anteriores, os conjuntos de dados de treinamento dos modelos foram gerados com 200 elementos, sendo que o percentual de rótulos 0's e 1's em cada um deles segue os resultados da Tabela 5.10. Segundo cada um dos 8 desenhos utilizados, foram gerados aleatoriamente 100 conjuntos de dados de treinamento e, utilizando um mesmo conjunto de dados de teste (composto por 100 elementos), os métodos de classificação implementados foram aplicados a cada um deles. Com os 100 valores dos percentuais de acertos de classificação para cada método, foi possível calcular os desvios-padrão em relação à média. Os valores obtidos estão apresentados na Tabela 5.11.

Observando os resultados obtidos para os desvios-padrão de cada método em cada conjunto de dados, queremos estabelecer um único valor que pode ser considerado razoável para o desvio que, em média, é comum para métodos com mesmo desempenho (no caso, para o mesmo método) aplicado em conjuntos de dados semelhantes. Dessa forma, calculamos a média dos

	10%/90%	25%/75%	50%/50%	75%/25%	90%/10%
LORC	1	0.965	0.982	0.975	0.937
LORCy	1	1	1	1	1
Random LORC	0.936	0.946	0.983	0.826	0.638
Random LORCy	1	1	1	1	1
Reg. Logística	0.7	0.7	0.6	0.6	0.6
CART	0.932	0.952	0.965	0.999	0.997
Flor. Aleatórias	0.979	0.994	0.997	1	0988
SVM	1	1	1	1	1
kNN	1	1	1	1	1
Média	0.9497	**0.9508**	0.9474	0.9333	0.9067

Tabela 5.9 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de elementos em cada classe de rótulos (o primeiro valor é o percentual de dados com rótulo 0 e o último é o percentual de dados com rótulo 1) no conjunto de treinamento, mantendo o conjunto de teste fixo. Resultado para o Conjunto de Dados 7

Conjunto	**Número de Elementos (treinamento)**	**Percentual de 0's**	**Percentual de 1's**
C1	200	50%	50%
C2	200	75%	25%
C3	200	75%	25%
C4	200	50%	50%
C5	200	75%	25%
C6	200	50%	50%
C7	200	25%	75%
C8	200	50%	50%

Tabela 5.10 Resumo dos conjuntos de dados simulados utilizados para avaliação dos métodos de classificação supervisionada

	C1	C2	C3	C4	C5	C6	C7	C8
LORC	0.055	0.025	0.035	0.061	0.041	0.047	0	0
LORCy	0.008	0.014	0.021	0.025	0.033	0.048	0	0
Random LORC	0.017	0.02	0.025	0.049	0.039	0.05	0	0.076
Random LORCy	0.013	0.013	0.029	0.029	0.034	0.045	0	0.012
Reg. Logística	0.062	0	0.037	0.065	0.029	0.043	0	0
CART	0.046	0.058	0.036	0.045	0.055	0.05	0.183	0
Flor. Aleatórias	0.015	0.037	0.018	0.029	0.039	0.043	0.022	0
SVM	0.006	0.026	0.02	0.025	0.03	0	0	0
kNN	0.013	0.015	0.032	0.031	0.034	0	0	0.026

Tabela 5.11 Desvio-padrão dos resultados em percentuais de acertos de 100 aplicações dos métodos de classificação em conjuntos de dados de treinamento distintos, dentro de cada desenho de conjunto proposto

desvios-padrão apresentados, obtendo o valor para o desvio médio que consideraremos daqui em diante: 0.028. Portanto, ao compararmos métodos diferentes, se o resultado em percentual

de acertos na classificação variar até este valor (0.028 ou 2.8%), consideraremos que eles tiveram o mesmo desempenho. Dessa forma, pretendemos contornar possíveis conclusões erradas resultantes de pequenas variações ocorridas nos dados.

5.3.2.4 Tempo de Processamento

Esta seção tem o objetivo de avaliar o tempo gasto por cada um dos algoritmos nos conjuntos de dados simulados em função do número de instâncias no conjunto de treinamento e no conjunto de teste. Os tempos de processamento foram obtidos considerando a etapa de obtençao dos parâmetros (para os métodos que passam por esta etapa), o ajuste do modelo e a classificação das instâncias do conjunto de teste. Como os conjuntos de dados são todos semelhantes em relação ao número de classes e ao número de atributos (exceto o Conjunto C6), apresentamos nas Tabelas 5.12 e 5.13 apenas os resultados obtidos para o Conjunto C1. Os resultados podem ser estendidos para qualquer conjunto de dados.

	100	500	1000	2000
LORC	2.31	6.94	11.81	21.17
LORCy	2.47	6.37	11.75	21.36
Reg. Logística	0.02	0.0	0.02	0.0
CART	0.0	0.01	0.02	0.0
Flor. Aleatórias	0.1	0.07	0.49	0.15
SVM	1.5	0.92	1.1	1.07
kNN	0.28	0.36	0.49	0.67

Tabela 5.12 Tempo de processamento (em segundos) dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de teste do modelo, mantendo o tamanho do conjunto de treinamento fixo (200 elementos).

	100	200	300	400	500	1000	1500	2000
LORC	1.12	2.4	4.32	7.22	11.13	66.42	194.55	453.53
LORCy	1.06	2.31	4.16	7.4	10.86	62.11	210.2	438.49
Reg. Logística	0.02	0.02	0.01	0.0	0.00	0.01	0.02	0.02
CART	0.01	0.0	0.01	0.01	0.01	0.01	0.02	0.04
Flor. Aleatórias	0.06	0.1	0.11	0.15	0.21	0.49	1.26	0.01
SVM	0.65	1.5	2.86	4.68	7.05	23.51	25.5	88.65
kNN	0.28	0.28	0.33	0.38	0.44	0.69	0.87	1.26

Tabela 5.13 Tempo de processamento (em segundos) dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo (100 elementos).

Podemos observar que o tempo de processamento da metodologia desenvolvida é bem maior que o dos demais métodos e que ela cresce a medida que o conjunto de dados cresce, es-

	100	500	1000	2000
LORC	2.31	6.94	11.81	21.17
LORCy	2.47	6.37	11.75	21.36
Reg. Logística	0.02	0.0	0.02	0.0
CART	0.0	0.01	0.02	0.0
Flor. Aleatórias	0.1	0.07	0.49	0.15
SVM	1.5	0.92	1.1	1.07
kNN	0.28	0.36	0.49	0.67

Tabela 5.14 Tempo de processamento (em segundos) dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de teste do modelo, mantendo o tamanho do conjunto de treinamento fixo (200 elementos).

pecialmente o de treinamento. Os resultados para o LORC e o LORCy são muito semelhantes, portanto falaremos apenas do LORC, considerando ambos.

Na Tabela 5.12 temos os valores para diferentes tamanhos do conjunto de dados de teste. Podemos perceber que o LORC aumenta o tempo de processamento em função do aumento no número de elementos do conjunto de teste de forma aproximadamente linear. Na Tabela 5.13 temos os valores para diferentes tamanhos do conjunto de dados de treinamento. Nesse caso, o aumento do tempo de processamento em função do número de elementos não aparenta ser linear. Ele parece aumentar de forma bastante rápida. Por isso, ajustamos uma função polinomial para analisar esse aumento no caso do conjunto de teste ter 100 elementos ($tempo = 0.12 * (\frac{np}{100})^2.75$, onde np é o número de elementos do conjunto de dados de treinamento) e observamos que ele é mais rápido que o crescimento quadrático. Este é um resultado ruim para este tipo de algoritmo que poderá ser utilizado para conjuntos de dados muito grandes. Para os tamanhos testados, os tempos são mais altos que os dos outros métodos, mas são tempos viáveis. O problema maior ocorre quando olharmos para o Random LORC e o Random LORC, cujo tempo de processamento é igual ao do LORC multiplicado pelo número de amostras Bootstrap utilizadas, já que o procedimento é todo repetido para cada uma. Portanto, o tempo de processamento dos métodos desenvolvidos neste trabalho, principalmente Random LORC e Random LORCy, são um ponto que precisa ser melhorado. Isso pode ser feito com algumas melhorias no código, que vão desde o armazenamento das distâncias entre as intâncias para que ela não precise ser recalculada em todas as repetições do algoritmo para um mesmo conjunto de dados, até a paralelização do código para que ele rode de maneira mais eficiente.

5.3.2.5 Resultados

Esta seção teve como pricipal objetivo definir valores a serem utilizados para os testes que serão apresentados na próxima seção. Dessa forma, pretendemos ter comparações nas quais nenhum método é privilegiado em relação ao outro, sendo juntos nas avaliações. Foram estabelecidos os seguintes valores:

- O número de instâncias nos conjuntos de dados de treinamento dos modelos será igual a 200.

- O percentual de instâncias em cada classe de rótulo nos conjuntos de dados de treinamento dos modelos, de acordo com o cenário avaliado, será dado pela Tabela 5.10

- O valor de desvio considerado para concluirmos que os métodos são igualmente eficientes é de 0.028.

É importante lembrar que até agora não avaliamos diretamente a eficiência de cada método nos conjuntos de dados sem ruído no rótulo para compará-los. Esta avaliação será apresentada juntamente com os resultados das aplicações dos métodos aplicados aos conjuntos com ruído no rótulo, Seção 5.4.

5.4 Conjuntos de Dados com Ruído no Rótulo

Neste seção exploraremos os conjuntos de dados simulados introduzindo diferentes tipos e intensidades de ruído no rótulo. Nosso objetivo é observar a robustez de cada uma das variações da metodologia LORC quando o conjunto de dados apresenta esse tipo de problema, comparando com os demais algoritmos de classificação supervisionada considerados neste trabalho. A seguir estão descritos os conjuntos de dados simulados, com as modificações que foram realizadas para as análises desta seção, e os testes realizados, seguidos pelos resultados observados e alguns comentários a respeito.

5.4.1 Os Conjuntos de Dados Simulados com Ruído no Rótulo

Os conjuntos de dados utilizados nas análises desta seção têm como base os mesmos 8 conjuntos de dados apresentados na Seção 5.3.1. Porém, em cada etapa da análise, os rótulos de algumas instâncias dos conjuntos de dados serão trocados propositalmente, com o objetivo de introduzir ruído nos rótulos. Para exemplificar a configuração dos pontos com ruído no rótulo em algum dos conjuntos de dados simulados, a Figura 5.4 mostra o Conjunto de Dados 1 com os diferentes tipos de ruído (NCAR, NAR e NNAR).

Com base nas avaliações realizadas na seção 5.3.2.1, o número de elementos nos conjuntos de dados de treinamento e de teste foram estabelecidos como 200 e 100, respectivamente. O percentual de elementos em cada classe de rótulo foi estabelecido com base na seção 5.3.2.2, de acordo com a Tabela 5.10. Cada um dos 9 algoritmos de classificação foi executado 10 vezes, sendo que em cada uma dessas execuções os conjuntos de dados de treinamento e de teste do modelo foram redefinidos. Em cada uma das 10 execuções, foram introduzidas trocas de rótulos de diferentes tipos e em diferentes intensidades, com o objetivo de analisar o desempenho dos métodos nos diferentes tipos de ruído (NAR, NCAR e NNAR).

5.4.2 Aplicações e Resultados

Os testes serão apresentados de acordo com o tipo de ruído no rótulo introduzido, afim de analisar separadamente o efeito de cada um deles no desempenho dos métodos de classificação.

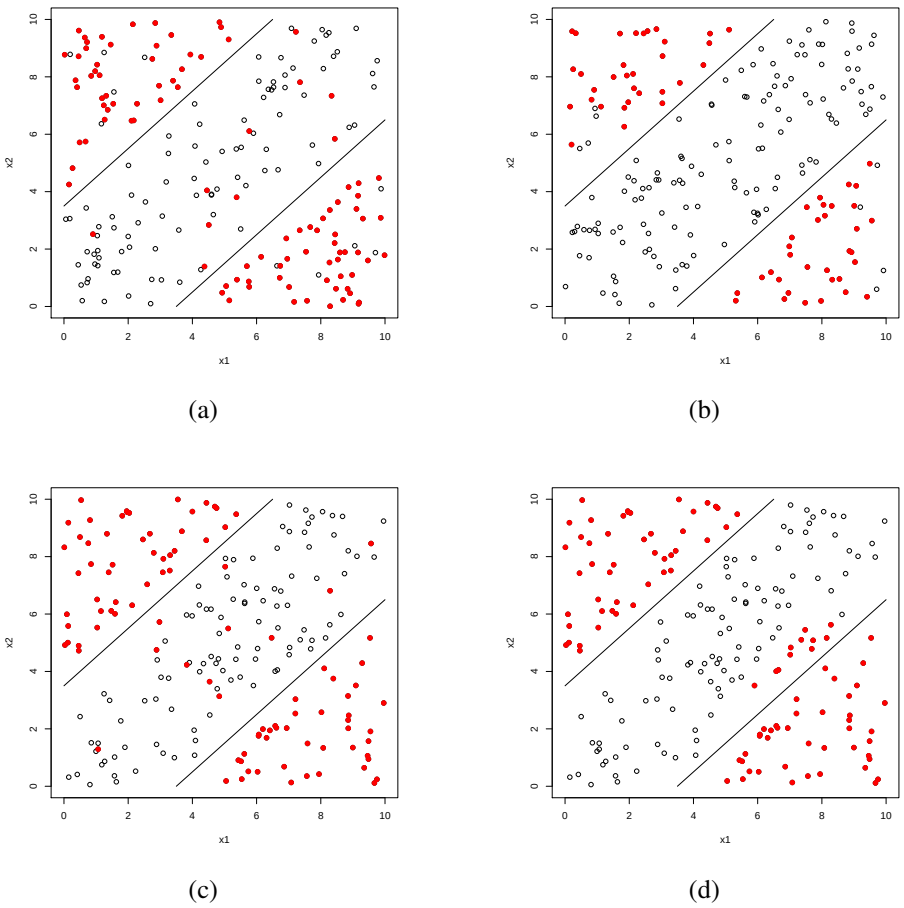


Figura 5.4 Exemplos de configurações de pontos do cenário 1 com ruído no rótulo. Em 5.4(a), o ruído é do tipo NCAR, com troca de rótulo em 10% dos pontos de cada classe. Em 5.4(b) e 5.4(c), o ruído é do tipo NAR, sendo que na primeira foram trocados os rótulos de 10% dos pontos com rótulo original 1 e no segundo, 10% dos rótulos de pontos com rótulo original 0. Em 5.4(d), o ruído é do tipo NNAR, com troca de rótulo em 10% dos pontos com rótulo original 0, porém concentrada em uma região do espaço de atributos próxima a um grupo de instâncias com rótulo 1.

5.4.2.1 Ruído do Tipo NCAR

Este tipo de ruído é o que menos afeta o desempenho dos métodos de classificação. Portanto, esperamos que nos conjuntos de dados com ruído do tipo NCAR, os métodos que têm bom desempenho quando não há ruído no rótulo também se sobressaiam, com seu percentual de acertos não sendo muito afetado. Devido à metodologia desenvolvida no LORC, esperamos que ele (e suas variações) apresente também um bom desempenho nestes conjuntos de dados.

Para introduzir o ruído NCAR nas variáveis, geramos o conjunto de dados de treinamento e, posteriormente, sorteamos aleatóriamente $x\%$ dos pontos de cada *cluster* para terem seus rótulos alterados, com $x \in 0\%, 5\%, 10\%, 15\%, 20\%, 25\%, 30\%, 35\%, 40\%$. Como estamos tratando

de ruído do tipo NCAR, cada *cluster* presente no conjunto de dados original teve o mesmo percentual de rótulos trocados.Por exemplo, se o conjunto de dados de treinamento foi construído segundo o cenário 1, ele é originalmente composto por 3 *clusters*, 2 deles com elementos cujo rótulo é igual a 1 e um com elementos cujo rótulo é igual a 0. Nesse caso, cada um dos 3 *clusters* terá *x*% de elementos cujos rótulos serão trocados.

Os resultados obtidos em percentual médio de acertos nas 10 simulações estão exibidos nas Tabelas a seguir. Os valores em negrito em cada coluna (cada percentual de troca de rótulo) representam os maiores valores de classificação, ou seja, o método que obteve melhor desempenho médio para tal percentual de rótulos trocados no conjunto de dados. Considerando o desvio-padrão médio apresentado na Seção 5.3.2.3, as células coloridas em cada coluna correspondem aos valores cuja diferença para o maior daquela coluna (em negrito) é de até 0.028. Ou seja, para cada coluna, as linhas cujas células foram coloridas representam os métodos com melhor desempenho médio na classificação, segundo o percentual de acertos.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.91	0.913	0.912	0.888	0.87	0.826	0.829	0.735	0.71	0.692
LORCy	0.982	0.941	0.892	0.866	0.835	0.791	0.766	0.684	0.671	0.687
Random LORC	0.954	0.962	0.943	0.939	0.931	0.897	0.896	0.827	0.763	0.772
Random LORCy	0.976	0.942	0.896	0.884	0.84	0.781	0.742	0.693	0.681	0.70
Reg. Logística	0.508	0.508	0.508	0.532	0.459	0.536	0.529	0.494	0.502	0.424
CART	0.871	0.874	0.849	0.859	0.853	0.819	0.80	0.758	0.723	0.686
Flor. Aleatórias	0.976	0.969	0.948	0.922	0.902	0.857	0.825	0.79	0.725	0.713
SVM	**0.993**	**0.977**	**0.969**	**0.966**	**0.939**	**0.921**	**0.911**	**0.845**	0.788	**0.808**
kNN	0.977	0.97	0.957	0.942	0.914	0.916	0.907	0.828	**0.821**	0.80

Tabela 5.15 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 1, com diferentes percentuais de troca de rótulo do tipo NCAR introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 1, quando não há ruído no rótulo (coluna 0%) podemos observar que os métodos com melhor percetual de acertos na classificação foram LORCy, Random LORCy, Florestas Aleatórias, SVM e kNN.

Quando começamos a introduzir ruído no rótulo, entre as variações da metodologia LORC, a que obteve melhor desempenho foi o Random LORC, conforme esperado. Seus resultados estão entre os melhores quando o percentual de rótulos trocados é de até 35%. Além do Random LORC, o Random Forest também esteve entre os melhores para os percetuais de troca de rótulo de 5% e 10%. Já o SVM esteve entre os melhores para quase todos os testes (exceto para 40%). O kNN obteve resultado entre os melhores para todos os percentuais de troca de rótulo analisados.

Portanto, para o conjunto de dados C1, o Random LORC pode ser considerado muito bom quando o percentual de troca de rótulos é menor que 40%, mas o kNN, que foi tão bom quanto ele dentro destes percentuais, também obteve melhores desempenhos quando há 40% ou mais dos rótulos trocados. O SVM teve o desempenho bem similar ao Random LORC.

Para o Cenário 2, quando não há troca de rótulos, todos os métodos, com exceção da Regressão Logística e do CART, apresentaram percentuais de acertos em torno de 100%, sendo

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.989	0.978	0.951	0.984	0.921	0.93	0.919	0.876	0.804	0.721
LORCy	**1**	0.929	0.905	0.852	0.817	0.751	0.764	0.734	0.708	0.623
Random LORC	0.996	0.986	0.978	0.981	0.945	0.955	0.939	0.891	0.858	0.759
Random LORCy	0.997	0.95	0.935	0.889	0.832	0.785	0.795	0.749	0.722	0.63
Reg. Logística	0.519	0.519	0.519	0.519	0.519	0.519	0.519	0.503	0.446	0.466
CART	0.958	0.962	0.963	0.963	0.929	0.861	0.828	0.817	0.78	0.729
Flor. Aleatórias	0.996	0.986	0.976	0.953	0.922	0.886	0.869	0.824	0.775	0.719
SVM	0.998	**0.996**	**0.996**	0.977	**0.962**	**0.965**	**0.967**	**0.971**	**0.886**	0.80
kNN	0.999	0.991	0.994	**0.992**	0.959	0.962	0.966	0.964	0.878	**0.836**

Tabela 5.16 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 2, com diferentes percentuais de troca de rótulo do tipo NCAR introduzidos no conjunto de treinamento do algoritmo.

considerados igualmente eficientes para este caso.

Ao introduzir ruído no rótulo, as variações do LORC que utilizam o rótulo na primeira etapa do método (construção da AGM) têm seu desempenho mais afetado que as outras variações, passando a não figurar mais entre os melhores. LORC e Florestas Aleatórias apresentam-se entre os melhores para os percentuais mais baixos de trocas de rótulos (LORC para 5% e 15% e Florestas Aleatórias para 5% e 10%), porém para percentuais maiores já passam a não aparecer entre os melhores. Dentre as variações do LORC, o Random LORC é o que se mostra mais robusto no Cenário 2 em relação a troca de rótulo, já que para percentuais de troca de até 40% das instâncias ele está sempre entre os métodos de melhor desempenho (exceto para o percentual de 35%), juntamente com o SVM e o kNN. O kNN é o único que permanece como melhor quando o percentual de troca é de 45%.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.858	0.888	0.887	0.861	0.794	0.801	0.815	0.819	0.699	0.603
LORCy	0.946	0.911	0.886	0.839	0.811	0.811	0.754	0.765	0.728	0.683
Random LORC	0.891	0.891	0.898	0.865	0.866	**0.84**	**0.823**	**0.823**	**0.762**	0.693
Random LORCy	0.922	0.874	0.853	0.831	0.783	0.762	0.731	0.695	0.644	0.614
Reg. Logística	0.34	0.335	0.342	0.351	0.366	0.351	0.36	0.342	0.366	0.342
CART	0.957	0.94	**0.94**	**0.952**	**0.886**	0.818	0.772	0.777	0.704	0.592
Flor. Aleatórias	**0.971**	**0.947**	0.939	0.891	0.855	0.804	0.799	0.759	0.73	0.611
SVM	0.94	0.905	0.906	0.862	0.835	0.794	0.763	0.808	0.739	0.649
kNN	0.925	0.891	0.893	0.87	0.828	0.811	0.755	0.816	0.761	**0.72**

Tabela 5.17 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 3, com diferentes percentuais de troca de rótulo do tipo NCAR introduzidos no conjunto de treinamento do algoritmo.

O Cenário 3 foi desenhado com objetivo de proporcionar um bom desempenho do CART e do Random Forest, conforme comentado anteriormente. Este fato realmente ocorreu, tanto para os conjuntos de dados sem ruído no rótulo quanto para os percentuais mais baixos deste ruído. Porém ao aumentar a quantidade de instâncias com rótulos trocados no conjunto de

dados de treinamento, os dois métodos não se mostraram muito robustos (principalmente o Random Forest), tendo o desempenho pior que o de outros métodos.

Ao considerar o conjunto de dados sem ruído no rótulo, podemos observar que além do CART e do Random Forest, o LORCy também está entre os melhores. Para os conjuntos de dados com 5% e 10% de rótulos trocados, o CART e o Random Forest foram os melhores e para 15% apenas o CART. A partir de 20% de rótulos trocados, o Random LORC passa a estar sempre entre os métodos de melhor desempenho na classificação, juntamente com o CART para os percentuais de 20% e 25%, com o Random Forest para 30% com o SVM e o kNN para 35% e 40% e apenas com o kNN para 45%.

