PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

RANDOM FORESTS ESTOCÁSTICO

SILVIO NORMEY GÓMEZ

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Ciência da Computação na Pontifícia Universidade Católica do Rio Grande do Sul.

Orientador: Paulo Henrique Lemelle Fernandes

Porto Alegre
2012

Dados Internacionais de Catalogação na Publicação (CIP)

G633r Gómez, Silvio Normey
Random forests estocástico / Silvio Normey Gómez. – Porto Alegre, 2012.
64 p.

Diss. (Mestrado) – Fac. de Informática, PUCRS.
Orientador: Prof. Dr. Paulo Henrique Lemelle Fernandes.

1. Informática. 2. Mineração de Dados. 3. Random Forests. I. Fernandes, Paulo Henrique Lemelle. II. Título.

CDD 005.72

Ficha Catalográfica elaborada pelo
Setor de Tratamento da Informação da BC-PUCRS

Pontifícia Universidade Católica do Rio Grande do Sul
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

TERMO DE APRESENTAÇÃO DE DISSERTAÇÃO DE MESTRADO

Dissertação intitulada "Random Forests Estocástico", apresentada Silvio Normey Gómez como parte dos requisitos para obtenção do grau de Mestre em Ciência da Computação, Processamento Paralelo e Distribuído, aprovada em 31/08/2012 pela Comissão Examinadora:

Prof. Dr. Paulo Henrique Lemelle Fernandes – Orientador — PPGCC/PUCRS

Prof. Dr. Duncan Dubugras Alcoba Ruiz – PPGCC/PUCRS

Dra. Lucelene Lopes - FACIN/PNPD

Homologada em 06/12/2012, conforme Ata No. 26 pela Comissão Coordenadora.

Prof. Dr. Paulo Henrique Lemelle Fernandes
Coordenador.

PUCRS
Campus Central
Av. Ipiranga, 6681 – P32– sala 507 – CEP: 90619-900
Fone: (51) 3320-3611 – Fax (51) 3320-3621
E-mail: ppgcc@pucrs.br
www.pucrs.br/facin/pos

Dedico este trabalho à minha filha, Maria Cecilia, a quem espero servir como exemplo.

AGRADECIMENTOS

Primeiramente quero agradecer a Deus pela minha vida e por ter me presenteado com minha filha, Maria Cecilia, uma criança saudável e feliz. Agradeço à minha filha por ter aguardado pacientemente durante estes dois anos. Mesmo sendo um pai presente, quero estar mais perto e por mais tempo. Agradeço à minha esposa Kelly pela paciência, apoio e carinho. Agradeço à minha enteada Regina por cuidar da Maria Cecilia inúmeras vezes. Agradeço aos meus pais, irmão e demais familiares, que mesmo longe torcem por mim. Um sincero agradecimento ao meu orientador, Prof. Dr. Paulo Henrique Lemelle Fernandes, pela imensa paciência, compreensão, apoio e orientação. Aprendi a entender e respeitar o seu trabalho. Quero agradecer ao PPGCC (Programa de Pós-graduação em Ciência da Computação) e à Pontífica Universidade Católica do Rio Grande do Sul, um lugar único em nosso estado. Quero agradecer à todos que foram meus professores de qualquer matéria, assunto, curso ou outro que eu tenha estudado ou praticado na minha vida. Todos foram importantes de certa forma. Quero agradecer à todos os meus colegas de trabalho, colegas de graduação e mestrado. Um grande abraço e agradecimento ao meu colega de mestrado e amigo Alan Ricardo dos Santos pela força e apoio nos momentos de dificuldade, trabalho em conjunto e pelos almoços no Palatu's. Quero agradecer ao meu clube de coração, o Deportivo Social Sarandí Universitário de Rivera, lugar que frequentei durante minha infância e adolescência e onde aprendi valores através do esporte, importantes para a vida toda. Por último deixo aqui três palavras chave: trabalho, dedicação e profissionalismo.

RANDOM FORESTS ESTOCÁSTICO

RESUMO

Na área de Mineração de Dados, experimentos vem sendo realizados utilizando Conjuntos de Classificadores. Estes experimentos são baseados em comparações empíricas que sofrem com a falta de cuidados no que diz respeito à questões de aleatoriedade destes métodos. Experimentamos o Random Forests para avaliar a eficiência do algoritmo quando submetido a estas questões. Estudos sobre os resultados mostram que a sensibilidade do Random Forests é significativamente maior quando comparado com a de outros métodos encontrados na literatura, como Bagging e Boosting. O proposito desta dissertação é diminuir a sensibilidade do Random Forests quando submetido a aleatoriedade. Para alcançar este objetivo, implementamos uma extensão do método, que chamamos de Random Forests Estocástico. Logo especificamos como podem ser alcançadas melhorias no problema encontrado no algoritmo combinando seus resultados. Por último, um estudo é apresentado mostrando as melhorias atingidas no problema de sensibilidade.

Palavras-chave: Mineração de Dados, Conjuntos de Classificadores, Random Forests, Bagging, Boosting.

STOCHASTIC RANDOM FORESTS

ABSTRACT

In the Data Mining area experiments have been carried out using Ensemble Classifiers. We experimented Random Forests to evaluate the performance when randomness is applied. The results of this experiment showed us that the impact of randomness is much more relevant in Random Forests when compared with other algorithms, e.g., Bagging and Boosting. The main purpose of this work is to decrease the effect of randomness in Random Forests. To achieve the main purpose we implemented an extension of this method named Stochastic Random Forests and specified the strategy to increase the performance and stability combining the results. At the end of this work the improvements achieved are presented.