Mais uma vez uma das variações do LORC (o LORCy, especificamente) obteve ótimo desempenho quando não há ruído no rótulo dos pontos que compõem o conjunto de dados de treinamento, mesmo neste conjunto de dados construído para ser adequado ao CART e ao Random Forest, tendo seu percentual de acertos de classificação comparável ao destes. Ao introduzir ruído no rótulo, observarmos que o CART se mostrou mais robusto que o Random Forest, mas ambos tiveram seu desempenho piorado em relação a outros métodos a medida que o ruído foi aumentando. Observamos ainda que a partir do percentual de 20% de rótulos trocados, o Random LORC foi o único método que teve seu desempenho entre os mélhores para todos os percentuais de troca de rótulo, novamente se mostrando um método bastante robusto.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.825	0.79	0.791	0.771	0.748	0.668	0.682	0.67	0.63	0.594
LORCy	0.899	0.844	0.816	0.757	0.716	0.696	0.647	0.658	0.589	0.594
Random LORC	0.854	0.849	0.828	0.796	0.785	0.706	0.685	0.715	0.662	0.623
Random LORCy	0.915	0.888	0.868	0.837	0.768	0.766	0.74	0.713	0.663	0.585
Reg. Logística	0.562	0.536	0.557	0.524	0.527	0.51	0.525	0.531	0.543	0.48
CART	0.836	0.825	0.829	0.799	0.778	0.79	0.739	0.674	0.632	0.632
Flor. Aleatórias	0.901	0.897	0.872	**0.863**	0.78	0.783	**0.766**	0.727	0.686	**0.674**
SVM	**0.922**	**0.913**	**0.893**	0.861	**0.834**	**0.792**	0.755	**0.782**	0.72	0.631
kNN	0.898	0.894	0.884	0.856	0.815	0.787	0.728	0.719	**0.729**	0.665

Tabela 5.18 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 4, com diferentes percentuais de troca de rótulo do tipo NCAR introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 4, ao considerar os conjuntos de dados sem ruído no rótulo, os métodos LORCy, Random LORCy, Florestas Aleatórias, SVM e kNN foram os que apresentaram maior percentual de acertos na classificação. Ao introduzir ruído no rótulo, o Random LORCy foi a variação da metodologia LORC que apresentou melhor desempenho, figurando entre os melhores métodos para quase todos os percentuais de ruído no rótulo até 30% (exceto para o percentual de 20%). O Random Forest mostrou desempenho bem semelhante ao Random LORCy, se destacando para os mesmos percentuais de ruído (apenas no percentual 45% o este método foi melhor que o Random LORCy). O SVM e o kNN aparecem entre os métodos com melhor desempenho para quase todos os percetuais de troca de rótulo, exceto 30% e 35% para o kNN e 45% para o SVM.

Para o Cenário 5, ao considerar os conjuntos de dados sem troca de rótulo, as variações

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.792	0.79	0.784	0.767	**0.766**	0.741	0.688	0.689	0.661	0.657
LORCy	**0.837**	**0.827**	0.795	0.764	0.718	0.711	0.673	0.662	0.662	0.634
Random LORC	0.794	0.803	0.799	0.777	0.74	0.739	0.711	0.692	0.715	0.644
Random LORCy	0.811	0.8	0.77	0.744	0.712	0.702	0.646	0.608	0.617	0.61
Reg. Logística	0.566	0.573	0.576	0.581	0.576	0.589	0.58	0.567	0.559	0.569
CART	0.772	0.764	0.76	0.774	0.744	0.713	0.685	0.668	0.645	0.599
Flor. Aleatórias	0.831	0.816	**0.818**	0.79	0.735	0.713	0.697	0.678	0.643	0.63
SVM	0.827	0.825	0.815	**0.795**	0.758	**0.76**	**0.737**	**0.714**	**0.726**	0.65
kNN	0.823	0.812	0.799	0.793	0.758	0.728	0.705	**0.714**	0.708	**0.663**

Tabela 5.19 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 5, com diferentes percentuais de troca de rótulo do tipo NCAR introduzidos no conjunto de treinamento do algoritmo.

do LORC que utilizam o rótulo na primeira etapa do método (construção da AGM), ou seja LORCy e Random LORCy, apresentaram-se como melhores em relação à acurácia na classificação, juntamente com Florestas Aleatórias, SVM e kNN.

Ao considearmos conjuntos de dados com ruído no rótulo no Cenário 5, o LORCy e o Random LORCy apresentam-se entre os melhores apenas para percentuais baixos de rótulos trocados (LORCy até 10% e Random LORCy até 5%). Florestas Aleatórias também apresenta-se entre os melhores para baixos percentuais de troca de rótulo (até 15%). Podemos observar que o Random LORC é a variação da metodologia LORC que apresenta melhor desempenho geral, aparecendo como um dos melhores métodos, juntamente com o SVM, para todos os percentuais de rótulos trocados. O LORC se mostra eficiente principalmente se o percentual de troca de rótulo não for muito baixo, estando entre os melhores métodos para os percentuais entre 15% e 25%, 35% e 45%. O kNN também apresenta-se entre os melhores para quase todos os percentuais de troca, exceto para 25% e 30%).

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.76	0.776	0.766	0.743	0.723	0.692	0.663	0.646	0.644	0.569
LORCy	0.872	0.825	0.815	0.751	0.726	0.676	0.693	0.617	0.636	0.594
Random LORC	0.829	0.814	0.806	0.75	0.80	0.721	0.764	0.706	0.666	0.642
Random LORCy	0.838	0.868	0.828	0.775	0.748	0.71	0.708	0.663	0.648	0.642
Reg. Logística	0.958	0.938	0.942	0.925	0.918	0.901	0.874	0.845	0.777	0.743
CART	0.972	0.965	0.966	0.925	0.846	0.826	0.794	0.689	0.669	0.607
Flor. Aleatórias	**0.981**	**0.986**	**0.983**	**0.978**	**0.969**	**0.956**	0.94	0.906	0.859	0.754
SVM	0.969	0.968	0.969	0.957	0.956	0.938	0.88	0.801	0.698	0.562
kNN	0.966	0.967	0.967	0.964	0.956	0.954	**0.951**	**0.931**	**0.901**	**0.852**

Tabela 5.20 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 6, com diferentes percentuais de troca de rótulo do tipo NCAR introduzidos no conjunto de treinamento do algoritmo.

O Cenário 6 apresenta muitos atributos que correspondem a ruído (18 das 20 que formam cada instância), portanto estamos buscando avaliar também outro tipo de robustez dos méto-

dos de classificação, a robustez em relação à introdução de variáveis de ruído no conjunto de dados analisado. A partir da Tabela 5.20, podemos observar que nenhuma das variação da metodologia LORC conseguiu bons resultados para este cenário.

Ao observar os conjuntos de dados sem ruído no rótulo, todos os demais métodos (Regressão Logística, CART, Florestas Aleatórias, SVM e kNN) tiveram bons resultados. Ao analisar conjuntos de dados com ruído no rótulo, o kNN foi o método que se mostrou mais robusto, com os resultados entre os melhores para todos os percentuais de ruído introduzidos. O Florestas Aleatórias também se mostrou robusto, estando com o percentual de classificações corretas entre os maiores para percentuais de ruído no rótulo de até 35%. Finalmente, o SVM esteve entre os melhores para percentuais de ruído no rótulo de até 25% e o CART até 10%.

Pelos resultados obtidos para o Cenário 6, podemos supor que a metodologia desenvolvida neste trabalho não é uma boa opção para tratar de conjuntos de dados com muitas variáveis de ruído, já que seu desempenho se mostrou aquém dos demais métodos especificamente para este cenário.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.995	0.972	0.966	0.954	0.95	0.931	0.948	0.890	0.864	0.830
LORCy	**1**	0.941	0.862	0.785	0.765	0.689	0.684	0.574	0.561	0.544
Random LORC	**1**	0.973	0.989	0.983	0.978	0.972	0.962	0.937	0.871	0.868
Random LORCy	**1**	0.974	0.934	0.872	0.837	0.801	0.785	0.726	0.67	0.676
Reg. Logística	0.70	0.70	0.70	0.70	0.70	0.727	0.702	0.737	0.776	0.727
CART	0.985	0.974	0.975	0.979	0.962	0.957	0.917	0.89	0.857	0.78
Flor. Aleatórias	0.999	0.986	0.984	0.968	0.939	0.895	0.887	0.808	0.751	0.724
SVM	**1**	**1**	**1**	**1**	**0.999**	**0.998**	**0.998**	**0.998**	**0.944**	**0.97**
kNN	**1**	0.999	0.997	**1**	0.979	0.995	0.996	0.987	0.968	0.92

Tabela 5.21 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 7, com diferentes percentuais de troca de rótulo do tipo NCAR introduzidos no conjunto de treinamento do algoritmo.

O Cenário 7 sem ruído no rótulo é formado por 4 *clusters* rotulados compactos, de forma que o resultado observado era o esperado: resultado ótimo de todas as variações do LORC, conforme podemos verificar na primeira coluna da Tabela 5.21. Além dos 4, CART, Florestas Aleatórias, SVM e kNN também tiveram desempenho ótimo, acertando a classificação de aproximadamente todos os pontos.

A medida que as trocas de rótulo vão sendo introduzidas nos conjuntos de dados, todas as variações do LORC, assim como o CART e o Florestas Aleatórias, passam a apresentar piores desempenhos a partir de determinados percentuais. O LORC e Random LORCy só ficam entre os melhores para o percentual de troca de rótulo de 5%. CART e Florestas Aleatórias se mantém entre os melhores resultados até os percentuais 15% e 10%, respectivamente. A variação do LORC que se mostra mais robusta neste caso é o Random LORC, cujo resultado está entre os melhores para todos os percentuais de troca de rótulo menores que 30%. Mesmo a partir deste percentual, a acurácia do método fica sempre acima de 87%, o que implica bons resultados, apesar de o SVM e o kNN serem melhores neste caso em que o percentual de troca de rótulo é alto. Com relação a metodologia LORC, essa perda de desempenho que não seria teoricamente

esperada se deve ao fato de que o número de instâncias em cada *cluster* é relativamente pequeno e o ruído acaba não sendo introduzido de forma completamente aleatória, como era suposto nas demosntrações de eficiência desenvolvidas na Seção 4.1. Desta forma, o SVM e kNN se destacam, principalmente o SVM, que se mostra o único melhor para o percentual mais alto de toca de rótulo (45%).

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.499	0.491	0.499	0.528	0.51	0.523	0.481	0.49	0.518	0.506
LORCy	**0.996**	0.954	0.893	0.843	0.826	0.791	0.737	0.682	0.665	0.624
Random LORC	0.544	0.528	0.522	0.529	0.53	0.504	0.505	0.502	0.522	0.52
Random LORCy	0.982	0.968	0.902	0.867	0.832	0.801	0.757	0.703	0.658	0.626
Reg. Logística	0.994	**0.994**	**0.996**	**0.994**	**0.995**	**0.996**	**0.994**	**0.996**	**0.996**	**0.993**
CART	0.502	0.796	0.847	0.866	0.806	0.816	0.668	0.686	0.703	0.691
Flor. Aleatórias	0.993	0.986	0.984	0.966	0.942	0.933	0.886	0.856	0.821	0.769
SVM	0.981	0.761	0.768	0.762	0.749	0.719	0.803	0.637	0.716	0.625
kNN	0.484	0.497	0.491	0.506	0.496	0.479	0.512	0.528	0.52	0.479

Tabela 5.22 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 8, com diferentes percentuais de troca de rótulo do tipo NCAR introduzidos no conjunto de treinamento do algoritmo.

O Cenário 8 foi criado com objetivo de representar um cenário no qual o LORC (e o Random LORC) apresentam grandes dificuldades, ou seja, cenário no qual o método não seria adequado. Já o LORCy (e o Random LORCy) apresentam modificações metodológicas capazes de contornar o problema, supostamente podendo apresentar bons resultados para este cenário. O Cenário 8 também é bem propício a um bom desempenho da Regressão Logística, o que pode ser verificado nos resultados apresentados na Tabela 5.22, onde podemos perceber que este método aparece entre os melhores em relação ao percentual de acertos na classificação tanto para os conjuntos de dados sem ruído no rótulo quanto para os conjuntos com todos percentuais de troca de rótulo (desde 5% até 40%).

Quando não há rótulos trocados nos conjuntos de dados, além da Regressão Logística, os métodos LORCy, Random LORCy, Florestas Aleatórias e SVM também estão entre os melhores. Conforme previsto anteriormente, as variações do LORC que utilizam o rótulo na primeira etapa do método (para a construção da AGM) foram capazes de contornar o problema encontrado pelas variações que não têm essa característica, em relação a este tipo de cenário. Observe a diferença no desempenho entre elas apresentado na Tabela.

Ao introduzir ruído no rótulo, o Random LORCy teve seu desempenho entre os melhores apenas para o 5% dos rótulos trocados. A medida que o percentual de ruído no rótulo aumenta, a acurácia apresentada pelo LORCy e pelo Random LORCy se distancia cada vez da Regressão Logística, que apresentou desempenho excelente neste cenário, independente do percentual de rótulos trocados. De toda forma, estes métodos apresentam desempenhos bem melhores que o LORC, Random LORC e kNN, que não conseguem captar praticamente nenhuma informação neste cenário, obtendo sempre uma média de acertos de classificação em torno de 50%.

5.4.2.2 Ruído do Tipo NAR

Este tipo de ruído, que pode ser assimétrico entre as classes diferentes, costuma afetar bastante o desempenho dos algoritmos de classificação. Dessa forma, espera-se que a medida que o percentual de ruído vá aumentando no conjunto de dados de treinamento, o percentual de acertos dos algoritmos vá diminuindo consideravelmente. Conforme vimos nas demonstrações de eficiência do LORC, se o ruído se distribuir de maneira bem uniforme nos *clusters* em que ele estiver presente, a metodologia proposta será eficiente. Nesses casos, esperamos que seu desempenho possa superar os dos métodos tradicionais de classificação supervisionada que estão sendo comparados.

Para introduzir o ruído NAR nas variáveis, geramos normalmente o conjunto de dados de treinamento e, posteriormente, sorteamos aleatóriamente $x\%$ dos pontos de cada *cluster* composto por instâncias de somente uma das classes para terem seus rótulos alterados, com $x \in 10\%, 20\%, 30\%, 40\%$. O mesmo procedimento será realizado, trocando os rótulos da outra classe. Assim introduzimos o ruído de forma desbalanceada, de forma a haver rótulos trocados em apenas uma das classes.

Primeiramente apresentaremos os resultados para os conjuntos de dados nos quais as trocas de rótulos ocorreram apenas nas classes 0. Eles estão nas Tabelas a seguir.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.906	0.909	0.895	0.884	0.902	0.843	0.859	0.863	0.815	0.876
LORCy	0.987	**0.985**	**0.978**	**0.975**	**0.965**	**0.973**	**0.94**	**0.932**	**0.929**	**0.909**
Random LORC	0.962	0.95	0.95	0.958	0.956	0.937	0.919	0.908	0.901	0.884
Random LORCy	0.972	0.955	0.925	0.916	0.893	0.877	0.815	0.825	0.835	0.766
Reg. Logística	0.504	0.492	0.458	0.451	0.442	0.415	0.429	0.427	0.41	0.415
CART	0.90	0.866	0.90	0.897	0.85	0.865	0.818	0.77	0.798	0.769
Flor. Aleatórias	0.977	0.968	0.954	0.937	0.908	0.893	0.828	0.814	0.808	0.758
SVM	**0.995**	0.983	0.969	0.967	0.958	0.952	0.903	0.901	0.873	0.856
kNN	0.974	0.966	0.96	0.951	0.942	0.918	0.875	0.891	0.864	0.836

Tabela 5.23 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 1, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 0 para 1) introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 1, quando avaliamos conjuntos de dados sem ruído no rótulo, os métodos com maior acurácia na classificação são LORCy, Random LORCy, Florestas Aleatórias, SVM e CART.

Ao avaliar os conjuntos de dados segundo o Cenário 1 introduzindo ruído no rótulo NAR da forma descrita, O LORCy é o método que apresenta melhores resultados para todos os percentuais de ruído avaliados, desde 5% até 45%. O Random LORC também se detaca neste cenário, figurando entre os melhores em relação ao percentual de acertos para todos os percentuais de ruído maiores que 5%, exceto o 25%. O SVM está entre os melhores para percentuais de ruído de até 25%, o kNN até 20% e Forestas Aleatórias até 10%, porém eles não se mostram robustos a percentuais maiores de ruído no rótulo, quando perdem um pouco de desempenho.

Para o Cenário 2, quando não há ruído no rótulo dos elementos que compõem os conjuntos de dados de treinamento dos modelos, os melhores métodos em relação ao percentual de acertos

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.983	0.95	0.95	0.944	0.954	0.924	0.93	0.901	0.911	0.916
LORCy	0.993	**0.993**	0.976	**0.987**	**0.983**	0.965	0.956	0.945	0.936	0.934
Random LORC	0.976	0.975	0.961	0.956	0.959	0.957	0.966	0.957	**0.957**	**0.936**
Random LORCy	0.983	0.98	0.943	0.938	0.929	0.892	0.882	0.853	0.813	0.841
Reg. Logística	0.506	0.506	0.506	0.506	0.506	0.506	0.506	0.492	0.475	0.457
CART	0.962	0.942	0.943	0.976	0.918	0.903	0.908	0.86	0.82	0.813
Flor. Aleatórias	0.959	0.966	0.965	0.972	0.954	0.936	0.94	0.879	0.874	0.858
SVM	0.981	0.981	0.964	0.986	0.979	0.975	0.967	0.96	0.925	0.918
kNN	**0.995**	0.985	**0.977**	0.972	0.971	**0.977**	**0.968**	**0.963**	0.937	**0.936**

Tabela 5.24 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 2, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 0 para 1) introduzidos no conjunto de treinamento do algoritmo.

de classifcação são LORC, LORCy, Random LORC, Random LORCy, SVM e kNN.

Ao tratarmos os conjuntos de dados segundo o Cenário 2, introduzindo ruído no rótulo, os métodos LORCy e kNN ganham destaque por estarem entre os melhores para todos os percentuais de ruído no rótulo avaliados, desde 5% até 45%. Além destes, também se mostraram muito bens neste cenário os métodos Random LORC e SVM, que não ficaram com os melhores resultados apenas em um percentual de troca de rótulo (15% para o Random LORC e 40% para o SVM). Florestas Aleatórias apresentou acurácia entre as melhores para conjuntos de dados com baixos percentuais de troca de rótulo (até 15%) e LORC e CART apresentaram-se enre os melhores apenas para 2 e 1 valores, respectivamente, de percentual de troca de rótulo.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.864	0.855	0.867	0.856	0.816	0.823	0.828	0.795	0.809	0.834
LORCy	0.952	0.946	0.948	**0.95**	**0.926**	**0.912**	0.894	**0.904**	**0.88**	**0.886**
Random LORC	0.863	0.876	0.848	0.827	0.869	0.801	0.797	0.821	0.781	0.739
Random LORCy	0.935	0.893	0.895	0.857	0.843	0.81	0.809	0.762	0.761	0.72
Reg. Logística	0.356	0.348	0.352	0.343	0.345	0.343	0.347	0.358	0.346	0.345
CART	**0.976**	**0.974**	**0.977**	**0.95**	0.91	0.905	**0.897**	0.809	0.783	0.854
Flor. Aleatórias	**0.976**	**0.974**	0.948	0.921	0.881	0.847	0.826	0.791	0.741	0.712
SVM	0.938	0.935	0.926	0.885	0.839	0.808	0.815	0.753	0.746	0.713
kNN	0.933	0.922	0.916	0.898	0.882	0.844	0.862	0.795	0.799	0.785

Tabela 5.25 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 3, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 0 para 1) introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 3, sem ruído no rótulo, LORCy, CART e Florestas Aleatórias foram os métodos com melhores acurácias. É importante lembrar que este cenário é bem adequado aos métodos CART e Florestas Aleatórias, de forma a ser esperado o bom desempenho destes métodos. O LORCy consegue acompanhar este desempenho no cenário proposto, estando junto com os dois métodos entre os melhores desempenhos na classificação.

Ao introduzir o ruído NAR, trocando rótulos de pontos da classe 0, podemos observar que

os métodos CART e Florestas Aleatórias não apresentam tanta robustez quando o LORCy. O CART se mostra mais robusto que Florestas Aleatórias, mostrando-se entre os maiores valores de acurácia para percentuais de troca de rótulo de até 30% contra os de Florestas aleatórias que apareceram entre os maiores para percentuais de troca de até 10%. A frente de ambos e de todos os outros, o LORCy ficou entre os melhores para todos os percentuais de troca de rótulo avaliados, desde 5% até 45%.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.852	0.842	0.829	0.841	0.831	0.816	0.79	0.823	0.799	0.76
LORCy	0.897	0.892	0.891	0.892	0.899	0.88	**0.879**	**0.878**	**0.866**	**0.86**
Random LORC	0.875	0.879	0.886	0.865	0.875	0.853	0.833	0.823	0.819	0.827
Random LORCy	0.897	0.893	0.891	0.879	0.873	0.866	0.851	0.834	0.825	0.79
Reg. Logística	0.528	0.511	0.565	0.562	0.599	0.594	0.594	0.594	0.594	0.594
CART	0.829	0.846	0.84	0.822	0.831	0.822	0.791	0.809	0.783	0.71
Flor. Aleatórias	0.918	0.908	0.905	0.886	0.877	0.837	0.814	0.811	0.809	0.778
SVM	**0.92**	**0.913**	**0.917**	**0.906**	**0.901**	**0.893**	0.866	0.865	0.831	0.813
kNN	0.904	0.893	0.892	0.868	0.855	0.835	0.798	0.79	0.789	0.735

Tabela 5.26 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 4, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 0 para 1) introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 4, sem ruído no rótulo, os métodos LORCy, Random LORCy, Florestas Aleatórias, SVM e kNN foram os que métodos com melhores acurácias. Estes mesmos métodos permanecem sendo os melhores ao introduzir o ruído NAR, trocando rótulos de pontos da classe 0, quando o percentual de troca de rótulo é de até 10%. A partir deste percentual, o kNN deixa de estar entre os melhores resultados. Florestas Alatórias permanece entre os melhores até o percentual de 20% de troca de rótulo, o Random LORCy até 30% e o SVM até 35%. Acima deste percentual, nenhum método acompanha o LORCy com o melhor desempenho. Portanto, LORCy se mostrou o método mais robusto (e mais adequado) no cenário apresentado.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.771	0.77	0.767	0.773	0.765	0.757	0.744	0.714	0.733	0.719
LORCy	0.824	**0.827**	**0.818**	**0.817**	0.802	**0.805**	**0.805**	**0.786**	**0.772**	**0.752**
Random LORC	0.785	0.773	0.77	0.781	0.754	0.78	0.777	0.748	0.745	0.727
Random LORCy	0.814	0.79	0.797	0.768	0.743	0.756	0.723	0.714	0.659	0.632
Reg. Logística	0.556	0.548	0.534	0.544	0.536	0.518	0.498	0.484	0.475	0.458
CART	0.774	0.755	0.772	0.759	0.739	0.735	0.741	0.685	0.684	0.618
Flor. Aleatórias	**0.832**	0.818	0.805	0.789	0.777	0.765	0.736	0.707	0.649	0.643
SVM	0.824	0.807	0.816	0.803	**0.804**	0.784	0.768	0.745	0.721	0.707
kNN	0.811	0.796	0.8	0.788	0.753	0.755	0.765	0.734	0.728	0.676

Tabela 5.27 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 5, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 0 para 1) introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 5, quando os conjuntos de dados utilizados não têm ruído no rótulo, os

melhores desempenhos em relação à acurácia na classificação foram das variações da metodologia LORC que utilizam o rótulo na etapa de construção da AGM, ou seja, LORCy e Random LORCy, juntamente com Florestas Aleatórias, SVMe kNN.

Ao introduzir ruído no rótulo neste cenário, o LORCy se mostrou o mais robsuto, apresentando os melhores resultados para todos os percentuais de ruído no rótulo testados. Para percentuais mais baixos, Florestas Aleatórias e SVM também mostraram bom desempenho, estando entre os melhores para percentuais de até 20% e 25%. A partir de 25%, a variação Random LORC passa a figurar entre os melhores ao lado do LORCy, mostrando-se uma boa opção para percetuais mais altos de ruído no rótulo.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.758	0.771	0.748	0.759	0.84	0.77	**0.811**	0.801	**0.791**	0.733
LORCy	0.704	0.728	0.686	0.678	0.759	0.753	0.757	0.712	0.76	0.648
Random LORC	0.798	0.764	0.77	0.801	0.758	0.766	0.729	0.635	0.748	0.599
Random LORCy	0.691	0.688	0.679	0.845	0.745	0.739	0.572	**0.816**	0.667	0.573
Reg. Logística	0.738	0.736	0.74	0.798	0.793	**0.793**	0.693	0.792	0.723	**0.747**
CART	0.976	**0.977**	**0.976**	**0.901**	**0.86**	0.641	0.794	0.70	0.723	0.663
Flor. Aleatórias	**0.985**	0.778	0.678	0.662	0.726	0.517	0.657	0.703	0.615	0.734
SVM	0.585	0.586	0.588	0.584	0.541	0.538	0.493	0.59	0.494	0.599
kNN	0.797	0.657	0.738	0.55	0.684	0.645	0.623	0.615	0.558	0.547

Tabela 5.28 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 6, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 0 para 1) introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 6, apenas CART e Florestas Aleatórias se mostraram como os melhores na classificação de novos pontos, ao tratarmos de conjuntos de dados sem ruído no rótulo. Ao considerar os conjuntos com rótulos trocados segundo o tipo NAR, trocando apenas os pontos da classe de rótulo 0, podemos observar o CART foi o método com melhores resultados para a acurácia na classificação para percentuais de troca de rótulo de até 20%. O método LORC passa a aparecer entre os melhores para percentuais de troca de rótulo a partir de 20%, mostrando maior robustez. Alguns outros métodos apresentaram bons resultados em alguns poucos percentuais de trocas de rótulos, com destaque para a Regressão Logística que também mostrou certa robustez para percentuais altos.

O Cenário 7 apresenta um cenário no qual a maior parte dos métodos de classificação tende a apresentar bons resultados. Tanto que, quando não há ruído no rótulo, apenas Regressão Logística fica fora do grupo dos métodos de melhor desempenho, conforme podemos observar na Tabela 5.29. O percentual de acertos de classificação é de 100% ou bem próximo disso para todos os demais métodos testados.

Ao introduzir o ruído NAR, trocando rótulos da classe 0 para 1, LORCy e SVM são os métodos que apresentam os melhores desempenhos para todos os percentuais testados de troca de rótulos. Além deles, kNN e LORCy também se apresentam entre os melhores para a maioria dos percentuais de troca de rótulo (até 40% para o kNN e até 30% para o Random LORC), demonstrando bons desempenhos. Finalmente, podemos observar que LORC, CART e Florestas Aleatórias aparecem entre os melhores para percentuais de até até 15%, mostrando serem

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.998	0.976	0.974	0.977	0.958	0.944	0.931	0.94	0.90	0.932
LORCy	**1**	0.999	0.992	**0.999**	0.986	0.977	0.976	0.973	0.958	0.97
Random LORC	**1**	0.993	0.987	0.99	0.983	0.974	0.973	0.954	0.938	0.937
Random LORCy	**1**	0.989	0.98	0.968	0.964	0.943	0.925	0.924	0.902	0.914
Reg. Logística	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70
CART	0.982	0.977	0.982	0.975	0.962	0.964	0.946	0.939	0.893	0.892
Flor. Aleatórias	0.997	0.991	0.988	0.978	0.966	0.95	0.937	0.932	0.902	0.911
SVM	**1**	**1**	**1**	**0.999**	**1**	**0.993**	**1**	**0.994**	**0.984**	**0.998**
kNN	**1**	0.999	0.997	0.998	**1**	0.991	0.995	0.985	0.975	0.965

Tabela 5.29 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 7, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 0 para 1) introduzidos no conjunto de treinamento do algoritmo.

muito bons para percentuais baixos de troca de rótulo.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.515	0.52	0.517	0.518	0.52	0.517	0.515	0.512	0.516	0.515
LORCy	**0.995**	0.992	0.987	0.99	0.979	0.977	0.957	0.958	0.953	0.919
Random LORC	0.549	0.534	0.507	0.529	0.543	0.516	0.516	0.516	0.517	0.518
Random LORCy	0.988	0.974	0.949	0.94	0.905	0.883	0.89	0.844	0.851	0.816
Reg. Logística	0.994	**0.993**	**0.993**	**0.992**	**0.992**	**0.992**	**0.992**	**0.992**	**0.992**	**0.992**
CART	0.483	0.517	0.544	0.526	0.521	0.517	0.517	0.526	0.522	0.526
Flor. Aleatórias	0.994	0.985	0.991	0.977	0.968	0.961	0.94	0.932	0.916	0.896
SVM	0.975	0.753	0.738	0.728	0.715	0.694	0.695	0.676	0.667	0.665
kNN	0.523	0.527	0.513	0.519	0.517	0.517	0.517	0.519	0.517	0.517

Tabela 5.30 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 8, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 0 para 1) introduzidos no conjunto de treinamento do algoritmo.

Lembrando que o Cenário 8 foi criado com objetivo de representar um cenário no qual o LORC (e o Random LORC) apresentam grandes dificuldades, ou seja, cenário no qual o método não seria adequado. Já o LORCy (e o Random LORCy) apresenta modificações metodológicas capazes de contornar o problema, supostamente podendo apresentar bons resultados para este cenário. O Cenário 8 também é bem propício a um bom desempenho da Regressão Logística, o que foi visto na categoria anterior de testes (com ruído do tipo NCAR) e que também pode ser verificado nos resultados apresentados na Tabela 5.30, onde podemos perceber que este método aparece entre os melhores em relação ao percentual de acertos na classificação tanto para os conjuntos de dados sem ruído no rótulo quanto para os conjuntos com todos percentuais de troca de rótulo (desde 5% até 40%).

Quando não há rótulos trocados nos conjuntos de dados, além da Regressão Logística, os métodos LORCy, Random LORCy, Florestas Aleatórias e SVM também estão entre os melhores. Novamente as variações do LORC que utilizam o rótulo na primeira etapa do método (para a construção da AGM) foram capazes de contornar o problema encontrado pelas variações

que não têm essa característica, em relação a este tipo de cenário. Observe a diferença no desempenho entre elas apresentado na Tabela 5.30.

Ao introduzir ruído no rótulo, o Random LORCy teve seu desempenho entre os melhores apenas para o 5% dos rótulos trocados. Já o LORCy acompanha o desempenho da Regressão Logística para percentuais de ruído no rótulo de até 25% enquanto Florestas Aleatórias acompanha até 20%. À medida que o percentual de ruído no rótulo aumenta, a acurácia apresentada destes métodos se distancia cada vez da Regressão Logística, que apresentou desempenho excelente neste cenário, independente do percentual de rótulos trocados. De toda forma, estes métodos apresentam desempenhos bem melhores que os demais (LORC, Random LORC, CART, SVM e kNN) neste cenário.

Finalizados os resultados para trocas de rótulos nas classes 0, daqui em diante serão apresentados os resultados para os conjuntos de dados nos quais as trocas de rótulos ocorreram apenas nas classes 1. Eles estão nas Tabelas a seguir.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.90	0.911	0.916	0.902	0.894	0.911	0.896	0.887	0.88	0.83
LORCy	0.976	0.925	0.914	0.873	0.858	0.811	0.781	0.765	0.775	0.734
Random LORC	0.959	0.96	0.953	0.942	0.923	0.923	0.906	0.903	0.869	0.828
Random LORCy	0.976	0.962	0.954	0.92	0.915	0.898	0.873	0.865	0.846	0.842
Reg. Logística	0.515	0.547	0.556	0.569	0.581	0.562	0.57	0.566	0.57	0.57
CART	0.895	0.883	0.889	0.889	0.878	0.87	0.847	0.841	0.845	0.823
Flor. Aleatórias	0.966	0.966	0.958	0.936	0.925	0.901	0.881	0.865	0.859	0.825
SVM	**0.991**	**0.984**	**0.97**	**0.967**	**0.946**	**0.941**	**0.918**	**0.911**	**0.893**	**0.855**
kNN	0.968	0.96	0.961	0.951	**0.946**	0.924	0.903	0.907	0.863	0.844

Tabela 5.31 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 1, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 1 para 0) introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 1, ao observar os conjuntos de dados sem ruído no rótulo, os melhores desempenhos em relação à acurácia na classificação foram das variações da metodologia LORC que utilizam o rótulo na etapa de construção da AGM, ou seja, LORCy e Random LORCy, juntamente com Florestas Aleatórias, SVM e kNN.

Ao introduzir o ruído do tipo NAR trocando rótulos da classe 1 para 0, Random LORC e SVM foram os métodos que apresentaram os melhores desempenhos para todos os percentuais de troca de rótulo testados, mostrando-se robustos para este tipo de ruído no Cenário 1. Eles foram seguidos de perto pelo kNN, que só não apareceu entre os melhores parar o percentual de troca de 40%. Para percentuais baixos de troca de rótulo (de até 10%), Random LORCy e Florestas Aleatórias também estão entre os melhores e para percentuais mais altos (a partir de 30%), o LORC foi um dos melhores.

Para o Cenário 2, quando não há troca de rótulo nos conjuntos de dados de treinamento do modelo, todos os métodos com exceção da Regressáo Logística e do CART, apresentaram resultados entre os melhores desempenhos na acúracia da classificação de novas instâncias.

Ao introduzir ruído no rótulo, o SVM foi o método que obteve seu desempenho entre os melhores, em relação a acurácia na classificação, para todos os percentuais testados de troca de

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.994	0.989	0.97	0.992	0.967	0.935	0.946	0.921	0.855	0.881
LORCy	0.999	0.944	0.908	0.859	0.831	0.782	0.774	0.687	0.702	0.701
Random LORC	0.998	0.988	0.984	0.975	0.954	0.951	0.93	0.891	0.852	0.837
Random LORCy	0.993	0.974	0.962	0.947	0.917	0.888	0.885	0.837	0.852	0.815
Reg. Logística	0.482	0.482	0.482	0.482	0.482	0.482	0.482	0.482	0.482	0.482
CART	0.943	0.934	0.945	0.962	0.951	0.922	0.896	0.863	0.828	0.831
Flor. Aleatórias	0.988	0.985	0.985	0.977	0.944	0.93	0.903	0.881	0.826	0.828
SVM	0.999	0.995	**0.996**	**0.997**	**0.971**	**0.96**	0.947	**0.942**	**0.9**	**0.903**
kNN	**1**	**0.997**	0.966	0.985	0.948	0.949	**0.95**	0.87	0.841	0.872

Tabela 5.32 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 2, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 1 para) introduzidos no conjunto de treinamento do algoritmo.

rótulo. O LORC também se mostrou uma boa opção, apresentando-se entre os melhores para todos os percentuais, exceto para 40%. Além deles, Random LORC e kNN apresentaram-se entre os melhores para percentuai de troca de rótulo de até 30%, exceto para o percentual de 10%, no qual o kNN não esteve entre os métodos de melhor desempenho. Florestas aleatórias se mostrou também uma boa opção para baixos percentuais de troca de rótulo (até 20%).

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.847	0.829	0.863	0.862	0.835	0.884	0.858	0.864	0.885	0.86
LORCy	0.932	0.909	0.888	0.886	0.845	0.829	0.812	0.809	0.787	0.79
Random LORC	0.893	0.892	0.914	0.904	0.898	0.892	0.907	0.904	**0.928**	**0.917**
Random LORCy	0.911	0.904	0.884	0.904	0.871	0.86	0.884	0.847	0.854	0.844
Reg. Logística	0.323	0.323	0.335	0.348	0.326	0.384	0.412	0.524	0.489	0.586
CART	0.956	0.972	0.957	0.958	0.954	0.928	0.903	0.912	0.889	0.88
Flor. Aleatórias	**0.971**	**0.975**	**0.974**	**0.974**	**0.959**	**0.949**	**0.933**	**0.917**	0.908	0.897
SVM	0.939	0.93	0.94	0.935	0.923	0.918	0.932	0.899	0.889	0.916
kNN	0.926	0.897	0.897	0.902	0.885	0.889	0.888	0.884	0.883	0.875

Tabela 5.33 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 3, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 1 para 0) introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 3, que foi construído de forma a atender bem os requisitos do CART e do Florestas Aleatórias para obter bons desempenhos destes métodos, podemos observar que quando não há troca de rótulo nos conjuntos de dados utilizados, estes são os dois métodos que apresentam os melhroes desempenhos em relação à acurácia da classificação.

Ao considerar os conjuntos de dados com ruído do tipo NAR trocando os rótulos da classe 0 para 1, podemos observar, segundo a Tabela 5.33, que o método Florestas Aleatórias me mostra robusto, tendo os resultados entre os melhores para todos os percentuais de troca de rótulo testados. Já o CART, figura entre os melhores para percentuais de até 25%. A partir deste percentual, ele deixa de estar entre os melhroes, e outro método entra nesse grupo, o Random LORC, mais uma vez se mostrando uma boa opção para altos percentuais de troca de

rótulo.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.816	0.794	0.799	0.79	0.763	0.799	0.69	0.708	0.655	0.665
LORCy	0.885	0.856	0.81	0.772	0.713	0.723	0.66	0.623	0.614	0.619
Random LORC	0.847	0.843	0.823	0.796	0.765	0.784	0.731	0.702	0.652	0.658
Random LORCy	0.907	0.895	0.888	0.873	0.806	0.828	0.766	0.767	0.752	**0.765**
Reg. Logística	0.555	0.531	0.479	0.441	0.431	0.439	0.418	0.42	0.418	0.427
CART	0.836	0.83	0.837	0.82	0.812	0.82	0.78	**0.792**	**0.755**	0.734
Flor. Aleatórias	0.882	0.881	0.868	0.847	0.818	0.81	0.775	0.757	0.721	0.742
SVM	**0.933**	**0.921**	**0.912**	**0.887**	**0.833**	**0.845**	**0.783**	0.782	0.724	0.728
kNN	0.883	0.892	0.868	0.87	0.794	0.812	0.766	0.739	0.666	0.686

Tabela 5.34 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 4, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 1 para 0) introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 4, ao analisar conjuntos de dados sem ruído o rótulo, Random LORCy e SVM foram os métodos que obtiveram os melhores resultados em relação ao percentual médio de acertos na classificação de novas instâncias. Ao considerar os conjuntos de dados, introduzindo ruído no rótulo, o único método que esteve entre os melhores para todos os percentuais de ruído no rótulo testados foi o Random LORCy. O SVM foi muito bem para percentuais de troca de rótulo até 35%. O CART apresentou bons resultados para percentuais intermediários de troca de rótulo, estando entre os melhores para os percentuais de 20% até 35%.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.759	0.761	0.765	0.754	0.753	0.746	0.75	0.726	0.74	**0.749**
LORCy	**0.82**	0.796	0.793	0.753	0.728	0.716	0.722	0.696	0.696	0.695
Random LORC	0.772	0.762	0.775	0.769	0.75	0.743	0.748	0.717	0.704	0.726
Random LORCy	0.804	0.794	0.802	0.777	0.765	0.76	0.77	0.744	0.737	0.74
Reg. Logística	0.546	0.568	0.575	0.573	0.581	0.595	0.594	0.591	0.59	0.588
CART	0.753	0.762	0.762	0.761	0.743	0.72	0.734	0.707	0.716	0.712
Flor. Aleatórias	0.8	0.795	0.802	0.791	0.767	0.75	0.745	0.727	0.744	0.715
SVM	0.817	**0.813**	**0.814**	**0.797**	**0.792**	**0.764**	**0.783**	**0.761**	**0.76**	0.729
kNN	0.798	0.795	0.793	0.795	0.763	0.749	0.753	0.727	0.734	0.719

Tabela 5.35 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 5, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 1 para 0) introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 5, em conjuntos de dados sem ruído no rótulo, os métodos com melhores desempenho na clssificação de novas instâncias foram as variações da metodologia LORC que utilizam o rótulo na etapa da construção da AGM (LORCy e Random LORCy), juntamente com Florestas Aleatórias, SVM e kNN.

Para os conjuntos de dados com ruído no rótulo do tipo NAR trocando os rótulos da classe 1 para 0, Random LORCy e SVM apresentaram os melhores resultados para todos os percentuais de troca de rótulo testados, mostrando-se robustos para este tipo de ruído no Cenário 5. Para

percentuais baixos de troca de rótulo, LORCy obteve desempenho entre os melhores até o percentual 10%, e Florestas Aleatórias e kNN até 25%, com exceção do percentual 20%, no qual o kNN não esteve entre os melhores. LORC e Random LORC estiveram entreo os melhores para apenas alguns (2 e 3) percentuais de troca de rótulo entre os analisados.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.814	0.823	0.80	0.775	0.776	0.759	0.697	0.72	0.665	0.648
LORCy	0.871	0.818	0.816	0.755	0.737	0.707	0.716	0.694	0.623	0.676
Random LORC	0.855	0.848	0.827	0.813	0.834	0.77	0.715	0.615	0.578	0.529
Random LORCy	0.876	0.878	0.86	0.853	0.87	0.828	0.789	0.755	0.729	0.779
Reg. Logística	0.97	0.927	0.923	0.923	0.904	0.891	0.864	0.843	0.825	0.797
CART	0.975	0.973	0.968	0.955	0.927	0.904	0.897	0.839	0.831	0.801
Flor. Aleatórias	**0.987**	**0.985**	**0.984**	**0.981**	**0.97**	**0.964**	**0.944**	**0.911**	**0.892**	0.847
SVM	0.974	0.971	0.96	0.95	0.946	0.936	0.905	0.882	0.837	0.806
kNN	0.974	0.968	0.956	0.952	0.947	0.945	0.922	0.908	0.865	**0.862**

Tabela 5.36 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 6, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 1 para 0) introduzidos no conjunto de treinamento do algoritmo.

O Cenário 6 apresenta muitos atributos que correspondem a ruído. A partir da Tabela 5.36, podemos observar que nenhuma das variação da metodologia LORC conseguiu bons resultados para este cenário.

Ao observar os conjuntos de dados sem ruído no rótulo, todos os demais métodos (Regressão Logística, CART, Florestas Aleatórias, SVM e kNN) tiveram bons resultados. Ao analisar conjuntos de dados com ruído no rótulo, Florestas Aleatórias foi o método que se mostrou mais robusto, com os resultados entre os melhores para todos os percentuais de ruído introduzidos. O kNN também mostrou bom desempenho, estando entre os melhores para todos os percentuais de troca de rótulo, exceto para 15%. Finalmente, o SVM esteve entre os melhores para percentuais de ruído no rótulo de até 30% (exceto para 15%) e o CART até 15%, mostrando-se bons para percentuais mais baixos de troca de rótulo deste tipo.

Pelos resultados obtidos para o Cenário 6, podemos supor que a metodologia desenvolvida neste trabalho não é uma boa opção para tratar de conjuntos de dados com muitas variáveis de ruído, já que seu desempenho se mostrou aquém dos demais métodos especificamente para este cenário.

Para o Cenário 7, ao utilizar conjuntos de dados sem ruído no rótulo, todas as variações da metodologia LORC (ou seja, LORC, LORCy, Random LORC e Random LORCy) tiveram seus desempenhos entre os melhores, juntamente com CART, Florestas Aletatórias, SVM e kNN. Este cenário foi contruído de forma a propiciar bom desemepenho da metodologia LORC (assim como da maior parte dos demais métodos), de forma que este resultado está dentro do esperado.

Ao introduzir ruído do NAR, trocando rótulos das classe 1 para 0, o SVM esteve entre os métodos de melhor desempenho para todos os percentuais de troca de rótulo analisados. O Random LORC e o kNN também se mostraram muito bons, não estando entre os melhroes apenas para percentuais bem altos (35% e 45%). LORC, CART e Florestas Aleatórias apresentaram-se

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	**1**	0.975	0.974	0.978	0.96	0.961	0.941	0.94	0.949	0.822
LORCy	**1**	0.926	0.881	0.814	0.757	0.756	0.685	0.63	0.622	0.563
Random LORC	**1**	0.991	0.987	0.984	0.988	0.987	0.98	0.972	0.95	0.927
Random LORCy	**1**	0.977	0.946	0.93	0.891	0.874	0.847	0.809	0.806	0.752
Reg. Logística	0.70	0.70	0.70	0.65	0.589	0.466	0.495	0.575	0.599	0.60
CART	0.983	0.979	0.977	0.985	0.976	0.968	0.942	0.92	0.932	0.87
Flor. Aleatórias	0.998	0.996	0.986	0.979	0.953	0.941	0.911	0.844	0.868	0.801
SVM	**1**	**0.999**	**1**	**1**	**0.998**	**0.999**	**1**	**0.997**	**0.993**	**0.959**
kNN	**1**	**0.999**	0.997	0.995	0.988	0.997	0.993	0.983	0.96	0.88

Tabela 5.37 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 7, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 1 para 0) introduzidos no conjunto de treinamento do algoritmo.

entre os melhores para os percentuais mais baixos de troca de rótulo (até 15% para LORC e Florestas Aleatórias e até 20% para o CART). Podemos observar que o SVM teve mais de 95% de acurácia para todos os percentuais de ruído testados, mostrando um excelente desempenho neste cenário. LORC e Random LORC também se mostraram bem robustos, com acurácia acima de 92% para todos os percentuais de ruído.

	0%	5%	10%	15%	20%	25%	30%	35%	40%	45%
LORC	0.51	0.49	0.49	0.49	0.49	0.494	0.491	0.492	0.491	0.488
LORCy	0.995	0.925	0.878	0.866	0.816	0.806	0.777	0.74	0.71	0.655
Random LORC	0.516	0.499	0.501	0.506	0.493	0.497	0.491	0.49	0.49	0.49
Random LORCy	0.984	0.971	0.935	0.936	0.91	0.893	0.882	0.817	0.83	0.795
Reg. Logística	**0.996**	**0.999**	**0.999**	**0.999**	**1**	**1**	**1**	**1**	**1**	**1**
CART	0.49	0.561	0.506	0.526	0.503	0.516	0.49	0.495	0.501	0.505
Flor. Aleatórias	0.995	0.991	0.988	0.987	0.971	0.96	0.945	0.941	0.924	0.912
SVM	0.981	0.735	0.72	0.713	0.69	0.684	0.676	0.658	0.631	0.627
kNN	0.492	0.492	0.502	0.50	0.49	0.49	0.49	0.49	0.49	0.49

Tabela 5.38 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 8, com diferentes percentuais de troca de rótulo do tipo NAR (trocando 1 para 0) introduzidos no conjunto de treinamento do algoritmo.

O Cenário 8 foi criado com objetivo de representar um cenário no qual o LORC (e o Random LORC) apresenta grandes dificuldades, ou seja, é um cenário no qual o método não seria adequado. Já o LORCy (e o Random LORCy) apresenta modificações metodológicas capazes de contornar o problema, supostamente podendo apresentar bons resultados para este cenário. O Cenário 8 também é bem propício a um bom desempenho da Regressão Logística, o que foi visto nas categorias anteriores de testes (com ruído do tipo NCAR e NAR) e que também pode ser verificado nos resultados apresentados na Tabela 5.38, onde podemos perceber que este método aparece entre os melhores em relação ao percentual de acertos na classificação tanto para os conjuntos de dados sem ruído no rótulo quanto para os conjuntos com todos percentuais de troca de rótulo (desde 5% até 45%).

Quando não há rótulos trocados nos conjuntos de dados, além da Regressão Logística, os métodos LORCy, Random LORCy, Florestas Aleatórias e SVM também estão entre os melhores. Novamente as variações do LORC que utilizam o rótulo na primeira etapa do método (para a construção da AGM) foram capazes de contornar o problema encontrado pelas variações que não têm essa característica, em relação a este tipo de cenário.

Ao introduzir ruído no rótulo, o Random LORCy teve seu desempenho entre os melhores apenas para o 5% dos rótulos trocados. Já o Florestas Aleatórias acompanha o desempenho da Regressão Logística para percentuais de ruído no rótulo de até 15%. À medida que o percentual de ruído no rótulo aumenta, a acurácia apresentada destes métodos se distancia cada vez da Regressão Logística, que apresentou desempenho excelente neste cenário, independente do percentual de rótulos trocados. De toda forma, estes métodos apresentam desempenhos melhores que os demais (LORC, Random LORC, CART, SVM e kNN) neste cenário, para a maior parte dos percentuais de troca de rótulo analisados.

Finalizados os resultados para trocas de rótulos nas classes 1, encerramos as análises de conjuntos de dados simulados com ruído do tipo NAR. Na próxima seção apresentaremos os testes para ruído do tipo NNAR.

5.4.2.3 Ruído do Tipo NNAR

Este tipo de ruído é o que mais afeta o desempenho dos métodos de classificação. Ele pode variar bastante, pois ocorre de diversas formas diferentes. Por isso, é bastante difícil generalizar e simular um teste que possa representá-lo bem. Então, a solução que encontramos foi analisar um exemplo de configuração deste ruído que ocorre com certa frequencia. Dessa forma, optamos por analisar este tipo de ruído separadamente dos anteriores, sem utilizar os resultados desta seção para tirar conclusões gerais sobre os resultados.

Essa forma frequente do ruído do tipo NNAR ocorre quando há rótulos trocados próximos às fronteiras das regiões de classificação. Para representar este cenário, considere o seguinte: Se d_M é a distância máxima de qualquer ponto do *cluster* C_i para a linha que estabele a fronteira de decisão entre C_i e outro *cluster* C_j, então os pontos de C_i aptos a terem os rótulos alterados são os que estão à distância de até $\frac{d_M}{4}$ da linha da fronteira. Para introduzir o ruído NNAR nas variáveis, geramos o conjunto de dados de treinamento e, posteriormente, sorteamos aleatóriamente $x\%$ dos pontos aptos a terem os rótulos alterados para terem seus rótulos trocados, com $x \in 0\%, 10\%, 20\%, 30\%, 40\%, 50\%, 60\%, 70\%, 80\%, 90\%$. No caso dos testes implementados, o percentual de troca de rótulo é influenciado apenas pela proximidade com a fronteira de decisão, ou seja, pela proximidade com os elementos de outro *cluster* distinto. Dessa forma, a classe dos elementos não é levada em consideração ao estabelecer esse percentual.

O ruído NNAR foi implementado nos conjuntos de dados de 1 a 5, pois ele não ficaria bem estabelecido nos demais conjuntos de dados simulados. O conjunto de dados 6, que tem 20 dimensões, poderia ter o ruído implementado apenas nos dois atributos significantes para a classificação. Porém sem ter uma fronteira de decisão definida, diferentemente dos anteriores, a introdução do ruído conforme estabelecido nesta seção não poderia ser aplicada. O conjunto de dados 7, formado por *clusters* compactos, tem as regiões de classificação bem estabelecidas, com os *clusters* com uma distância grande um do outro, de forma que não haveriam pontos na região definida para possíveis trocas de rótulos. Além disso, também não há uma linha

definida da fronteira de decisão, assim como no conjunto de dados 6. O conjunto de dados 8 contém todos os pontos a uma mesma distância (muito pequena) da fronteira das regiões de classificação, de forma que também não faria sentido colocar ruído da forma definida nesta seção.

Os resultados obtidos em percentual médio de acertos nas 10 simulações estão exibidos nas Tabelas a seguir. Os valores em negrito em cada coluna (cada percentual de troca de rótulo) representam os maiores valores de classificação, ou seja, o método que obteve melhor desempenho médio para tal percentual de rótulos trocados no conjunto de dados. Considerando o desvio-padrão médio apresentado na Seção 5.3.2.3, as células coloridas em cada coluna correspondem aos valores cuja diferença para o maior daquela coluna (em negrito) é de até 0.028. Ou seja, para cada coluna, as linhas cujas células foram coloridas representam os métodos com melhor desempenho médio na classificação, segundo o percentual de acertos.

	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%
LORC	0.92	0.908	0.933	0.906	0.91	0.872	0.834	0.842	0.851	0.847
LORCy	0.987	0.966	0.961	0.948	0.944	0.926	0.9	0.907	0.889	0.879
Random LORC	0.964	0.957	0.965	0.959	0.943	0.914	0.92	0.913	0.908	0.909
Random LORCy	0.981	0.956	0.952	0.951	0.932	0.929	0.909	0.909	0.874	0.872
Reg. Logística	0.507	0.548	0.552	0.564	0.515	0.583	0.572	0.599	0.628	0.614
CART	0.877	0.888	0.876	0.881	0.914	0.892	0.897	0.881	0.885	0.869
Flor. Aleatórias	0.972	0.971	0.975	0.969	0.959	0.964	0.959	0.945	0.928	0.922
SVM	**0.993**	**0.991**	**0.988**	**0.985**	**0.983**	**0.979**	**0.969**	**0.965**	**0.936**	**0.928**
kNN	0.984	0.973	0.975	0.959	0.969	0.951	0.946	0.942	0.833	0.923

Tabela 5.39 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 1, com diferentes percentuais de troca de rótulo do tipo NNAR introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 1, ao observar os conjuntos de dados sem ruído no rótulo, os melhores desempenhos em relação à acurácia na classificação foram das variações da metodologia LORC que utilizam o rótulo na etapa de construção da AGM, ou seja, LORCy e Random LORCy, juntamente com Florestas Aleatórias, SVM e kNN.

Ao introduzir o ruído do tipo NNAR, Florestas Aleatórias, SVM e kNN foram os métodos que apresentaram os melhores desempenhos para todos os percentuais de troca de rótulo testados, mostrando-se robustos para este tipo de ruído no Cenário 1. Além deles, o Random LORC foi o que ficou entre os melhores para alguns dos percentuais de troca de rótulo mais baixos (20% e 30%) e mais altos (80% e 90%). Para percentuais baixos de troca de rótulo (de até 20%), LORCy também está entre os melhores.

Para o Cenário 2, quando não há troca de rótulo nos conjuntos de dados de treinamento do modelo, todos os métodos com exceção da Regressáo Logística, apresentaram resultados entre os melhores desempenhos na acúracia da classificação de novas instâncias.

Ao introduzir ruído do tipo NNAR no rótulo, o SVM e o Rnadom LORC foram os métodos que obtiveram seus desempenhos entre os melhores, em relação a acurácia na classificação, para todos os percentuais testados de troca de rótulo. O LORCy e o kNN também se mostraram boas opções para percentuais mais baixos, apresentando-se entre os melhores para para até 50% de

	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%
LORC	0.97	0.971	0.954	0.949	0.948	0.94	0.931	0.923	0.904	0.865
LORCy	0.997	0.996	0.991	0.979	0.971	0.964	0.932	0.914	0.905	0.87
Random LORC	0.99	0.988	0.986	0.984	0.983	0.979	**0.981**	0.968	**0.965**	**0.947**
Random LORCy	0.992	0.975	0.962	0.941	0.93	0.93	0.89	0.869	0.854	0.827
Reg. Logística	0.469	0.469	0.469	0.469	0.469	0.459	0.457	0.447	0.408	0.403
CART	0.97	0.975	0.969	0.962	0.945	0.92	0.88	0.879	0.868	0.854
Flor. Aleatórias	0.994	0.993	0.982	0.971	0.954	0.941	0.926	0.897	0.884	0.849
SVM	0.997	0.995	0.988	**0.993**	0.979	0.982	0.976	**0.971**	0.943	0.865
kNN	**0.998**	0.994	**0.996**	0.99	**0.987**	**0.983**	0.939	0.911	0.884	0.857

Tabela 5.40 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 2, com diferentes percentuais de troca de rótulo do tipo NNAR introduzidos no conjunto de treinamento do algoritmo.

troca de rótulo.

	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%
LORC	0.879	0.881	0.876	0.875	0.919	0.855	0.868	0.899	0.825	0.879
LORCy	0.957	0.934	0.934	0.944	0.908	0.898	0.891	0.896	0.855	0.875
Random LORC	0.924	0.915	0.913	0.909	0.906	0.903	0.894	0.894	0.86	0.891
Random LORCy	0.937	0.921	0.907	0.913	0.887	0.878	0.874	0.846	0.805	0.825
Reg. Logística	0.329	0.329	0.329	0.329	0.339	0.33	0.334	0.336	0.338	0.355
CART	0.967	0.968	0.963	0.942	0.923	0.894	0.903	0.865	**0.874**	0.863
Flor. Aleatórias	**0.979**	**0.98**	**0.968**	**0.965**	**0.945**	**0.926**	**0.92**	**0.921**	0.855	0.882
SVM	0.954	0.951	0.936	0.926	0.931	0.916	0.911	0.914	0.873	*0.906*
kNN	0.949	0.925	0.907	0.902	0.908	0.87	0.897	0.886	0.864	0.894

Tabela 5.41 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 3, com diferentes percentuais de troca de rótulo do tipo NNAR introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 3, que foi construído de forma a atender bem os requisitos do CART e do Florestas Aleatórias para obter bons desempenhos destes métodos, podemos observar que quando não há troca de rótulo nos conjuntos de dados utilizados, estes são os dois métodos que apresentam os melhore desempenhos em relação à acurácia da classificação, juntamente com o LORCy e SVM, cujos desempenhos também foram muito bons.

Ao considerar os conjuntos de dados com ruído do tipo NNAR, podemos observar que o método Florestas Aleatórias me mostra robusto, tendo os resultados entre os melhores para todos os percentuais de troca de rótulo testados. Já o CART, figura entre os melhores para a maior parte dos percentuais, exceto 50%, 70% e 90%. Para percentuais mais altos de troca de rótulo, também figuram entre os melhores o LORCy, o Random LORC e o SVM.

Para o Cenário 4, ao analisar conjuntos de dados sem ruído o rótulo, LORCy, Random LORCy, Florestas Aleatórias, SVM e kNN foram os métodos que obtiveram os melhores resultados em relação ao percentual médio de acertos na classificação de novas instâncias.

Ao considerar os conjuntos de dados, introduzindo ruído no rótulo do tipo NNAR, os úni-

	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%
LORC	0.822	0.817	0.792	0.771	0.776	0.776	0.73	0.732	0.731	0.731
LORCy	0.887	0.856	0.851	0.823	0.823	0.813	0.79	0.785	0.782	0.752
Random LORC	0.844	0.845	0.833	0.822	0.809	0.831	0.788	0.751	0.758	0.748
Random LORCy	0.904	0.894	0.88	0.864	0.884	0.869	0.83	0.835	0.831	0.781
Reg. Logística	0.521	0.508	0.492	0.457	0.478	0.457	0.414	0.433	0.432	0.417
CART	0.839	0.859	0.842	0.846	0.806	0.795	0.814	0.778	0.798	0.786
Flor. Aleatórias	**0.913**	0.92	**0.903**	0.889	**0.907**	**0.901**	0.865	**0.876**	0.859	**0.839**
SVM	**0.913**	**0.922**	**0.903**	**0.89**	0.881	0.888	**0.87**	0.87	**0.863**	0.831
kNN	0.889	0.888	0.884	0.874	0.874	0.851	0.837	0.834	0.809	0.804

Tabela 5.42 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 4, com diferentes percentuais de troca de rótulo do tipo NNAR introduzidos no conjunto de treinamento do algoritmo.

cos métodos que estiveram entre os melhores para todos os percentuais de ruído no rótulo testados foram Florestas Aleatórias e SVM. O Random LORCy esteve entre os melhores para os percentuais mais baixos, de até 40%.

	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%
LORC	0.774	0.762	0.779	0.76	0.778	0.763	0.761	0.739	0.699	0.698
LORCy	0.859	0.847	0.815	0.803	0.799	0.766	0.768	0.748	0.721	0.706
Random LORC	0.789	0.802	0.799	0.764	0.787	0.7613	0.767	0.745	0.736	0.728
Random LORCy	0.864	0.832	0.822	0.8	0.779	0.779	0.75	0.74	0.711	0.693
Reg. Logística	0.46	0.502	0.484	0.54	0.567	0.571	0.529	0.57	0.544	0.57
CART	0.798	0.785	0.752	0.765	0.727	0.729	0.717	0.719	0.724	0.699
Flor. Aleatórias	**0.876**	0.857	**0.856**	0.827	0.81	0.795	0.792	0.777	0.743	0.734
SVM	0.869	**0.858**	0.85	**0.85**	**0.845**	**0.84**	**0.842**	**0.821**	**0.763**	**0.754**
kNN	0.838	0.835	0.837	0.832	0.83	0.811	0.819	0.79	0.76	0.728

Tabela 5.43 Percentual médio de acertos dos métodos de classificação supervisionada para o Conjunto de Dados 5, com diferentes percentuais de troca de rótulo do tipo NNAR introduzidos no conjunto de treinamento do algoritmo.

Para o Cenário 5, em conjuntos de dados sem ruído no rótulo, os métodos com melhores desempenho na clssificação de novas instâncias foram as variações da metodologia LORC que utilizam o rótulo na etapa da construção da AGM (LORCy e Random LORCy), juntamente com Florestas Aleatórias e SVM.

Ao considerar os conjuntos de dados, introduzindo ruído no rótulo do tipo NNAR, o único método que esteve entre os melhores para todos os percentuais de ruído no rótulo testados foi o SVM. O kNN foi bem para grande parte dos percentuais de troca de rótulo, exceto 50% e 70%. Florestas Aleatórias teve bom desemepnho para os percentuais mais baixos (10%, 20% e 30%) e mais altos (80% e 90%). O Random LORC só esteve entre os melhores para os percentuais mais altos 80% e 90%.

A partir dos resultados obtidos nos 5 conjuntos de dados com diversos percentuais de ruído do tipo NNAR introduzidos nas regiões próximas às fronteiras das regiões de classificação,

podemos tirar algumas conclusões:

- O SVM foi o método que se mostrou mais robusto a este tipo de ruído na maioria dos conjuntos de dados, pois esteve entre os métodos de melhor desempenho para todos os percentuais de troca de rótulo em 4 dos 5 conjuntos de dados.

- Logo após o SVM, o Florestas Aleatórias também se mostrou robusto a esta forma de ruído do tipo NNAR, pois esteve entre os melhores para todos os percentuais em 3 dos 5 conjuntos de dados.

- Entre as variações do LORC, o Random LORC foi o método que apresentou maior robustez para este tipo de ruído, no geral, especialmente para percentuais mais altos de troca de rótulo. As variações LORcy e Random LORCy também estiveram entre os melhores algumas vezes, especialmente quando o percentual de troca de rótulo é baixo.

- Regressão Logística e CART foram os métodos que apresentaram os piores desempenhos em todos os conjuntos de dados. A exceção ocorreu apenas no conjunto de dados 3, no qual o CART obteve bom desempenho para baixos percentuais de troca de rótulo. Mas isso se deve a este conjunto ter sido desenhado de acordo com o CART, proporcionando seu bom desempenho. Mesmo assim, para a maior parte dos percentuais de troca de rótulo ele não esteve entre os métodos de melhor desempenho.

As conclusões que podem ser feitas a partir dos testes desta seção são bastante específicos para a configuração dos testes que foi estabelecida. Desta forma, ressaltamos novamente que é difícil generalizar conclusões para ruído do tipo NNAR, devido a grande variação de possíveis configurações. De toda forma, para este tipo específico de configuração, podemos dizer que nossa metodologia ficou aquém de outros métodos de classificação, especialmente o SVM e o Florestas Aleatórias.

5.4.3 Comentários

Nesta seção apresentaremos um resumo dos resultados apresentados neste capítulo para os ruídos no rótulo dos tipos NCAR e NAR, juntamente com comentários e conclusões que podem ser obtidas a partir deles.

Em primeiro lugar, observamos que o ruído no rótulo afeta todos os métodos implementados. A medida que o percentual de ruído no rótulo vai aumentando, o valor da acurácia média dos métodos, em geral, tende a ir diminuindo. Portanto, este é realmente um problema importante a ser analisado.

Vamos analisar inicialmente os resultados obtidos para os conjuntos de dados sem ruído no rótulo. Observe que temos 3 valores médios de acurácia para cada método, sendo que cada um foi obtido em conjuntos de dados distintos segundo os cenários propostos. Estes resultados estão sempre nas primeiras colunas das Tabelas 5.12 a 5.35, sendo que as 8 primeiras Tabelas também mostram os resultados dos testes com conjuntos de dados afetados pelo ruído NCAR, as 8 intermediárias pelo ruído NAR com troca de rótulos da classe 0 e as 8 últimas pelo ruído NAR com troca de rótulo da classe 1.

5.4.3.1 Conjuntos de Dados Sem Ruído no Rótulo

Para cada um dos Cenários implementados, temos:

- Para o Cenário 1, os melhores métodos nas 3 Tabelas são sempre os mesmos: LORCy, Random LORCy, Florestas Aleatórias, SVM e kNN.
- Para o Cenário 2, os melhores métodos nas 3 Tabelas são os mesmos quase sempre: LORC, LORCy, Random LORC, Random LORCy, Florestas Aleatórias (exceto para a Tabela de ruído do tipo NAR trocando rótulos da classe 0 para 1), SVM e kNN.
- Para o Cenário 3, os melhores métodos nas 3 Tabelas são os mesmos quase sempre: LORCy (exceto para a Tabela de ruído do tipo NAR trocando rótulos da classe 1 para 0), CART e Florestas Aleatórias.
- Para o Cenário 4, os métodos que são melhores para todas as 3 Tabelas são o Random LORCy e o SVM, além do LORCy (exceto para a Tabela de ruído do tipo NAR trocando rótulos da classe 1 para 0), o Florestas Aleatórias (exceto para a Tabela de ruído do tipo NAR trocando rótulos da classe 1 para 0) e o kNN (exceto para a Tabela de ruído do tipo NAR trocando rótulos da classe 1 para 0).
- Para o Cenário 5, os melhores métodos para as 3 Tabelas são sempre os mesmos: LORCy, Random LORCy, Florestas Aleatórias, SVM e kNN.
- Para o Cenário 6, os métodos que são melhores para as 3 Tabelas são o CART e o Florestas Aleatórias. Os métodos Regressão Logística, SVM e kNN foram melhores para as Tabelas do ruído NCAR e do ruído NAR trocando os rótulos da classe 1 para 0, mas não obtiveram o mesmo sucesso trocando os rótulos da classe 0 para 1.
- Para o Cenário 7, os melhores métodos para as 3 Tabelas são os mesmos sempre: LORC, LORCy, Random LORC, Random LORCy, CART, Florestas Aleatórias, SVM e kNN, ou seja, todos exceto a Regressão Logística.
- Para o Cenário 8, os melhores métodos para as 3 Tabelas são sempre os mesmos: LORCy, Random LORCy, Regressão Logística, Florestas Aleatórias e SVM.

Consideraremos que um método de classificação esteve entre os melhores para um determinado cenário se em pelo menos 2 dos 3 resultados referentes ao cenário em questão, para conjuntos de dados sem ruído no rótulo, este método apareceu entre os melhores. Assim, observando os resultados resumidos, podemos obter as seguintes análises:

- Os conjuntos de dados nos quais a maior parte dos métodos teve certa dificuldade para obtér bons resultados em relação à acurária das classificações foram os correspondentes ao Cenário C3. Este cenário foi construído para ser adequado à metodologia de classificação do CART e do Florestas Aleatórias, de forma que o bom desempenho desses métodos, observado nos resultados, era esperado. Nenhum dos outros métodos implementados foi capaz de ter seu desempenho acompanhando o do CART e do Florestas

Aleatórias neste cenário, com exceção do LORCy, que se mostrou uma ótima opção, assim como os dois primeiros.

- Os conjuntos de dados nos quais a maior parte dos métodos teve certa facilidade para obtér bons resultados em relação à acurácia das classificações foram os correspondentes aos Cenários C2 e C7. Em ambos os cenários todos os métodos foram eficientes na classificação, com exceção do CART e da Regressão Logística para o Cenário C2 e apenas da Regressão Logística para o C7.

- Nenhuma variação da metodologia LORC se mostrou robusta em relação ao ruído nas variáveis, conforme podemos observar nos resultados do Cenário C6, nos quais o desempenho de todas elas ficou muito aquém dos demais métodos testados.

- As variações do LORC que utilizam o rótulo na etapa de construção da AGM (LORCy e Random LORCy) foram capazes de contornar o problema dos *clusters* rotulados complexos, representados no Cenário C8, apresentando bons resultados juntamente com Regressão Logística, Florestas Aleatórias e SVM.

- Conforme previsto no desenvolvimento metodológico, as variações LORCy (principalmente) e Random LORCy foram as que apresentaram melhores resultados para os conjuntos de dados sem ruído no rótulo. Apenas no Cenário C6 nenhuma delas aparece entre as melhores na acurácia média das classificações, mas isso se deve à falta de robustez da metodologia em relação a ruído nas variáveis. Nos demais cenários vistos, sempre as duas (na maior parte das vezes) ou uma delas (apenas para o Cenário C3) está entre as melhroes.

- Os principais concorrentes do LORCy e do Random LORCy para os conjuntos de dados simulados sem ruído no rótulo foram o SVM e o Florestas Aleatórias, já que o SVM só não teve seu desempenho entre os melhores para um dos cenários (C3) e o Florestas Aleatórias apresentou-se como um bom método em todos os cenários analisados sem ruído no rótulo.

- Os métodos que apresentaram os piores resultados, em geral, foram o CART e a Regressão Logística, muitas vezes devido aos formatos dos conjuntos de dados simulados não serem adequados às restrições apresentadas nestas metodologias.

SVM e Florestas Aleatórias, que tiveram bons desempenhos, são métodos de conhecida eficência na literatura. Ao obter resultados para os métodos LORCy e Random LORCy tão bons quanto o destes métodos para grande parte dos cenários simulados (e até melhor para alguns), temos indícios de que nossa metodologia pode ser utilizada com eficiência em diversos casos. Além disso, as outras metodologias utilizadas como comparação, também são utilizadas com frequência na prática, de forma que nossa metodologia ter superado o desempenho delas em grande parte dos testes realizados também corrobora com a qualidade do método. Portanto, LORCy e Random LORCy se apresentam como ótimas opções para classificação supervisionada, quando os conjuntos de dados de treinamento não têm problema de ruído no rótulo.

5.4.3.2 Conjuntos de Dados Com Ruído no Rótulo

Primeiramente vamos analisar os resultados dos conjuntos de dados nos quais foi introduzido ruído do NCAR, que foram apresentados na Seção 5.4.2.1. No geral, para esse tipo de ruído, a variação da metodologia LORC que apresentou melhores resultados em relação à acurácia da classificação de novas instâncias foi o Random LORC. Isso fortalece a idéia de que ao introduzir a reamostragem Boostrap no método LORC, a eficiência para os conjuntos de dados com ruído no rótulo poderia ser melhor.

Inicialmente, para os 5 primeiros Cenários implementados, temos:

- Para o Cenário C1, o melhor método foi o kNN, com seu desempenho entre os melhores para todos os percentuais de troca de rótulo. Para percentuais de troca de rótulo menores até 35%, o Random LORC também foi ótimo, se mantendo entre os melhores, assim como o SVM (mas este também ficou entre os melhores para o percentual 45%). Ou seja, as 3 metodologias tiveram grande sucesso neste Cenário, especialmente ao tratar de conjuntos de dados com percentual de trocas de rótulo abaixo de 40%.

- Para o Cenário C2, o melhor método foi também o kNN, com seu desempenho entre os melhores para todos os percentuais de troca de rótulo. Para percentuais de troca de rótulo até 40%, o SVM também foi ótimo, se mantendo entre os melhores. O Random LORC também se saiu muito bem neste cenário, estando entre os melhores para todos os percentuais de troca de rótulo, exceto 35% e 45%. Ou seja, as 3 metodologias tiveram grande sucesso neste Cenário, especialmente em conjuntos de dados com trocas de rótulo em percentuais abaixo 35%.

- Para o Cenário C3, o CART se mostrou robusto apenas para percentuais mais baixos de ruído no rótulo, estando entre os melhores métodos para percentuais de até 25%. Porém, para percentuais maiores, ele não aparece mais entre os melhores, aparentando não ser tão robusto quando há muitos rótulos trocados. O Random LORC se destaca neste cenário, pois aparece entre os melhores métodos para todos os percentuais de ruído no rótulo maiores que 15%. Podemos observar ainda que para percentuais de até 15%, quando ele não aparece entre os melhores segundo o critério definido, ele acerta por volta de 5% a menos que o CART, em média. Dessa forma, o Random LORC se mostra uma boa opção para o caso em que é sabido que há um nível considerável de ruído no rótulo, porém não se sabe qual a quantidade.

- Para o Cenário C4, o SVM aparenta ser um pouco melhor que os demais métodos, tendo seus desempenho entre os melhores para todos os percentuais de troca de rótulo até 40%. Para percentuais de troca de rótulo de até 30% (exceto para 20%), Random LORCy e Florestas Aleatórias também têm seus desempenhos entre os melhores. E para percentuais de até 25% e a partir de 40%, o kNN também é bom neste cenário. Todos estes métodos se mostraram adequados para analisar conjuntos de dados com desenho semelhante ao Cenário 4.

- Para o Cenário C5, Random LORC e SVM tiveram seus desempenhos entre os melhores para todos os percentuais de troca de rótulo analisados, sendo os melhores métodos neste

cenário, quando os conjuntos de dados de treinamento apresentam ruído do tipo NCAR. O kNN não ficou entre os melhores para dois valores dos percentuais de troca de rótulo no Cenário 5 (25% e 30%). Os 3 métodos são boas opções neste cenário.

Observando os resultados de acurácia média na classificação para os 5 primeiros cenários, introduzindo ruído no rótulo do tipo NCAR nas instâncias dos conjuntos de dados, podemos observar que 3 métodos tiveram destaque ao considerarmos todos os percentuais analisados de ruído: Random LORC, SVM e kNN. Em geral, estes métodos figuraram entre os melhores para quase todos os percentuais de ruído no rótulo analisados para estes cenários, com algumas exceções: o Random LORC não esteve entre os melhores no Cenário 4 (mas neste cenário o Random LORCy mostrou bons resultados), o SVM e o kNN no Cenário 3. Alguns métodos, como Florestas Aleatórias, apresentaram bom desempenho apenas para percentuais baixos de ruído NCAR (até 10%, na maior parte dos cenários). Como geralmente o percentual de ruído no rótulo em um conjunto de dados reais não é conhecido, podemos dizer que a partir destes resultados, seria prudente utilizar o Random LORC, o SVM ou o kNN para a classificação em conjuntos de dados com ruído NCAR. Conforme observado, eles têm bom desempenho, em geral, sendo que um pode se apresentar melhor que os demais para determinadas configurações dos pontos que compõem o conjunto de dados de treinamento do algoritmo.

Agora vamos ver, resumidamente, os resultados para os Cenários 6, 7 e 8:

- Para o Cenário C6, o melhor método foi o kNN. Novamente, podemos observar que para conjuntos de dados com variáveis de ruído, a metodologia LORC não se mostra eficiente, assim como observamos quando não havia ruído no rótulo.

- Para o Cenário C7, o melhor método foi o SVM, sendo que kNN e Random LORC também ficaram entre os melhores para percentuais de troca de rótulo até 40% e 25%, respectivamente. Teoricamente, esperávamos que a metologia LORC (com suas variações) tivesse ótimo desempenho neste cenário, mas não foi isso o que observamos nas aplicações. Apesar de não terem ficado entre os melhores para grande parte dos percentuais de troca de rótulo, podemos observar que LORC e Random LORC tiveram acurácia média sempre acima de 85%, sendo bons resultados apesar de não serem ótimos, conforme a teoria. Provavelmente isso ocorreu pois na prática cada subconjunto de um *cluster* não tinha proporções idênticas (ou semelhantes) de rótulos trocados, assim como ocorre na teoria. Isso só seria possível caso o conjunto de dados tivesse uma quantidade muito grande de instâncias e os pontos a terem rótulos trocados fossem cuidadosamente selecionados. Superficialmente, podemos considerar que quando o percentual de troca de rótulo é baixo, temos uma configuração mais próxima desta. Tanto que quando o percentual de ruído do tipo NAR é de 5%, temos LORC, Random LORC e LORCy com desempenhos entre os melhores.

- Para o Cenário C8, conforme esperado, a Regressão Logística manteve um ótimo desempenho para todos os percentuais de troca de rótulo analisados, sendo o melhor método neste cenário. Podemos observar que, em relação os SVM, que geralmente é uma metodologia de bastante sucesso, o Random LORCy apresentou valores de acurácia média mais alta na grande maioria dos percentuais de troca de rótulo analisados.

Agora apresentaremos as análises dos resultados dos conjuntos de dados nos quais foi introduzido ruído do NAR, que foram apresentados na Seção 5.4.2.2. No geral, para esse tipo de ruído, duas variações da metodologia LORC apresentaram bons resultados em relação à acurácia da classificação de novas instâncias: o Random LORC (assim como nos conjuntos de dados com ruído NCAR) e o LORCy (assim como nos conjuntos de dados sem ruído no rótulo).

Inicialmente, para os 5 primeiros Cenários implementados, temos:

- Para o Cenário C1, ao introduzir ruído NAR nas classes 0, os melhores métodos foram LORCy (melhor em todos os percentuais de troca de rótulo) e Random LORC (melhor em todos os percentuais, exceto em 5% e 25%). Para percentuais de troca de rótulo acima de 25%, estes dois métodos foram melhores que todos os demais. Para percentuais mais baixos de troca de rótulo, SVM (até 25%) e kNN (até 20%) também ficaram entre os melhores, juntamente com o LORCy e o Random LORCy. De forma geral, as duas variações da metodologia LORC que utilizam o rótulo na construção da AGM se mostraram as mais adequadas neste cenário. Ao introduzir ruído NAR nas classes 1, os melhores métodos foram Random LORC e SVM (melhores em todos os percentuais de troca de rótulo), seguidos pelo kNN (melhor em todos os percentuais, exceto em 40%). De forma geral, estes 3 métodos se mostraram os mais adequadas neste cenário. Considerando os resultados dos ruídos NAR implementados nas duas classes, podemos observar que o Random LORC é o método mais adequado este cenário, pois obteve ótimos resultados em quase todos conjuntos de dados com este tipo de ruído que seguem o Cenário C1. Em seguida, SVM e kNN se mostram boas opções.

- Para o Cenário C2, ao introduzir ruído NAR nas classes 0, os melhores métodos (melhores em todos os percentuais de troca de rótulo) foram LORCy e kNN. Logo em seguida, temos o Random LORC (melhor em todos os percentuais, exceto em 15%) e o SVM (melhor em todos os percentuais, exceto em 40%). Ao introduzir ruído NAR nas classes 1, o melhor método foi o SVM (melhor em todos os percentuais de troca de rótulo), seguido pelo LORC (melhor em todos os percentuais, exceto em 40%), pelo Random LORC (melhor em todos os percentuais abaixo de 35%) e pelo kNN (melhor em todos os percentuais abaixo de 35%, exceto em 10%). Considerando os resultados dos ruídos NAR implementados nas duas classes, podemos observar que o SVM é o método mais adequado este cenário, pois obteve ótimos resultados em quase todos conjuntos de dados com este tipo de ruído que seguem o Cenário C2. Logo em seguida, também se apresentam como boas opções o Random LORC e o kNN.

- Para o Cenário C3, ao introduzir ruído NAR nas classes 0, o melhor método (melhor em todos os percentuais de troca de rótulo) foi o LORCy. Em seguida, temos o CART (melhor em todos os percentuais até 30%). Ao introduzir ruído NAR nas classes 1, o melhor método foi o Florestas Aleatórias (melhor em todos os percentuais de troca de rótulo), seguido pelo CART para baixos percentuais de troca de rótulo (melhor em todos os percentuais até 25%) e pelo Random LORCy para altos percentuais de troca de rótulo (melhor em todos os percentuais acima de 25%). Considerando os resultados dos ruídos NAR implementados nas duas classes, fica difícil estabeler um único método que seja

melhor em todos os casos. Temos o LORCy como campeão para o ruído na classe 0 e o Florestas Aleatórias como campeão para o ruído na classe 1, porém eles não apresentam o mesmo desempenho na outra categoria do ruído NAR.

- Para o Cenário C4, ao introduzir ruído NAR nas classes 0, o melhor método foi o LORCy (melhor em todos os percentuais de troca de rótulo). Em seguida, temos o SVM (melhor em todos os percentuais até 35%) e o Random LORCy (melhor em todos os percentuais até 30%). Ao introduzir ruído NAR nas classes 1, o melhor método foi o Random LORCy (melhor em todos os percentuais de troca de rótulo), seguido pelo SVM (melhor em todos os percentuais até 35%). Considerando os resultados dos ruídos NAR implementados nas duas classes, podemos observar que o Random LORCy e o SVM são os métodos mais adequados, com uma pequena vantagem para o Random LORCy devido ao SVM não ser bom em nenhuma das duas categorias quando o percentual de troca de rótulo é maior que 35%.

- Para o Cenário C5, ao introduzir ruído NAR nas classes 0, o melhor método (melhor em todos os percentuais de troca de rótulo) foi o LORCy. Em seguida, temos os métodos que apresentaram bons resultados apenas para percentuais mais baixos de ruído no rótulo, o SVM (melhor em todos os percentuais até 25%) e o Florestas Aleatórias (melhor em todos os percentuais até 20%), e o Random LORC que apresentou bons resultados para percentuais mais altos de ruído no rótulo (melhor em todos os percentuais acima 20%). Ao introduzir ruído NAR nas classes 1, os melhores métodos foram o Random LORCy e o SVM (melhores em todos os percentuais de troca de rótulo), seguidos pelo Florestas Aleatórias que teve bons resultados apenas para baixos percentuais de troca de rótulo (melhor em todos os percentuais até 25%). Considerando os resultados dos ruídos NAR implementados nas duas classes, é difícil afirmar que um método é melhor para as ambas. Caso haja certeza de que o percentual de troca de rótulo no conjunto de dados a ser analisado é baixo, o método mais adequado poderia ser o SVM, seguido do Florestas Aleatórias e do LORCy. Caso contrário, seria interessante saber em qual classe há maiores probabilidades de ruído, de forma que caso fosse na classe 0 o métodos mais adequados seria o LORCy, e caso fosse na classe 1, o Random LORCy ou o SVM.

Observando estes resultados resumidos para os 5 primeiros cenários, podemos concluir que:

- LORCy é o melhor método em todos os cenários ao tratar conjuntos de dados com qualquer percentual de ruído no rótulo do tipo NAR na classe 0 entre 5% e 45%. Apenas para percentuais bastante baixos deste tipo de ruído, Florestas Aleatórias se mostra bom para todos os 5 cenários. SVM mostra bom desempenho para grande parte dos conjuntos de dados com este tipo de ruído, exceto para percentuais altos de ruído e para o Cenário C3, no qual ele não fica entre os melhores para nenhum percentual de ruído. Enfim, o LORCy é melhor absoluto para este tipo de ruído nos 5 Cenários analisados.

- Para os conjuntos de dados com ruído no rótulo do tipo NAR na classe 1, não houve um método absolutamente melhor como ocorreu com o LORCy no caso anterior. Em geral, podemos observar um melhor desempenho do SVM, que foi o melhor método em 3 dos 5

cenários (Cenários C1, C2 e C5). No Cenário C4, ele também teve um bom desempenho, sendo que apenas no Cenário C3 o método não esteve entre os melhores para a maior parte dos percentuais de ruído no rótulo. Nestes cenários nos quais o SVM não obteve sucesso total, tivemos Florestas Aleatórias como melhor para todos os percentuais de ruído no rótulo no Cenário C3 (porém não apareceu como melhor em nenhum outro cenário) e Random LORCy como melhor para todos os percentuais de ruído no rótulo no Cenário C4 (sendo que ele também se mostrou melhor para todos os percentuais de ruído no rótulo no Cenário C5, junto ao SVM). Forçando uma generalização para este tipo de ruído, podemos dizer que o melhor método é o SVM, seguido pelo Random LORCy.

- Não conseguimos chegar a um método comum cujo desempenho seja o melhor para as duas categorias de ruído NAR. Isso significa que é importante conhecer o conjunto de dados a ser analisado para saber em qual classe de rótulos é provavel que haja maior proporção de instâncias mal rotuladas. Dessa forma, teremos mais ferramentas para escolher melhor o método de classificação para analisar o conjunto de dados.

- Em relação às variações da metodologia LORC, podemos observar que as que utilizam o rótulo na etapa de construção da AGM (LORCy e Random LORCy) têm melhor desempenho que as outras quando o ruído no rótulo presente nos conjuntos de dados é do tipo NAR.

Agora vamos ver, resumidamente, os resultados para conjuntos de dados com ruído do tipo NAR para os Cenários 6, 7 e 8:

- Para o Cenário 6, ao introduzir ruído NAR nas classes 0, nenhum método foi melhor em todos os percentuais de troca de rótulo. O que teve sucesso em uma quantidade maior de percentuais de ruído foi o LORC (melhor em todos os percentuais a partir de 20%), que só não teve ser desemepnho entre os melhores para os 3 percentuais mais baixos testados. Em seguida, temos o CART (melhor em todos os percentuais até 20% e para 30%), que apresentou bons resultados apenas para percentuais mais baixos de ruído no rótulo. Ao introduzir ruído NAR nas classes 1, o melhor método foi o Florestas Aleatórias (melhor em todos os percentuais de troca de rótulo), seguido pelo kNN (melhor em todos os percentuais exceto 15%). Mais distantes, temos o SVM (melhor para percentuais de troca de rótulo até 30%, exceto para 15%) e o CART (melhor para todos os percentuais de troca de rótulo até 15%) com bons desempenhos na categorias em que o percentual de ruído no rótulo é baixo. Considerando os resultados dos ruídos NAR implementados nas duas classes, é difícil afirmar que um método é melhor para as ambas. Caso haja certeza de que o percentual de troca de rótulo no conjunto de dados a ser analisado é baixo mas não se saiba qual das classes de rótulo tem maior probabilidade de troca de rótulo, o método mais adequado poderia ser o CART. Caso seja conhecida a classe com maior probabilidade de dados mal rotulados, se for a classe 1, o SMV e o kNN seriam mais adequados; se for a classe 0, o LORC (de preferência se o percentual esperado de ruído no rótulo não for muito baixo).

- Para o Cenário 7, ao introduzir ruído NAR nas classes 0, 3 métodos foram melhores em todos os percentuais de troca de rótulo: LORCy, SVM e kNN. Exceto para os percentuais

mais altos de troca de rótulo, o Random LORC também ficou entre os melhores (melhor em todos os percentuais até 30%, mas com ótima acurácia média acima deste percentual também). Ao introduzir ruído NAR nas classes 1, o melhor método foi o SVM (melhor em todos os percentuais de troca de rótulo), seguido pelo Random LORC e pelo kNN (melhores em todos os percentuais até 35%). Considerando os resultados dos ruídos NAR implementados nas duas classes, podemos perceber que, no geral, o SVM é o melhor método neste cenário, seguido de perto pelo Random LORC e kNN, que também são ótimas opções.

- Para o Cenário 8, ao introduzir ruído NAR nas classes 0, a Regressão Logística foi o melhor método em todos os percentuais de troca de rótulo. Em seguida, LORCy (melhor em todos os percentuais até 25%) ficou entre os melhores para percentuais mais baixos de ruído no rótulo, assim como Florestas Aleatórias (melhor em todos os percentuais até 20%). Ao introduzir ruído NAR nas classes 1, o melhor método também foi a Regressão Logística (melhor em todos os percentuais de troca de rótulo), seguido de longe pelo Florestas Aleatórias (melhor em todos os percentuais até 25%). Considerando os resultados dos ruídos NAR implementados nas duas classes, podemos perceber que a Regressão Logística é o melhor método neste cenário.

As conclusões a respeito dos resultados obtidos para o ruído no rótulo do tipo NNAR estão concentradas na Seção 5.4.2.3, por se tratarem de um caso específico desse tipo de ruído, que proporciona as mais diversas possibilidades de configurações. Por ser um ruído de difícil generalização, seus resultados foram apresentados separadamente dos ruídos do tipo NCAR e NAR.

Para finalizar, vamos tirar algumas conclusões a respeitos dos métodos de classificação aplicados aos conjuntos de dados simulados com ruído no rótulo, especialmente segundo os modelos NCAR e NAR.

Conforme já é sabido no campo de aprendizagem de máquina, não existe um método que é melhor para todos os tipos de conjuntos de dados. Cada método tem pontos altos e baixos, obtendo sucesso em determinados cenários e fracasso em outros. Nosso objetivo com estes testes foi de tentar verificar, na prática, os casos em que as variações desenvolvidas da metodologia LORC podem ter vantagens e desvantagens sobre os demais métodos já consagrados na área. Com essa perspectiva, podemos colocar as seguintes conclusões:

- Quando os conjuntos de dados analisados não apresentam dados mal rotulados (ruído no rótulo), a variação LORCy é uma ótima opção para classificação nos cenários em que não há variáveis de ruído. Random LORCy também é uma opção, mas geralmente recomendamos a primeira, pois além de apresentar mais resultados entre os melhores, ela é menos complexa computacionalmente.

- Quando os conjuntos de dados utilizados apresentam ruído no rótulo NCAR, a variação com melhor desempenho é o Random LORC. Ela costuma apresentar bons resultados em relação à acurácia na classificação, porém os métodos já consolidados SVM e kNN têm resultados muito bons neste cenário também, de forma a serem preferíveis ao Random LORC, em geral.

- Quando os conjuntos de dados utilizados apresentam ruído no rótulo NAR com dados mal rotulados apenas na classe 0 (segundo as configurações propostas nos cenários testados), a variação LORCy tem ótimo desempenho, sendo bastante recomendada como opção de metodologia análises similares. Em alguns casos deste tipo, as variações Random LORC e Random LORCy também são recomendadas, mas o LORCy é preferível, pois além de apresentar mais resultados entre os melhores, ele é menos complexo computacionalmente.

- Quando os conjuntos de dados utilizados apresentam ruído no rótulo NAR com dados mal rotulados apenas na classe 1 (segundo as configurações propostas nos cenários testados), a variação Random LORCy é a que apresenta melhor desempenho. O SVM tem melhores desempenhos, em geral, mas isso depende do formato dos conjuntos de dados.

- Quando os conjuntos de dados utilizados apresentam ruído no rótulo NNAR, observamos que é muito difícil simular algum conjunto de dados que represente este ruído de uma forma mais geral. Desta forma, escolhemos uma configuração específica de ruído no rótulo que pode ser modelada pelo NNAR para executar os testes. Para esta configuração específica, os melhores métodos foram o SVM e o Florestas Aleatórias. Entre as variações do LORC, o melhor foi o Random LORC, especialmente para percentuais mais altos de troca de rótulo.

- Com relação as variações do LORC e seu desempenho diante do problema de ruído no rótulo, temos uma conclusão importante que foi observada a partir dos resultados dos testes em conjuntos de dados simulados. Quando o ruído no rótulo ocorre equilibradamente nas duas classes (ruído NCAR), é mais adequado utilizar a metodologia que não utiliza o rótulo na etapa de construção da AGM. Mais especificamente o ideal é utilizar o Random LORC, já que as variações que utilizam o bootstrap são melhores opções para conjuntos de dados com ruído no rótulo. Já no caso do ruído ocorrer em apenas uma das classes ou a proporção de dados mal rotulados ser muito maior em uma das classes do que na outra (ruído NAR), é mais adequado utilizar a metodologia que utiliza o rótulo na etapa de construção da AGM (LORCy ou Random LORCy), apesar do LORC e, principalmente o Random LORC, também apresentarem bons resultados (mesmo assim, aquém do LORCy e do Random LORCy) na maioria dos conjuntos de dados simulados.

CAPÍTULO 6

Aplicações a Dados Reais

Os problemas reais de classificação com os quais costumamos nos deparar envolvem muitas dimensões, de forma a não ser possível uma clara vizualização do "formato"em que os pontos estão distribuidos no espaço, como foi possível com os dados simulados em duas dimensões. Dessa forma, fica difícil prever qual dos métodos de classificação seria supostamente mais adequado a cada conjunto de dados analisado.

Para analisar a eficiência do LORC e suas variações em conjuntos de dados reais, foram selecionados 5 bancos de dados disponíveis na internet [Lichman, 2013]. A eles foram aplicados o LORC, LORCy, Random LORC e Random LORCy, além da Regressão Logística, CART, Florestas Aleatórias, SVM e kNN, para efeitos de comparação dos resultados. Cada um dos conjuntos de dados foi dividido em um conjunto de construção do modelo e outro de avaliação. Para evitar qualquer tipo de tendência nessa divisão, utilizamos amostragem sistemática para selecionar o grupo de avaliação do modelo. Os métodos foram aplicados 5 vezes a cada conjunto de dados, variando os conjuntos de construção e validação do modelo. Para cada uma dessas 5 aplicações, foi selecionado um número inicial i entre 1 e 5 (sem repetição) e o conjunto de teste foi composto de todos os pontos nas posições $i+5k$, com $k = 0, ..., n/5$, onde n é o número total de elementos do conjunto de dados. Caso $n/5$ não seja inteiro, substitui-se pelo maior inteiro menor que $n/5$. Dessa forma, o conjunto de validação do modelo é constituido por cerca de $1/5$ dos dados e o de construção por cerca de $4/5$, e cada um dos pontos faz parte do conjunto de validação apenas uma das cinco vezes. Ao utilizar este tipo de divisão do conjunto de dados para teste, estamos fazendo o processo de validação cruzada *k-fold* descrito na Seção 2.3, sendo nesse caso $k = \frac{n}{5}$.

É importante citar que todos os atributos foram padronizados (entre 0 e 1), para evitar que diferentes ordens de grandeza interferissem nos resultados.

Para observar o comportamento dos métodos de classificação em dados reais, na presença de ruído no rótulo, introduzimos ruído em diferentes percentuais (10%, 20%, 30%, 40% e 50%) trocando os rótulos de pontos selecionados aleatoriamente de acordo apenas com a classe (rótulo) original. Dessa forma, ao colocar o ruído NCAR, serão trocados os rótulos de determinado percentual de pontos cujo rótulo original é 0 e do mesmo percentual de pontos cujo rótulo original é 1. No caso do ruído NAR, em uma parte dos testes foram trocados os rótulos de determinado percentual de pontos cujo rótulo original é 0 e na outra parte, cujo rótulo original é 1, de forma a analisarmos os conjuntos de dados com ruídos em cada uma das classes, separadamente.

A definição dos parâmetros dos métodos utilizados foi feita da mesma forma que fizemos para os conjuntos de dados simulados e está explicada na Seção 5.2. Dessa forma, estabelecemos parâmetros da melhor forma possível para obtermos os melhores desempenhos de todos

os métodos na geração do classificador. Além disso, nos testes em dados reais também é interessante observar o quanto é "normal"a acurácia da classificação variar em função de diferentes composições dos conjuntos de treinamento e de teste (já que estamos utilizando diferentes formações, segundo o método de validação cruzada já explicado) para cada método. A partir do percentual médio de acertos obtido em cada uma das composições (conjunto de treinamento e conjunto de teste) dos conjuntos de dados usados, e para cada percentual de ruído no rótulo implementado, foram observados os acertos de cada método na classificação dos elementos do conjunto de teste. A partir desses valores dos percentuais de acertos de classificação para cada método, foi possível calcular os desvios-padrão em relação à média, de forma que construimos uma tabela com o desvio-padrão calculado para cada percentual de ruído versus o método utilizado para a classificação, para cada conjunto de dados reais. Nesse caso, optamos por calcular a média desses desvios para cada conjunto de dados, considerando todos os métodos e todos os percentuais de ruído no rótulo, de forma a obtér um único valor de desvio a ser considerado "normal"para o percentual de acertos de métodos com o mesmo poder de desempenho. Ao apresentarmos os resultados relativos a cada um dos conjuntos de dados a seguir, colocaremos o desvio que foi considerado nas análises do desempenho dos métodos.

6.1 Os Conjuntos de Dados Reais

A seguir serão brevemente descritos os conjuntos de dados reais que foram utilizados para os testes de desempenho dos algoritmos de classificação.

6.1.1 *Ionosphere* [Lichman, 2013]

Os dados deste conjunto foram coletados em Goose Bay, Labrador, no Canadá, por um sistema composto por uma matriz de fases de antenas de alta frequência. "Bons"retornos de radar são aqueles que mostram evidência de algum tipo de estrutura na ionosfera. Retornos "ruins"são aqueles que não o fazem, seus sinais passam através da ionosfera. Os sinais recebidos foram processados usando uma função de autocorrelação cujos argumentos são o tempo de um pulso e o número do pulso. Houve 17 números de pulsos para o sistema de Goose Bay. Instâncias nesta base de dados são descritas por atributos que correspondem aos valores devolvidos por uma função a partir do sinal electromagnético complexo. O conjunto de dados é composto por 34 atributos contínuos e por uma resposta binária "bom"ou "ruim". Este banco tem 350 elementos. O total de respostas "bom"(0's) corresponde a aproximadamente 64% de todos os elementos do banco de dados e o restante, cerca de 36% são as respostas "ruim"(1's).

6.1.2 *Wisconsin Breast Cancer Dataset* [Lichman, 2013]

Os dados deste conjunto foram obtidos da Universidade de Wisconsin, Madison, pelo Dr. William H. Wolberg. Foram analisados casos de tumores de mama entre janeiro de 1989 e outubro de 1991. Para cada elemento do conjunto de dados há nove medidas (relativas a tamanho de célula, formato de célula, etc) que variam nos números inteiros entre 1 e 10, e uma resposta binária "maligno"ou "benigno". Este banco é composto por 680 elementos. O total de respostas

benignas (0's) corresponde a aproximadamente 65% de todos os elementos do banco de dados e o restante, cerca de 35% são as respostas malignas (1's).

6.1.3 *Wisconsin Diagnosis Breast Cancer (WDBC)* [Lichman, 2013]

As caracteristicas presentes nesses dados foram calculadas a partir de uma imagem digitalizada de um aspirado de agulha fina de uma massa de mama. Eles descrevem as características dos núcleos das células presentes na imagem. Para cada elemento do conjunto de dados há 30 atributos numéricos e uma resposta binária "maligno"ou "benigno". Este banco é composto por 565 elementos. O total de respostas benignas (0's) corresponde a aproximadamente 62% de todos os elementos do banco de dados e o restante, cerca de 38% são as respostas malignas (1's).

6.1.4 *Blood Transfusion Data* [Lichman, 2013]

Este banco contém dados de 748 doadores de sangue, selecionados aleatoriamente, do *Blood Transfusion Service Center* em Hsin-Chu City, na Tailândia. A cada 3 meses, este serviço passa seu ônibus em uma universidade na cidade para arrecadar doações de sangue. Estes dados incluem 4 atributos: meses desde a última doação, número de doações já realizadas, quantidade total de sangue doado e meses desde a primeira doação, e uma resposta binária que representa se o indivíduo doou sangue ou não em Março de 2007. O total de respostas negativas (0's) corresponde a 74% de todas as instâncias do banco e o total de respostas positivas (1's) a 26%.

6.1.5 *Mamography* [Lichman, 2013]

Atualmente, o método mais efetivo disponível para rastear câncer de mama é a mamografia. Todavia, a interpretação das mamografias leva a cerca de 70% de biópsias desnessárias com resultados benignos. Para reduzir este alto número de biópsias de mama desnecessárias, alguns métodos de diagnóstico por computador têm sido propostos. Estes sistemas buscam ajudar os médicos na decisão de fazer uma biópsia em uma lesão suspeita vista na mamografia ou, ao invés disso, de fazer um acampanhamento durante um certo período curto de tempo.

Este conjunto de dados pode ser utilizado para prever se a lesão de uma mamografia é benigna ou maligna a partir de 5 atributosreferentes à lesão e à paciente. O banco contém dados de 830 pacientes coletados no *Institute of Radiology of the University Erlangen-Nuremberg* entre 2003 e 2006. O total de respostas benignas (0's) corresponde a aproximadamente 39% de todos os elementos do banco de dados e o restante, cerca de 61% são as respostas malignas (1's).

A Tabela 6.1 resume as caracteristicas dos conjuntos de dados utilizados.

Com relação aos conjuntos de dados serem compostos de proporções semelhantes de elementos de cada classe de rótulo (balanceados), podemos observar pelos dados da Tabela 6.1 que nenhum dos conjuntos é muito desbalanceado. 4 conjuntos possuem entre 62% e 65% de elementos em cada uma das classes (e entre 35% e 38% na outra) e o conjunto *Blood Transfusion* é o mais desequilibrado, com aproximadamente $\frac{3}{4}$ dos elementos em uma classe e $\frac{1}{4}$ na

Conjunto	Elementos	Atributos	0's	1's
Ionosphere	350	34	64%	36%
Wisconsin Breast Cancer	580	9	65%	35%
Wisconsin Diagnosis Breast Cancer	565	30	62%	38%
Blood Transfusion	748	4	74%	26%
Mammography	830	5	39%	61%

Tabela 6.1 Resumo dos conjuntos de dados reais utilizados para avaliação dos métodos de classificação supervisionada

outra. Diante dessas composições, julgamos que a acurácia é uma boa medida para avaliar a qualidade dos classificadores nestes conjuntos de dados, não sendo muito indispensável utilizar outras medidas, como sensibilidade, especificidade e precisão. Mesmo assim, para termos mais ferramentas para julgar a qualidade dos métodos empregados, calculamos essas medidas para cada um dos métodos em cada conjunto de dados. Os resultados serão apresentados na próxima Seção.

6.2 Resultados

6.2.1 Acurácia

As Tabelas 6.2, 6.3 e 6.4 mostram os resultados em percentual médio de acertos de cada um dos métodos aplicados no conjunto de dados *Ionosphere*, conforme os testes descritos. Ao observar a primeira coluna das tabelas, temos os resultados da acurácia média na classificação para os conjuntos de dados sem a introdução voluntária de nenhum ruído no rótulo. Nesse caso, os métodos de melhor desempenho são LORC, Random LORC, Florestas Aleatórias e SVM.

Ao introduzir ruído no rótulo do tipo NCAR, no qual o ruído é colocado no mesmo percentual nas duas classes de rótulo, os resultados apresentados na Tabela 6.2 apontam que o único método cujo resultado está entre os melhores para todos os percentuais de ruído introduzidos é o Random LORC. Portanto, nesse caso, este foi o método que se mostrou mais robusto em relação a acurácia média na classificação para os dados do conjunto *Ionosphere* para este tipo de ruído. Logo atrás dele, o Florestas Aleatórias também se destacou, com o resultado estando entre os melhores para todos os percentuais de troca de rótulo até 30%. O SVM ficou entre os melhores para percentuais de ruído de até 20%, porém para os percentuais mais altos seu desempenho foi bastante afetado. E o LORC apareceu entre os melhores para os percentuais de 10% e 30%.

Quando o ruído introduzido foi do tipo NAR, no qual o ruído é colocado apenas em uma das classes de rótulo, os resultados apresentados nas Tabelas 6.3 e 6.4 destacam métodos robustos diferentes para cada caso (ruído na classe 0 ou na classe 1). Quando o ruído foi introduzido na classe de rótulos 0's, podemos observar que os métodos Random LORC e Forestas Aleatórias tiveram sua acurácia média entre as melhores para todos os percentuais de ruído, se mostrando os métodos mais robustos. Além destes, o LORC e o CART também se destacaram, se man-

tendo sempre entre os melhores desempenhos, exceto para o percentual de troca de rótulo de 20%. O LORCy também se mostrou robusto nesse caso, apresentando bons resultados para os percentuais mais altos de troca de rótulo. Já o SVM esteve entre os melhores para os percentuais mais baixos (até 20%). Por outro lado, quando o ruído foi introduzido na classe de rótulos 1's, o SVM se mostrou o método mais robusto, com resultados entre os melhores para todos os percentuais de troca de rótulo. Atrás dele, apareceram Florestas Aleatórias com bons resultados para percentuais de até 20% e o Random LORC apenas para percentual de até 10%.

	0%	10%	20%	30%	40%
LORC	0.897	0.874	0.806	0.791	0.666
LORCy	0.851	0.777	0.771	0.64	0.583
Random LORC	0.894	0.866	0.831	0.771	0.726
Random LORCy	0.803	0.751	0.751	0.626	0.597
Reg. Logística	0.869	0.857	0.803	0.8	**0.76**
CART	0.869	0.857	0.797	0.717	0.626
Flor. Aleatórias	0.926	0.911	0.86	**0.811**	0.694
SVM	**0.946**	**0.92**	**0.883**	0.626	0.606
kNN	0.874	0.794	0.78	0.734	0.7

Tabela 6.2 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NCAR no conjunto de dados de treinamento do modelo, para os dados do conjunto *Ionosphere*. Desvio-médio: 0.055

	0%	10%	20%	30%	40%
LORC	0.9	0.883	0.857	0.837	**0.8**
LORCy	0.851	0.843	0.806	0.811	0.789
Random LORC	0.894	0.883	0.877	0.84	0.789
Random LORCy	0.837	0.786	0.734	0.743	0.731
Reg. Logística	0.869	0.851	0.82	0.809	0.749
CART	0.869	0.903	0.803	0.826	0.766
Flor. Aleatórias	0.926	**0.923**	**0.914**	**0.846**	0.794
SVM	**0.931**	0.917	0.883	0.8	0.7
kNN	0.874	0.846	0.809	0.769	0.726

Tabela 6.3 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR ao trocar rótulos de pontos da classe 0 para 1 no conjunto de dados de treinamento do modelo, para os dados do conjunto *Ionosphere*. Desvio-médio: 0.041

As Tabelas 6.5, 6.6 e 6.7 mostram os resultados em percentual médio de acertos de cada um dos métodos aplicados no conjunto de dados *Wisconsin Breast Cancer Dataset*, conforme os testes descritos. Ao observar a primeira coluna das tabelas, temos os resultados da acurácia média na classificação para os conjuntos de dados sem a introdução voluntária de nenhum ruído no rótulo. Nesse caso, todos os métodos testados como classificadores apresentaram bom desempenho em relação a acurácia média na classificação, com resultados bem semelhantes.

	0%	10%	20%	30%	40%
LORC	0.897	0.86	0.811	0.797	0.751
LORCy	0.851	0.854	0.751	0.774	0.789
Random LORC	0.891	0.886	0.826	0.8	0.734
Random LORCy	0.814	0.817	0.789	0.731	0.731
Reg. Logística	0.869	0.863	0.831	0.84	0.789
CART	0.869	0.837	0.8	0.783	0.723
Flor. Aleatórias	0.926	0.909	0.9	0.857	0.826
SVM	**0.931**	**0.934**	**0.917**	**0.931**	**0.9**
kNN	0.877	0.84	0.843	0.854	0.866

Tabela 6.4 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR ao trocar rótulos de pontos da classe 1 para 0 no conjunto de dados de treinamento do modelo, para os dados do conjunto *Ionosphere*. Desvio-médio: 0.051

Portanto, para este conjunto de dados, sem ruído no rótulo, podemos dizer que todos os métodos foram igualmente eficientes.

Ao introduzir ruído no rótulo do tipo NCAR, no qual o ruído é colocado no mesmo percentual nas duas classes de rótulo, os resultados apresentados na Tabela 6.5 apontam que o único método cujo resultado está entre os melhores para todos os percentuais de ruído introduzidos é o Random LORC. Portanto, nesse caso, este foi o método que se mostrou mais eficiente em relação a acurácia média na classificação para os dados do conjunto *Wisconsin Breast Cancer Dataset*. Logo atrás dele, a Regresão Logística e o kNN também se destacaram, com seus resultados estando entre os melhroes para todos os percentuais de troca de rótulo até 30%. Podemos observar ainda que as duas variações da metodologia LORC que utilizam o rótulo na etapa de construção da AGM perdem desempenho rapidamente a medida que vai sendo introduzido ruído no rótulo.

Quando o ruído introduzido foi do tipo NAR, no qual o ruído é colocado apenas em uma das classes de rótulo, os resultados apresentados nas Tabelas 6.6 e 6.7 mostram que o método que obteve resultado em relação a acurácia média nas classificações entre os melhores para ambos os tipos de ruído e para todos os percentuais foi o SVM. No caso em a troca foi na classe de rótulos 0's, o Random LORC ficou empatado com o SVM, mas no caso da troca dos rótulos ser na classe de 1's, ele só esteve entre os melhores para percentuais mais baixos (até 20%). Neste segundo caso, Regressão Logística, CART, Florestas Aleatórias e kNN obtiveram melhores resultados para todos os percentuais de troca de rpotulo juntamente com o SVM. Destes, no outro caso em que a troca foi na classe de rótulo 0, os que ficaram melhores foram a Regressão Logística, o CART e o kNN, com resultados entre os melhores para percentuais de troca de rótulo de até 30%.

As Tabelas 6.8, 6.9 e 6.10 mostram os resultados em percentual médio de acertos de cada um dos métodos aplicados no conjunto de dados *Wisconsin Diagnosis Breast Cancer (WDBC)*, conforme os testes descritos. Ao observar a primeira coluna das tabelas, temos os resultados da acurácia média na classificação para os conjuntos de dados sem a introdução voluntária de nenhum ruído no rótulo. Nesse caso, quase todos os métodos testados como classificadores (com exceção do CART, que teve resultado apenas um pouco pior) apresentaram bom desempenho

	0%	10%	20%	30%	40%
LORC	0.947	0.949	0.912	0.859	0.819
LORCy	0.944	0.881	0.797	0.75	0.685
Random LORC	0.951	0.951	0.941	0.913	**0.829**
Random LORCy	0.965	0.921	0.84	0.712	0.6
Reg. Logística	0.963	**0.971**	**0.963**	**0.938**	0.782
CART	0.94	0.934	0.921	0.903	0.734
Flor. Aleatórias	**0.972**	0.959	0.928	0.851	0.681
SVM	0.968	0.969	0.946	0.828	0.762
kNN	0.971	0.969	0.937	0.926	0.776

Tabela 6.5 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NCAR no conjunto de dados de treinamento do modelo, para os dados do conjunto *Wisconsin Breast Cancer Dataset*. Desvio-médio: 0.039

	0%	10%	20%	30%	40%
LORC	0.947	0.965	0.956	0.95	0.843
LORCy	0.944	0.944	0.909	0.897	0.863
Random LORC	0.95	0.96	**0.969**	0.956	0.896
Random LORCy	0.965	0.925	0.851	0.749	0.751
Reg. Logística	0.963	0.965	0.957	0.934	0.841
CART	0.94	0.932	0.934	0.938	0.804
Flor. Aleatórias	**0.969**	0.949	0.94	0.91	0.769
SVM	0.968	0.962	0.963	**0.963**	**0.918**
kNN	0.965	**0.969**	0.957	0.962	0.743

Tabela 6.6 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR ao trocar rótulos de pontos da classe 0 para 1 no conjunto de dados de treinamento do modelo, para os dados do conjunto *Wisconsin Breast Cancer Dataset*. Desvio-médio: 0.041

em relação a acurácia média na classificação, com resultados bem semelhantes. Portanto, para este conjunto de dados, sem ruído no rótulo, podemos dizer que todos os métodos, com exceção do CART, foram igualmente eficientes.

Ao introduzir ruído no rótulo do tipo NCAR, no qual o ruído é colocado no mesmo percentual nas duas classes de rótulo, os resultados apresentados na Tabela 6.8 apontam que o único método cujo resultado está entre os melhores para todos os percentuais de ruído introduzidos é o Random LORC. Portanto, nesse caso, este foi o método que se mostrou mais eficiente em relação a acurácia média na classificação para os dados do conjunto *Wisconsin Breast Cancer Dataset*. Logo atrás dele, o kNN também se destaca, com seus resultados estando entre os melhores para todos os percentuais de troca de rótulo até 30%. Além desses, os métodos SVM e Florestas aleatórias tiveram resultados entre os melhores para percentuais de troca de rótulo de até 20%, mas acima deste percentual o SVM perde bastante o desempenho. Podemos observar ainda que as duas variações da metodologia LORC que utilizam o rótulo na etapa de construção da AGM perdem desempenho rapidamente a começa a ser introduzido ruído no rótulo.

Quando o ruído introduzido foi do tipo NAR, no qual o ruído é colocado apenas em uma

	0%	10%	20%	30%	40%
LORC	0.947	0.918	0.916	0.872	0.888
LORCy	0.944	0.878	0.85	0.81	0.766
Random LORC	0.951	0.938	0.916	0.885	0.841
Random LORCy	0.965	0.938	0.916	0.882	0.851
Reg. Logística	0.963	0.946	0.928	0.901	0.888
CART	0.94	0.937	0.932	0.925	0.882
Flor. Aleatórias	**0.972**	0.969	0.951	0.919	0.901
SVM	0.968	**0.974**	**0.954**	0.94	**0.918**
kNN	0.96	0.969	0.963	**0.947**	**0.918**

Tabela 6.7 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR ao trocar rótulos de pontos da classe 1 para 0 no conjunto de dados de treinamento do modelo, para os dados do conjunto *Wisconsin Breast Cancer Dataset*. Desvio-médio: 0.041

das classes de rótulo, os resultados apresentados nas Tabelas 6.9 e 6.10 mostram que o único método que obteve seu desempenho entre os melhores para todos os percentuais de troca de rótulo em ambas as classes (tanto trocando os rótulos de elementos com rótulos 1 quanto dos elementos com rótulo 0) foi o Random LORC. Portanto, para o conjunto de dados *Wisconsin Breast Cancer Dataset*, ele foi o melhor método de classificação, sendo ainda o mais robusto para todos os tipos de ruído no rótulo, tanto do tipo NCAR quanto do tipo NAR em ambas as classes de rótulo. No caso da troca de rótulo apenas na classe de 0's (Tabela 6.9), nenhum outro método teve o mesmo desemepnho em todos os percentuais de troca de rótulo. Logo atrás do Random LORC, ficaram LORC e kNN, cujos resultados ficaram entre os melhores para os percentuais de até 30%, e em seguida Florestas Aleatórias e SVM, para os percentuais de até 20%. Novamente, acima deste percentual o SVM perde muito desempenho. Já no caso da troca de rótulo ocorrer apenas na classe de 1's (Tabela 6.10), Florestas Aleatórias e kNN ficaram empatados com o Random LORC, apresentando melhores desempenhos para todos os percentuais de troca de rótulo introduzidos no conjunto de dados. Em seguida, o SVM se mostrou entre os melhores para percentuais de ruído de até 30% e o LORC de até 20%.

	0%	10%	20%	30%	40%
LORC	0.959	0.931	0.92	0.844	0.701
LORCy	0.959	0.894	0.834	0.708	0.688
Random LORC	0.961	0.961	0.942	0.892	**0.835**
Random LORCy	0.966	0.871	0.8	0.722	0.621
Reg. Logística	0.95	0.933	0.908	0.832	0.696
CART	0.926	0.927	0.862	0.75	0.609
Flor. Aleatórias	0.961	0.956	0.945	0.846	0.733
SVM	**0.979**	0.954	**0.949**	0.566	0.395
kNN	0.966	**0.963**	0.943	**0.896**	0.729

Tabela 6.8 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NCAR no conjunto de dados de treinamento do modelo, para os dados do conjunto *Wisconsin Diagnosis Breast Cancer (WDBC)*. Desvio-médio: 0.03

	0%	10%	20%	30%	40%
LORC	0.959	0.954	0.938	0.915	0.865
LORCy	0.959	0.942	0.929	0.901	0.865
Random LORC	0.958	0.95	0.952	**0.947**	**0.901**
Random LORCy	0.95	0.904	0.835	0.752	0.708
Reg. Logística	0.95	0.927	0.885	0.848	0.773
CART	0.926	0.913	0.848	0.788	0.685
Flor. Aleatórias	0.961	0.956	0.933	0.873	0.75
SVM	**0.979**	**0.958**	**0.954**	0.908	0.823
kNN	0.952	0.954	0.938	0.915	0.821

Tabela 6.9 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR ao trocar rótulos de pontos da classe 0 para 1 no conjunto de dados de treinamento do modelo, para os dados do conjunto *Wisconsin Diagnosis Breast Cancer (WDBC)*. Desvio-médio: 0.033

	0%	10%	20%	30%	40%
LORC	0.959	0.943	0.924	0.885	0.874
LORCy	0.959	0.908	0.867	0.811	0.77
Random LORC	0.959	0.938	0.919	0.91	0.892
Random LORCy	0.965	0.924	0.908	0.873	0.88
Reg. Logística	0.95	0.933	0.917	0.89	0.86
CART	0.926	0.912	0.92	0.906	0.869
Flor. Aleatórias	0.965	0.954	0.935	**0.926**	0.906
SVM	**0.979**	**0.961**	**0.943**	0.919	0.881
kNN	0.965	0.958	0.94	**0.926**	**0.913**

Tabela 6.10 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR ao trocar rótulos de pontos da classe 1 para 0 no conjunto de dados de treinamento do modelo, para os dados do conjunto *Wisconsin Diagnosis Breast Cancer (WDBC)*. Desvio-médio: 0.024

As Tabelas 6.11, 6.12 e 6.13 mostram os resultados em percentual médio de acertos de cada um dos métodos aplicados no conjunto de dados *Blood Transfusion Data*, conforme os testes descritos. Ao observar a primeira coluna das tabelas, temos os resultados da acurácia média na classificação sem a introdução voluntária de nenhum ruído no rótulo. Nesse caso, quase todos os métodos testados como classificadores apresentaram bom desempenho em relação a acurácia média na classificação (exceto o Random LORCy e o kNN), com resultados bem semelhantes. Portanto, para este conjunto de dados, sem ruído no rótulo, podemos dizer que todos os métodos, com exceção do Random LORCy e do kNN, foram igualmente eficientes.

Ao introduzir ruído no rótulo do tipo NCAR, no qual o ruído é colocado no mesmo percentual nas duas classes de rótulo, os resultados apresentados na Tabela 6.11 apontam que os únicos métodos cujos resultados estão entre os melhores para todos os percentuais de ruído introduzidos são o Random LORC e a Regressão Logística. Portanto, nesse caso, estes foram os métodos que se mostraram mais eficiente em relação a acurácia média na classificação para os dados do conjunto *Blood Transfusion Data*. Logo atrás deles, o LORC também se destacou, com seus resultados entre os melhores para todos os percentuais, exceto 20%. Além destes,

SVM e CART se mostraram eficiente para percentuais mais baixos de troca de rótulos, estando entre os melhores até o percentual de 20%.

Quando o ruído introduzido foi do tipo NAR, no qual o ruído foi colocado apenas em uma das classes de rótulo, os resultados apresentados nas Tabelas 6.12 e 6.13 mostram que o único método que obteve seu desempenho entre os melhores para todos os percentuais de troca de rótulo em ambas as classes (tanto trocando os rótulos de elementos com rótulos 1 quanto dos elementos com rótulo 0) foi o Random LORC. Portanto, para o conjunto de dados *Blood Transfusion Data*, ele foi o melhor método de classificação, sendo ainda o mais robusto para todos os tipos de ruído no rótulo, tanto do tipo NCAR quanto do tipo NAR. No caso da troca de rótulo apenas na classe de 0's (Tabela 6.12), nenhum outro método teve o mesmo desemepnho em todos os percentuais de troca de rótulo. Logo atrás do Random LORC, ficaram LORCy e CART, cujos resultados ficaram entre os melhores para quase todos os percentuais de troca de rótulo, exceto um deles (20% e 40%, respectivamente).Em seguida, temos que a Regressão Logística e o SVM apresentaram resultados entre os melhores para percentuais mais baixos, de até 20%. Já no caso da troca de rótulo ocorrer apenas na classe de 1's (Tabela 6.13), o desempenho da maior parte dos métodos foi bem parecida para todos os percentuais analisados.O único método que se destacou por apresentar resultados piores que os demais para este tipo de ruído foi o Random LORCy.

	0%	10%	20%	30%	40%
LORC	0.778	0.745	0.734	0.703	**0.658**
LORCy	0.759	0.743	0.709	0.664	0.631
Random LORC	0.77	0.761	0.749	0.693	0.634
Random LORCy	0.717	0.668	0.631	0.557	0.528
Reg. Logística	0.765	0.763	**0.771**	**0.723**	0.632
CART	0.765	**0.773**	0.751	0.646	0.539
Flor. Aleatórias	0.769	0.737	0.734	0.607	0.559
SVM	**0.783**	0.767	0.755	0.679	0.587
kNN	0.75	0.731	0.695	0.631	0.598

Tabela 6.11 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NCAR no conjunto de dados de treinamento do modelo, para os dados do conjunto *Blood Transfusion Data*. Desvio-médio: 0.031

As Tabelas 6.14, 6.15 e 6.16 mostram os resultados em percentual médio de acertos de cada um dos métodos aplicado no conjunto de dados *Mamography*, conforme os testes descritos. Ao observar a primeira coluna das tabelas, temos os resultados da acurácia média na classificação para os conjuntos de dados sem a introdução voluntária de nenhum ruído no rótulo. Nesse caso, quase todos os métodos testados como classificadores apresentaram bom desempenho em relação a acurácia média na classificação (exceto LORCy, Random LORCy e kNN), com resultados bem semelhantes. Portanto, para este conjunto de dados, sem ruído no rótulo, podemos dizer que todos os métodos, com exceção dos 3 citados, foram igualmente eficientes.

Ao introduzir ruído no rótulo do tipo NCAR, no qual o ruído é colocado no mesmo percentual nas duas classes de rótulo, os resultados apresentados na Tabela 6.14 apontam que os métodos cujos resultados estão entre os melhores para todos os percentuais de ruído introdu-

	0%	10%	20%	30%	40%
LORC	0.778	0.742	0.715	**0.651**	0.503
LORCy	0.759	0.75	0.701	**0.651**	**0.56**
Random LORC	0.769	0.761	0.737	0.65	0.535
Random LORCy	0.713	0.668	0.567	0.537	0.456
Reg. Logística	0.765	**0.777**	0.734	0.61	0.352
CART	0.765	0.769	0.731	0.64	0.433
Flor. Aleatórias	0.767	0.745	0.711	0.58	0.42
SVM	**0.782**	0.754	**0.751**	0.596	0.412
kNN	0.751	0.722	0.648	0.545	0.402

Tabela 6.12 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR ao trocar rótulos de pontos da classe 0 para 1 no conjunto de dados de treinamento do modelo, para os dados do conjunto *Blood Transfusion Data*. Desvio-médio: 0.03

	0%	10%	20%	30%	40%
LORC	0.782	**0.777**	0.762	0.769	0.77
LORCy	0.759	0.762	0.766	0.769	0.762
Random LORC	0.767	0.774	0.766	0.773	0.77
Random LORCy	0.713	0.717	0.763	0.73	0.747
Reg. Logística	0.77	0.771	**0.774**	0.774	**0.771**
CART	0.771	0.761	0.773	0.763	0.77
Flor. Aleatórias	0.774	0.77	**0.774**	**0.779**	0.767
SVM	**0.789**	0.755	0.761	0.767	0.759
kNN	0.749	0.765	0.763	0.762	0.758

Tabela 6.13 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR ao trocar rótulos de pontos da classe 1 para 0 no conjunto de dados de treinamento do modelo, para os dados do conjunto *Blood Transfusion Data*. Desvio-médio: 0.034

zidos são Random LORC, Regressão Logística, CART e Florestas Aleatórias. Portanto, nesse caso, estes foram os métodos que se mostraram mais eficiente em relação a acurácia média na classificação para os dados do conjunto *Mamography*. Os demais métodos tiveram desempenho bem aquém destes.

Quando o ruído introduzido foi do tipo NAR, no qual o ruído foi colocado apenas em uma das classes de rótulo, os resultados apresentados nas Tabelas 6.15 e 6.16 mostram que o único método que obteve seu desempenho entre os melhores para todos os percentuais de troca de rótulo em ambas as classes (tanto trocando os rótulos de elementos com rótulos 1 quanto dos elementos com rótulo 0) foi o CART. Portanto, para o conjunto de dados *Mamography*, ele foi o melhor método de classificação, sendo ainda o mais robusto para todos os tipos de ruído no rótulo, tanto do tipo NCAR quanto do tipo NAR em ambas as classes de rótulo. No caso da troca de rótulo apenas na classe de 0's (Tabela 6.15), o Random LORC também teve o mesmo desempenho do CART, estando entre os melhores em todos os percentuais de troca de rótulo. Atrás do CART e do Random LORC, aparecem Florestas Aleatórias e LORC, cujos resultados ficaram entre os melhores para quase todos os percentuais de troca de rótulo, exceto 40%, para Florestas Aleatórias, e 10% e 20%, para o LORC). Já no caso da troca de rótulo

ocorrer apenas na classe de 1's (Tabela 6.16), os métodos de melhor desempenho em todos os percentuais de troca de rótulo foram CART e Florestas Aleatórias. Em seguida, aparecem o LORC e o SVM, que não ficaram entre os melhores para apenas um dos percentuais de ruído (30% e 20%, respectivamente). Random LORC e Regressão Logística apresentaram-se entre os melhores para percentuais de troca de rótulo de até 20%.

	0%	10%	20%	30%	40%
LORC	0.817	0.778	0.801	0.753	0.723
LORCy	0.742	0.66	0.661	0.607	0.587
Random LORC	0.817	0.816	0.801	0.773	0.758
Random LORCy	0.801	0.708	0.649	0.624	0.56
Reg. Logística	0.831	0.835	0.819	0.798	0.745
CART	**0.843**	**0.84**	**0.828**	**0.799**	**0.772**
Flor. Aleatórias	0.825	0.828	0.81	0.782	0.757
SVM	0.825	0.807	0.799	0.614	0.74
kNN	0.795	0.777	0.782	0.727	0.645

Tabela 6.14 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NCAR no conjunto de dados de treinamento do modelo, para os dados do conjunto de Mamografia. Desvio-médio: 0.028

	0%	10%	20%	30%	40%
LORC	0.813	0.786	0.787	0.771	0.733
LORCy	0.73	0.722	0.72	0.701	0.719
Random LORC	0.811	0.804	0.808	**0.777**	**0.746**
Random LORCy	0.776	0.747	0.731	0.67	0.67
Reg. Logística	0.813	0.819	0.793	0.72	0.643
CART	**0.841**	**0.834**	0.812	0.765	0.673
Flor. Aleatórias	0.825	0.831	**0.82**	0.772	0.665
SVM	0.836	0.804	0.787	0.746	0.643
kNN	0.787	0.781	0.77	0.708	0.678

Tabela 6.15 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR (troca de 0 para 1) no conjunto de dados de treinamento do modelo, para os dados do conjunto de Mamografia. Desvio-médio: 0.031

	0%	10%	20%	30%	40%
LORC	0.819	0.81	0.796	0.747	0.769
LORCy	0.73	0.695	0.654	0.636	0.62
Random LORC	0.814	0.807	0.799	0.757	0.719
Random LORCy	0.787	0.769	0.74	0.693	0.718
Reg. Logística	0.831	0.831	0.816	0.778	0.76
CART	0.843	0.837	0.825	0.822	0.788
Flor. Aleatórias	0.828	0.833	0.823	0.805	0.777
SVM	0.829	0.82	0.787	0.81	0.796
kNN	0.798	0.763	0.763	0.734	0.694

Tabela 6.16 Percentual médio de acertos dos métodos de classificação supervisionada para cada percentual de troca de rótulo tipo NAR (troca de 1 para 0) no conjunto de dados de treinamento do modelo, para os dados do conjunto de Mamografia. Desvio-médio: 0.032

6.2.2 Sensibilidade, Especificidade e Precisão

Nesta parte do trabalho serão apresentadas e discutidas as medidas de sensibilidade, especificidade e precisão de cada um dos métodos. Limitaremos a apresentar os resultados obtidos nos conjuntos de dados reais sem a introdução de ruído no rótulo. As Tabelas 6.17, 6.18 e 6.19 mostram estes valores.

	Ionosphere	*WBC*	*WDBC*	*Blood Transfusion*	*Mamography*
LORC	0.95	0.887	0.904	0.297	0.736
LORCy	0.977	0.878	0.9	0.183	0.587
Random LORC	0.968	0.891	0.904	0.183	0.719
Random LORCy	0.986	0.958	0.931	0.331	0.796
Reg. Logística	0.941	0.933	0.933	0.109	0.821
CART	0.928	0.912	0.89	0.349	0.779
Flor. Aleatórias	0.955	0.975	0.938	0.303	0.821
SVM	0.986	0.971	0.957	0.303	0.825
kNN	0.982	0.929	0.928	0.149	0.794

Tabela 6.17 Sensibilidade Média dos Métodos de Classificação nos Conjuntos de Dados Reais sem Ruído no Rótulo

Para o conjunto *Ionosphere*, os valores de sensibilidade de todos os métodos foram muito bons, sendo que os únicos que ficaram abaixo de 0.95 foram Regressão Logística e CART. Já para a especificidade, os resultados não foram tão bons. Apenas 3 métodos ficaram acima de 0.8, LORC, Florestas Aleatórias e SVM. Os piores resutados para especificidade foram do LORCy e Random LORC, que ficaram abaixo de 0.7. Em relação à precisão, todos os valores ficaram acima de 0.8, exceto o Random LORCy (0.779). Os maiores valores, acima de 0.9, foram do SVM e do Florestas Aleatórias, seguidos pelo LORC, cuja precisão média de 0.894.

Para o conjunto *Wisconsin Breast Cancer Dataset*, os valores de sensibilidade da maior parte dos métodos foi acima de 0.9, exceto para o LORC, o LORCy e o Random LORC. De toda forma, estes também não obtiveram resultados muito inferiores, ficando todos acima de 0.87. Em relação a especificidade, todos os métodos tiveram resultados muito bons, acima de

	Ionosphere	*WBC*	*WDBC*	*Blood Transfusion*	*Mamography*
LORC	0.805	0.98	0.992	0.935	0.86
LORCy	0.633	0.98	0.994	0.944	0.864
Random LORC	0.742	0.984	0.989	0.947	0.876
Random LORCy	0.516	0.968	0.966	0.845	0.794
Reg. Logística	0.742	0.98	0.961	0.974	0.841
CART	0.766	0.955	0.947	0.9	0.904
Flor. Aleatórias	0.875	0.971	0.975	0.923	0.827
SVM	0.875	0.966	0.992	0.937	0.839
kNN	0.711	0.977	0.966	0.937	0.797

Tabela 6.18 Especificidade Média dos Métodos de Classificação nos Conjuntos de Dados Reais sem Ruído no Rótulo

	Ionosphere	*WBC*	*WDBC*	*Blood Transfusion*	*Mamography*
LORC	0.894	0.959	0.984	0.584	0.831
LORCy	0.822	0.959	0.989	0.5	0.803
Random LORC	0.867	0.968	0.984	0.516	0.845
Random LORCy	0.779	0.942	0.966	0.397	0.784
Reg. Logística	0.864	0.961	0.933	0.559	0.829
CART	0.873	0.912	0.89	0.517	0.884
Flor. Aleatórias	0.93	0.947	0.957	0.546	0.817
SVM	0.932	0.939	0.985	0.596	0.829
kNN	0.855	0.957	0.975	0.419	0.786

Tabela 6.19 Precisão Média dos Métodos de Classificação nos Conjuntos de Dados Reais sem Ruído no Rótulo

0.95, com destaque para LORC, LORCy, Random LORC e Regressão Logística, que ficaram acima de 0.98. Para a precisão, os resultados também foram todos altos, acima de 0.9. O menor valor foi do CART (0.912), seguido pelo SVM (0.939), Random LORCy (0.942) e Florestas Aleatórias (0.947), cujos valores médios da precisão ficaram abaixo de 0.95.

Para o conjunto *Wisconsin Diagnosis Breast Cancer*, os valores de sensibilidade também foram bons, sendo que apenas o CART ficou abaixo de 0.9, mas bem próximo desse valor (0.89). O método com sensibilidade mais alta foi o SVM. A especificidade de todos os métodos ficou acima de 0.95, exceto para o CART. Mas mesmo para ele, o valor foi bem próximo aos demais métodos (0.947). Com relação a precisão, apenas o CART (0.89) e a Regressão Logística (0.933) ficaram abaixo de 0.95. Os maiores valores, acima de 0.98, foram alcançados pelo LORCy (0.989), SVM (0.985)e LORC e Random LORC empatados (0.984).

Para o conjunto *Blood Transfusion*, os valores de sensibilidade foram muito baixos para todos os métodos, estando sempre abaixo de 0.35. Observe que este é o conjunto de dados mais desbalanceado entre os que foram analisados e o único em que o percentual de elementos com rótulo "1"é menor que o de elementos com rótulo "0". Ou seja, há uma diferença razoável na quantidade de elementos de cada rótulo, sendo que os de rótulo "1"são minoria, podendo gerar certa dificuldade em sua correta classificação. A sensibilidade mede o percentual de elementos entre todos os de rótulo "1"que foram classificados corretamente. Com esses valores

tão baixos, podemos perceber que nenhum dos métodos foi eficiente para detectar essa minoria de elementos com rótulo "1", o que pode ser um problema. Mesmo assim, temos alguns um pouco melhores que outros. Os piores métodos foram LORCy, Random LORC, Regressão Logística e kNN, com resultados abaixo de 0.2. Em relação aos valores da especificidade, os métodos em geral apresentaram bons resultados. O único valor abaixo de 0.9 foi do Random LORCy (0.845). O melhor resultado foi da Regressão Logística, seguida pelo Random LORC e pelo LORCy. Os valores da precisão média também foram baixo para todos os métodos, sendo que os piores, menores que 0.5, foram Random LORCy (0.397) e kNN (0.419). Os demais valores ficaram todos dentro do intervalo de 0.5 a 0.6, sendo que os melhores valores foram referentes ao SVM (0.596), LORC (0.584) e Regressão Logística (0.559).

Para o conjunto *Mamography*, os valores de sensibilidade ficaram todos dentro da faixa de 0.7 a 0.83, exceto o LORCy, cujo valor foi 0.587. Os maiores valores foram, respectivamente, dos métodos SVM, Regressão Logística e Florestas Aleatórias (empatados) e Random LORCy. Em relação a especificidade, todos os método ficaram dentro do intervalo de 0.8 a 0.9, exceto o Random LORCy (0.794) e o kNN(0.797), que ficaram um pouco abaixo, e o CART (0.904), que ficou um pouco acima. Depois do CART, os melhores valores foram do Random LORC, LORCy e LORC, respectivamente. Para a precisão, a maior parte dos métodos obteve valores entre 0.8 e 0.9, exceto Random LORCy (0.784) e kNN (0.786), que foram inferiores nesse quesito. Os valores mais altos foram do CART (0.884), Random LORC (0.584) e Regressão Logística (0.559), que ficaram acima de 0.55.

6.3 Comentários

É interessante pensar que foram realizados testes com conjuntos de dados disponíveis e já utilizados anteriormente em testes de métodos de classificação supervisionada. Isso poderia ser um indício de que os bancos teriam sido usados como exemplos de sucesso para algum dos métodos que estamos comparando ao que criamos. Mesmo assim, foi observado que as variações do método proposto obtiveram resultados no mínimo compatíveis com os demais métodos, em todos os conjuntos de dados. Mais uma vez o nosso método se mostrou bastante competitivo com os demais. Além disso, podemos observar uma regularidade dos resultados, de forma que em nenhum dos casos apresentados, houve um desempenho consideravelmente ruim do nosso método em relação aos outros testados.

Mais especificamente discutindo cada caso, podemos observar que para os conjuntos de dados sem ruído no rótulo, tivemos vários métodos com bons desempenhos. Os que ficaram entre os melhores para todos os conjuntos de dados foram LORC, Random LORC, SVM e Florestas Aleatórias. Baseados nos testes desenvolvidos, podemos dizer então, que para estes conjuntos de dados reais, estes 4 métodos seriam os mais adequados ao tratar dados sem ruído no rótulo.

Podemos observar que no caso do conjunto *Blood Transfusion Data*, o LORCy também teve bom desempenho em relação a acurácia média das classificações. Talvez essa diferença de desempenho desta variação do LORC neste conjunto de dados em relação aos demais possa ser explicada pelo menor número de atributos dos elementos que o compõem. Isso pode ser um indício de que a variação que utiliza o rótulo na etapa de construção da AGM seja bom para

lidar com dados sem ruído no rótulo quando o número de atributos é pequeno.

Ainda sobre os dados sem ruído no rótulo, podemos observar que geralmente, os métodos que têm a melhor acurácia são os mesmo que têm os melhores valores de sensibilidade, especificidade e precisão. Porém isso nem sempre acontece, especialmente quando o conjunto de dados analisado é desbalanceado em relação à proporção de elementos em cada classe de rótulo. Entre os conjuntos que analisamos nessa seção, o que obteve diferenças consideráveis entre os métodos de melhor desempenho em relação à acurácia e em relação às outras medidas de desempenho foi o *Blood Transfusion*. Conforme comentamos anteriormente, este é o mais desbalanceado dos conjuntos de dados reais. Se observarmos apenas os valores da acurácia média, temos que os únicos métodos que ficam fora dos melhores são o Random LORCy e o kNN. Então, se o objetivo da análise for classificar corretamente a maior parte dos indivíduos em potenciais doadores ou não, estes métodos deveriam ser evitados. Porém se o objetivo da análise é classificar corretamente o maior percentual possível dos potenciais doadores (classificar corretamente as intâncias positivas, com rótulo "1"), que formam a classe com minoria de instâncias no conjunto de dados, o ideal seria analisar a sensibilidade dos métodos. Nesse caso, nenhum dos métodos apresentou desempenho considerado bom, mas entre os que ficaram melhores aparece o Random LORCy. Ou seja, mesmo não estando entre os métodos de maior acurácia, este método está entre os de melhor sensibilidade. Portanto, também é importante saber qual a resposta a ser encontrada com a análise para verificar o melhor método de classificação a ser escolhido.

Ao passarmos para uma visão geral dos resultados para os conjuntos de dados com ruído no rótulo, pode ser complicado generalizar. Assim como já discutimos anteriormente, não existe um método que sempre é melhor que os outros, independente das características do problema a ser analisado. De qualquer maneira, tentaremos tirar algumas conclusões mais gerais, com base nos conjuntos de dados reais que temos em mãos para realizar as análises dos algoritmos. Alguns itens podem ser observados a partir dos resultados, considerando cada tipo de ruído no rótulo:

- Para os conjuntos de dados com ruído do tipo NCAR, o Random LORC esteve entre os melhores métodos em relação à acurácia média das classificações em todos os conjuntos de dados. Depois dele, alguns métodos que apresentaram bons resultados foram Regressão Logística e Florestas Aleatórias, sendo que quase sempre eles tiveram desempenho aquém do Random LORC. Considerando os 3 melhores (incluindo mais, caso tenham desempenhos semelhantes), cada um destes 2 métodos teve bom desempenho em 3 dos 5 conjuntos de dados.

- Ainda para os conjuntos com ruído NCAR, os piores desempenhos, em geral, foram apresentados pelas variações do LORC que utilizam o rótulo na etapa de construção da AGM (LORCy e Random LORCy), que não ficaram entre os 3 melhores (incluindo mais, caso tenham desempenhos semelhantes) para nenhum dos conjuntos de dados, seguidos pelo CART e SVM, que ficaram entre os 3 melhores para apenas 1 dos 5 conjuntos de dados.

- Para os conjuntos de dados com ruído do tipo NAR trocando rótulos 0 para 1, o Random LORC foi o melhor método em relação à acurácia média das classificações em todos

os conjuntos de dados. Depois dele, os melhores métodos, no geral, foram o LORC e o SVM, que ficaram entre os 3 melhores (incluindo mais, caso tenham desempenhos semelhantes) para 4 dos 5 conjuntos de dados. Além deles, Regressão Logística, CART e SVM ficaram entre os 3 melhores para 3 conjuntos de dados cada um.

- Ainda para os conjuntos com ruído NAR trocando rótulos 0 para 1, o pior desempenho, em geral, foi apresentado pelo Random LORCy, que não ficou entre os 3 melhores (incluindo mais, caso tenham desempenhos semelhantes) para nenhum dos conjuntos de dados, seguido pelo LORCy e kNN, que ficaram entre os 3 melhores para apenas 2 dos 5 conjuntos de dados.

- Para os conjuntos de dados com ruído do tipo NAR trocando rótulos 1 para 0, 3 métodos estiveram entre os melhores em relação à acurácia média das classificações em todos os conjuntos de dados: Random LORC, SVM e Florestas Aleatórias. Depois deles, tendo ficado entre os 3 melhores (incluindo mais, caso tenham desempenhos semelhantes) para 4 conjuntos de dados, está o LORC. Além deles, Refressão Logística, CART e kNN ficaram entre os 3 melhores para 3 conjuntos de dados cada um.

- Ainda para os conjuntos com ruído NAR trocando rótulos 1 para 0, o pior desempenho, em geral, foi apresentado pelo LORCy, que ficou entre os 3 melhores (incluindo mais, caso tenham desempenhos semelhantes) para apenas 1 dos conjuntos de dados, seguido pelo Random LORCy, que ficou entre os 3 melhores para apenas 2 dos 5 conjuntos de dados.

Podemos observar que o método Random LORC é o que se mostrou mais estável, obtendo resultados muito bons em todos os conjuntos de dados e para todos os tipos de ruído no rótulo. Alguns outros métodos, como Florestas Aleatórias (principalmente), SVM, LORC e Regressão Logística, também obtiveram bons desempenhos em uma boa parte dos conjuntos de dados, mas nenhum foi tão bom para todas as situações quanto o Random LORC. Portanto, com base nos testes realizados nos dados reais, podemos dizer que o Random LORC é um método robusto para dados com ruído no rótulo, flexível em relação às características de diferentes conjuntos de dados, que pode ser utilizado com sucesso em diferentes problemas de classificação.

As variações da metodologia LORC que utilizam o rótulo na etapa da construção da AGM (LORCy e Random LORCy), conforme sugerido na descrição metodológica, confirmaram que não são boas opções quando os conjuntos de dados analisados têm ruído no rótulo. Eles ficaram sempre entre os métodos de pior desempenho, para todos os conjuntos de dados e todos os tipos de ruído no rótulo. Dessa forma, baseados nos dados reais analisados, podemos dizer que essas duas variações não são adequadas para classificação de dados nos quais existe suspeita de ruído no rótulo. Ao tratar dos dados sem ruído no rótulo, essas duas variações obtiveram bons resultados para alguns dos conjuntos de dados e não obtiveram o mesmo sucesso para outros, sendo que para a maior parte eles também tiveram desempenho inferior ao LORC e ao Random LORC.

CAPÍTULO 7

Conclusão

Este trabalho apresentou uma nova metodologia para classificação de dados binários, chamada *Label Noise Robust Classification Method* ou simplesmente LORC. O objetivo principal do método era o de ser um bom classificador para conjuntos de dados com ruído no rótulo e de ser um método flexível, podendo ser utilizado em diversos formatos de conjuntos de dados.

Foram desenvolvidas 4 variações do método, sendo que duas delas utilizam os rótulos na etapa de construção da AGM (LORCy e Random LORC) e outras duas não (LORC e Random LORC) e duas delas utilizam amostras Bootstrap para gerar vários classificadores e apartir do resultado da maioria deles, classificar novas instâncias (Random LORC e Random LORCy). Dessa forma, as 4 variações do método combinam de todas as formas possíveis as duas idéias.

A partir das definições feitas e dos lemas e teoremas demonstrados, foi comprovada a eficiência da metodologia para determinados tipos de conjuntos de dados, os compostos pelo que definimos como Clusters Compactos. Mostramos que o método realmente funciona, explorando a fundo suas caracteríticas e propriedades matemáticas, inclusive para conjuntos de dados com ruído no rótulo do tipo NCAR e NAR.

Foram realizados diversos testes de desempenho das variações do LORC propostas, e os resultados foram comparados com outros métodos de classificação popularmente utilizados e de eficiências comprovadas na literatura. A partir dos resultados obtidos, foi possível tirarmos várias conclusões a respeito dos métodos, tanto positivamente quanto negativamente, com a detecção de situações em que o método não é eficiente e de características que precisam ser melhoradas. Vamos detalhar um pouco mais destas conclusões:

- A variação do LORC que utiliza o rótulo na etapa de construção da AGM foi criada com o objetivo de lidar com alguns casos nos quais o método original não apresentava bons resultados, como por exemplo o caso extremo do conjunto de dados formado por Clusters Rotulados Complexos. Este objetivo parece ter sido cumprido, conforme os resultados apresentados para o conjunto de dados $C8$, no Capítulo 5. Além disso, essa variação também apresentou resultados muito bons para a maioria dos conjuntos de dados simulados sem ruído no rótulo, mas não mostrou o mesmo desempenho nos dados reais. Provavelmente isso ocorreu em função da quantidade de atributos das instâncias em cada conjunto de dados. Faz sentido pensar que nos casos em que o número de atributos é pequeno, ao acrescentarmos o rótulo como mais um atributo na etapa de construção da AGM faz com que a informação acrescida seja mais relevante do que quando o númerod de atributos é maior. Nos dados simulados, quase sempre havia só 2 atributos, enquanto nos dados reais essa quantidade sempre era maior. Portanto, o LORCy pode ser melhor que os demais para conjuntos de dados cujos atributos são poucos e sem ruído no rótulo.

- A variação do LORC que utiliza amostras Bootstrap foi desenvolvida com objetivo de amenizar os danos causados por conjuntos de dados com ruído no rótulo. A partir dos resultados dos testes com conjuntos de dados simulados e, principalmente, com dados reais, podemos concluir que o objetivo foi cumprido com sucesso. O Random LORC não apenas duperou as demais variações do LORC, como foi melhor que todos os demais métodos comparados. Ele se mostrou um método robusto para ruído no rótulo, apresentando bons resultados até mesmo para altos percentuais de rótulos trocados e para ambos os tipos de ruído NCAR e NAR. O método ainda se mostrou flexível em relação aos diversos formatos de conjuntos de dados, obtendo sucesso na grande maioria dos conjuntos testados.

- Até hoje não temos um método de classificação que seja o melhor para todos os conjuntos de dados a serem analisados. Portanto, é extremamente importante estudar bem os dados e a pergunta a ser respondida com a análise antes de escolher o método a ser utilizado.

- Essa análise preliminar do problema também é importante para sabermos quais as medidas de desempenho mais adequadas para avaliar os possíveis métodos de classificação. Conforme comentamos no Capitulo 2 e vimos nos testes de dados reais no Capítulo 6, mais especificamente para o conjunto de dados *Blood Transfusion*, em certas situações, especialmente quando o conjunto de dados é desbalanceado em relação ao número de elementos em cada classe, avaliar somente a acurácia dos métodos pode levar a escolhar ruins. Em casos deste tipo, pode ser interessante avaliar também medidas como sensibilidade, especificidade e precisão.

- No caso de conjuntos de dados balanceados em relação ao número de elementos em cada classe, a acurácia é uma boa medida de qualidade dos classificadores. Nesses casos, observamos pelos resultados do Capítulo 6 que o método que obteve o melhor desempenho geral, considerando todos os conjuntos de dados testados, foi o Random LORC. Portanto, concluímos que este método é a melhor escolha caso a análise a ser feita seja de um conjunto de dados balanceado.

- Foram observados dois pontos fracos da metodologia LORC. Um deles é a ineficiência que foi demonstrada para o conjunto de dados simulados $C6$, no qual havia muitas variáveis de ruído. Este resultado nos trás a suspeita de que, no caso de um determinado conjunto de dados com essa característica, nenhuma das variações do LORC seria adequada. Para que possamos ter essa conclusão com certeza, seria interessante desenvolvermos mais testes com conjuntos de dados (simulados e, principalmente reais) que tenham essa caracterítica.

- O segundo ponto fraco é o tempo de processamento gasto pelo LORC, especialmente pelas variações *Random*. Este tempo cresce muito à medida que o número de instâncias no conjunto de treinamento aumenta. Portanto, caso o método seja aplicado em um conjunto de dados muito grande (como dados de genética, por exemplo), este tempo de processamento pode ser bastante problemático. Algumas modificações no algoritmo podem ser realizadas para torná-lo mais eficiente. Em breve, pretendemos implementá-las, a fim de tornar o método ainda mais abrangente.

Enfim, após todo o desenvolvimento e as análises realizadas, temos uma metodologia construída e consolidada, que apresenta resultados extremamente satisfatórios dentro dos objetivos propostos, superando em muitos casos outros métodos de sucesso já existentes. Podemos concluir que os objetivos propostos foram cupridos com êxito.

APÊNDICE A

Testes Para Dados Simulados Sem ruído no Rótulo

Tabelas com os resultados dos testes variando a quantidade de pontos no conjunto de treinamento do modelo, para cada conjunto de dados simulados.

	50	100	150	200	250
LORC	0.8506	0.8909	0.9336	0.9464	0.9759
LORCy	0.9492	0.9722	0.989	0.9922	0.9983
Random LORC	0.9058	0.9118	0.9495	0.9626	0.9822
Random LORCy	0.952	0.9783	0.9921	0.9932	0.9972
Reg. Logística	0.615	0.5619	0.5825	0.6303	0.6271
CART	0.6734	0.772	0.8637	0.8863	0.9053
Flor. Aleatórias	0.8908	0.9387	0.9744	0.9805	0.981
SVM	0.9784	0.9957	0.9985	0.0.997	0.9995
kNN	0.9473	0.9737	0.9895	0.9919	0.9981

Tabela A.1 Percentual médio de acertos dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo para o conjunto de dados 1

	50	100	150	200	250
LORC	0.8652	0.9199	0.9699	0.9908	0.9996
LORCy	0.9925	0.9991	0.9998	1	1
Random LORC	0.9269	0.972	0.9905	0.9975	0.9999
Random LORCy	0.9679	0.9966	0.9984	0.9993	0.9998
Reg. Logística	0.5575	0.5213	0.515	0.5162	0.5365
CART	0.8466	0.9403	0.9778	0.9761	0.9814
Flor. Aleatórias	0.9704	0.984	0.9927	0.9917	0.9945
SVM	0.9972	0.9998	1	0.9998	1
kNN	0.9882	0.9989	0.9998	1	1

Tabela A.2 Percentual médio de acertos dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo para o conjunto de dados 2

	50	100	150	200	250
LORC	0.9388	0.9553	0.9819	0.991	0.9938
LORCy	0.9732	0.9843	0.992	0.9982	0.9987
Random LORC	0.9431	0.9534	0.972	0.9913	0.9919
Random LORCy	0.9645	0.9865	0.9885	0.9993	0.9979
Reg. Logística	0.8441	0.852	0.8506	0.9012	0.8579
CART	0.8591	0.9644	0.9805	0.994	0.9957
Flor. Aleatórias	0.9543	0.9818	0.988	0.9971	0.9965
SVM	0.9859	0.9968	0.9945	0.9993	0.9996
kNN	0.9706	0.9842	0.992	0.9981	0.9988

Tabela A.3 Percentual médio de acertos dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo para o conjunto de dados 3

	50	100	150	200	250
LORC	0.7407	0.7943	0.8182	0.8682	0.836
LORCy	0.8338	0.9222	0.9195	0.9612	0.9226
Random LORC	0.7541	0.8353	0.8356	0.8917	0.8484
Random LORCy	0.8544	0.9283	0.9325	0.9583	0.9339
Reg. Logística	0.4864	0.5136	0.5427	0.5716	0.5517
CART	0.6218	0.7499	0.7816	0.856	0.8246
Flor. Aleatórias	0.7951	0.8852	0.9099	0.9339	0.914
SVM	0.8595	0.9454	0.9515	0.9634	0.9521
kNN	0.8398	0.93	0.928	0.9624	0.9243

Tabela A.4 Percentual médio de acertos dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo para o conjunto de dados 4

	50	100	150	200	250
LORC	0.7796	0.8125	0.8467	0.8564	0.8424
LORCy	0.8506	0.8728	0.9452	0.9417	0.9315
Random LORC	0.8077	0.8333	0.8789	0.8792	0.8532
Random LORCy	0.8524	0.8838	0.9345	0.9449	0.9316
Reg. Logística	0.5326	0.5683	0.6093	0.5585	0.5863
CART	0.6829	0.7494	0.8382	0.8543	0.8516
Flor. Aleatórias	0.8009	0.867	0.9363	0.9456	0.932
SVM	0.8713	0.9071	0.9533	0.964	0.9501
kNN	0.8455	0.8778	0.9383	0.9471	0.9323

Tabela A.5 Percentual médio de acertos dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo para o conjunto de dados 5

	50	100	150	200	250
LORC	0.719	0.706	0.705	0.727	0.657
LORCy	0.741	0.741	0.812	0.764	0.755
Random LORC	0.779	0.764	0.739	0.801	0.719
Random LORCy	0.799	0.843	0.806	0.804	0.817
Reg. Logística	0.912	0.943	0.941	0.964	0.962
CART	0.975	0.98	0.956	0.966	0.973
Flor. Aleatórias	0.981	0.984	0.971	0.995	0.989
SVM	0.932	0.968	0.963	0.983	0.986
kNN	0.96	0.966	0.964	0.976	0.992

Tabela A.6 Percentual médio de acertos dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo para o conjunto de dados 6

	50	100	150	200	250
LORC	0.9518	0.9638	0.965	0.9593	0.9598
LORCy	1	1	1	1	1
Random LORC	0.9969	0.9841	0.9873	0.9729	0.9785
Random LORCy	0.999	1	1	1	1
Reg. Logística	0.8973	0.8999	0.8996	0.8986	0.9
CART	0.869	0.9238	0.974	0.9744	0.9831
Flor. Aleatórias	0.966	0.9851	0.9973	0.9948	0.9991
SVM	1	1	1	1	1
kNN	1	1	1	1	1

Tabela A.7 Percentual médio de acertos dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo para o conjunto de dados 7

	50	100	150	200	250
LORC	0.5	0.5	0.4901	0.49	0.4902
LORCy	0.5	0.5	0.5	0.5	0.5
Random LORC	0.5101	0.5105	0.5062	0.5043	0.505
Random LORCy	0.5458	0.5918	0.5938	0.591	0.5935
Reg. Logística	1	1	1	1	1
CART	0.53	0.51	0.5	0.5	0.5
Flor. Aleatórias	0.7469	0.9986	0.9986	0.9983	0.9993
SVM	0.76	1	1	0.98	0.96
kNN	0.4798	0.4729	0.4582	0.4552	0.4388

Tabela A.8 Percentual médio de acertos dos métodos de classificação supervisionada para cada tamanho do conjunto de dados de treinamento do modelo, mantendo o tamanho do conjunto de teste fixo para o conjunto de dados 8

Referências Bibliográficas

[A. Malossini, 2006] A. Malossini, E. Blanzieri, R. T. N. (2006). Detecting potential labeling errors in microarrays by data perturbation. *Bioinformatics*, pages 2114–2121.

[Altman, 1992] Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, pages 175–185.

[Assunção et al., 2006] Assunção, R. M., Neves, M. C., Câmara, G., and Freitas, C. C. (2006). Efficient regionalization techniques for socio-economic geographical units using minimum spanning trees. *International Journal of Geographical Information Science*, pages 797–811.

[Banerjee et al., 2014] Banerjee, B., Varma, S., Buddhiraju, K. M., and Eeti, L. N. (2014). Unsupervised multi-spectral satellite image segmentation combining modified mean-shift and a new minimum spanning tree based clustering technique. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, pages 888–894.

[Barandela and Gasca, 2000] Barandela, R. and Gasca, E. (2000). Decontamination of training samples for supervised pattern recognition methods. *Advances in Pattern Recognition, Lecture Notes in Computer Science*, pages 621–630.

[Bell, 1996] Bell, J. F. (1996). Application of classification trees to the habitat preference of upland birds. *Journal of Applied Statistics*, pages 349–360.

[Bi and Jeske, 2010] Bi, Y. and Jeske, D. R. (2010). The efficiency of logistic regression compared to normal discriminant analysis under class-conditional classification noise. *Journal of Multivariate Analysis*, pages 1622–1637.

[Bootkrajang and Kabán, 2013] Bootkrajang, J. and Kabán, A. (2013). Classification of mislabelled microarrays using robust sparse logistic regression. *Bioinformatics*, pages 870–877.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine Learning*, pages 5–32.

[Brodley and Friedl, 1999] Brodley, C. E. and Friedl, M. A. (1999). Identifying mislabeled training data. *Journal of Artificial Intelligence Research*, pages 131–167.

[Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, pages 273–297.

[Dasarathy, 1991] Dasarathy, B. V. (1991). Nearest neighbor(nn) norms: Nn pattern classification techniques. *IEEE Computer Society Press*.

[Efron, 1979] Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, pages 1–26.

[Ferreira et al., 2001] Ferreira, C. A., Soares, J. F., and Cruz, F. R. B. (2001). Reconhecimento de padrões em estatística: Uma abordagem comparativa. *Proceedings of the V Brazilian Conference on Neural Networks*, pages 409–414.

[Frénay and Verseysen, 2014] Frénay, B. and Verseysen, M. (2014). Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, pages 845–869.

[Guan-Wei Wang, 2014] Guan-Wei Wang, Chun-Xia Zhang, J. Z. (2014). Clustering based on sequential representation of minimum spanning tree. *Applied Mathematics and Computation*, pages 521–534.

[Hastie et al., 2009] Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

[Hosmer and Lemeshow, 1989] Hosmer, D. W. and Lemeshow, S. (1989). *Applied Logistic Regression*. Wiley.

[Izenman, 2008] Izenman, A. J. (2008). *Modern Multivariate Statistical Tecquiniques: Regression, Classification and Manifold Learning*. Springer.

[Jiang and Zhou, 2004] Jiang, Y. and Zhou, Z. H. (2004). Editing training data for knn classifiers with neural network ensemble. *Advances in Neural Networks, Lecture Notes in Computer Science*, pages 356–361.

[Kruskal, 1956] Kruskal, J. B. (1956). On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society*, pages 48–50.

[Lawrence and Scholkopf, 2001] Lawrence, N. D. and Scholkopf, B. (2001). Estimating a kernel fisher discriminant in the presence of label noise. *Proceedings of the 18 th International Conference on Machine Learning*.

[Lia et al., 2007] Lia, Y., Wessels, L. F., de Riddera, D., and Reinders, M. J. (2007). Classification in the presence of class noise using a probabilistic kernel fisher method. *Pattern Recognition*, pages 3349–3357.

[Lichman, 2013] Lichman, M. (2013). UCI machine learning repository.

[Magder and Hughes, 1997] Magder, L. S. and Hughes, J. P. (1997). Logistic regression when the outcome is measured with uncertainty. *American Journal of Epidemiology*, pages 195–203.

[Maletic and Marcus, 2000] Maletic, J. I. and Marcus, A. (2000). Data cleansing: Beyond integrity analysis. *Proceedings of the Conference on Information Quality*, pages 200–209.

[Muhlenbach et al., 2004] Muhlenbach, F., Lallich, S., and Zighed, D. A. (2004). Identifying and handling mislabelled instances. *Journal of Intelligent Information Systems*, pages 89–109.

[Nettleton et al., 2010] Nettleton, D. F., Orriols-Puig, A., and Fornells, A. (2010). A study of the effect of different types of noise on the precision of supervised learning techniques. *Artificial Intelligence Review*, pages 275–306.

[Okamoto and Nobuhiro, 1997] Okamoto, S. and Nobuhiro, Y. (1997). An average-case analysis of the k-nearest neighbor classifier for noisy domains. *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, pages 238–243.

[Olman et al., 2009] Olman, V., Mao, F., Wu, H., and Xu, Y. (2009). Parallel clustering algorithm for large data sets with applications in bioinformatics. *IEEE TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS*, pages 344–352.

[Prim, 1957] Prim, R. C. (1957). Shortest connection networks and some generalizations. *Bell System Technical Journal*, pages 1389–1401.

[R Core Team, 2014] R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

[Scholkopf and Smola, 2002] Scholkopf, B. and Smola, A. J. (2002). *Learning with Kernels*. The MIT Press.

[Sánchez et al., 2003] Sánchez, J., Barandela, R., Marqués, A., Alejo, R., and Badenas, J. (2003). Analysis of new techniques to obtain quality training sets. *Pattern Recognition Letters*, pages 1015–1022.

[Theoharatos et al., 2005] Theoharatos, C., Economou, G., and Fotopoulos, S. (2005). Color edge detection using the minimal spanning tree. *Pattern Recognition*, pages 603–606.

[Thomas H. Cormen and Stein, 2009] Thomas H. Cormen, Charles E. Leiserson, R. L. R. and Stein, C. (2009). *Introduction to Algorithms*. The MIT Press.

[Tsaprouni. et al., 2014] Tsaprouni., L., Yang, T., Bell, J., Dick, K., Kanoni, S., Nisbet, J., Viñuela, A., Grundberg, E., Nelson, C., Meduri, E., Buil, A., Cambien, F., Hengstenberg, C., Erdmann, J., Schunkert, H., Goodall, A., Ouwehand, W., Dermitzakis, E., Spector, T., Samani, N., and Deloukas, P. (2014). Cigarette smoking reduces dna methylation levels at multiple genomic loci but the effect is partially reversible upon cessation. *Epigenetics*, pages 1382–1396.

[Wilson and Martinez, 2000] Wilson, D. R. and Martinez, T. R. (2000). Reduction techniques for instance based learning algorithms. *Machine Learning*, pages 257–286.

[Xu et al., 2002] Xu, Y., Olman, V., and Xu, D. (2002). Clustering gene expression data using a graph-theoretic approach: An application of minimum spanning trees. *Bioinformatics*, pages 536–545.

[Yang et al., 2012] Yang, T., Mahdavi, M., Jin, R., Zhang, L., and Zhou, Y. (2012). Multiple kernel learning from noisy labels by stochastic programming. *Proceedings of the 29 th International Conference on Machine Learning*.

[Yasui et al., 2004] Yasui, Y., Pepe, M., amd Bao-Ling Adam, L. H., and Feng, Z. (2004). Partially supervised learning using an em-boosting algorithm. *Biometrics*, pages 199–206.

[Zhang et al., 2009] Zhang, C., Wu, C., Blanzieri, E., Zhou, Y., Wang, Y., Du, W., and Liang, Y. (2009). Methods for labeling error detection in microarrays based on the effect of data perturbation on the regression model. *Bioinformatics*, pages 2708–2714.

[Zhu and Wu, 2004] Zhu, X. and Wu, X. (2004). Class noise vs. attribute noise: A quantitative study. *Artificial Intelligence Review*, pages 177–210.