Keywords: Data Mining, Ensemble Classifiers, Random Forests, Bagging, Boosting.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

GB	*Gigabytes*
TB	*TeraBytes*
PB	*Petabytes*
KDD	*Knowledge Discovery in Databases*
RF	*Random Forests*
RFS	*Random Forests Estocástico*

SUMÁRIO

1. INTRODUÇÃO

O grande volume de dados gerados pelos sistemas de computação aumentam consideravelmente dia a dia. Os sistemas que produzem esta quantidade de dados são executados em diferentes contextos, como corporações, empresas de diferentes portes, instituições governamentais e todo tipo de entidade que possa ter um processo ou atividade informatizada. Há pouco tempo atrás, o volume de dados era calculado em gigabyte (GB). Com o rápido crescimento do volume de dados gerados pelos sistemas, este passou a ser calculado em terabyte (TB) e petabyte (PB). Estes dados são de grande valor, e podem auxiliar empresas no planejamento e na tomada de decisões.

Importantes informações estão escondidas, e não podem ser descobertas de maneira rápida e fácil pelos sistemas convencionais. Para atender esta necessidade, surgiu a Mineração de Dados. Esta técnica tem como objetivo descobrir informações não triviais dentro desse conjunto de dados. A Mineração de Dados utiliza métodos de outras áreas, como por exemplo, Estatística Clássica e Inteligência Artificial, para extrair conhecimento sobre um conjunto de dados, segundo Freitas [13].

Com o desenvolvimento da área, foram implementadas diferentes tarefas de Mineração de Dados, como: Análise de Regras de Classificação, Análise de Padrões de Sequência, Análise Clusters, Análise de Outliers e Classificação e Predição. Especificamente dentro da tarefa de Classificação e Predição existe a Combinação de Classificadores, que basicamente combina o resultado de um grupo de classificadores com o objetivo de aumentar a precisão da classificação. Incluídos no conjunto dos algoritmos que implementam esta técnica estão Bagging, Boosting e Random Forests.

Estudos apresentados por Fernandes et al. [12] identificaram que conjuntos de classificadores como Bagging e Boosting, quando submetidos à aleatoriedade, classificam dados de forma instável. O efeito causado pela aleatoriedade produz uma importante variação no resultado da classificação, fazendo com que em algumas situações estes fiquem distantes do desvio padrão.

Para estudar o comportamento do Random Forests quando submetido à aleatoriedade, experimentamos este utilizando como exemplo o experimento do estudo citado anteriormente. A seguir apresentaremos a motivação para a realização do presente trabalho e qual o caminho a ser seguido para atingir o objetivo do mesmo.

1.1 Objetivos do trabalho

Motivados pela hipótese de saber se é possível adaptar o método Random Forests para que este seja menos vulnerável à aleatoriedade, os seguintes objetivos específicos foram traçados buscando responder esta questão:

- Implementar uma extensão do método de mineração de dados Random Forests, o qual chamaremos de Random Forests Estocástico, e especificar como podem ser atingidas melhorias de precisão e estabilidade com a combinação de seus resultados;

- Experimentar o Random Forests Estocástico e analisar seus resultados, comparando estes com os resultados do Random Forests. Buscamos assim identificar se melhorias na precisão e diminuição da variabilidade dos resultados podem ser alcançadas.

1.2 Estrutura do Trabalho

Este documento esta organizado da seguinte forma: o Capítulo 1 apresenta a introdução, hipótese de pesquisa, objetivos do trabalho e sua estrutura. O Capítulo 2 apresenta a revisão da literatura, bem como conceitos fundamentais para o entendimento do presente trabalho, como Mineração de Dados, Classificadores e Combinação de Classificadores . Para um melhor entendimento dos algoritmos estudados, detalhamos o funcionamento de Bagging, Boosting e Random Forests. Trabalhos relacionados a esta dissertação são apresentados no final do capítulo. No Capítulo 3, o experimento que demostra a sensibilidade do Random Forests quando submetido à aleatoriedade é definido e um estudo sobre seus resultados é apresentado. O Capítulo 4 descreve a implementação do Random Forests Estocástico e apresenta, como podem ser obtidos ganhos de precisão e como a variabilidade dos resultados de classificação pode se diminuída através da combinação de resultados. Descreve também o experimento ao qual o Random Forests Estocástico foi submetido. No final do capítulo, os resultados experimentais são analisados e comparados com os resultados obtidos no experimento com Random Forests. Por último, no Capítulo 5, a conclusão do trabalho e a contribuição do mesmo são apresentadas.

2. REVISÃO DA LITERATURA

No presente capítulo serão apresentados conceitos importantes para o entendimento do contexto desta dissertação. Uma noção geral sobre Mineração de Dados, Classificadores e Combinação de Classificadores será transmitida. Em um segundo momento, uma abordagem mais técnica é dirigida dentro da área de Combinação de Classificadores, apresentando um conjunto de três algoritmos: Bagging, Boosting e Random Forests. Os resultados da pesquisa sobre o assunto citado acima são de fundamental importância para o desenvolvimento do trabalho.

2.1 Mineração de Dados

A Mineração de Dados faz parte do processo conhecido como *KDD* (Knowledge Discovery in Databases), que pode ser observado na Figura 2.1. Este processo consiste em um série de etapas de transformação, pre-processamento dos dados até o pós-processamento dos resultados da mineração de dados.

Segundo Tan et al. [20] a Mineração de Dados permite a extração não trivial de conhecimento previamente desconhecido e potencialmente útil em bases de dados. É possível descobrir informações úteis como padrões ou anomalias que por outros meios poderiam ser ignorados. Para atingir tal objetivo esta utiliza técnicas de varias outras áreas (Estatística Clássica, Inteligência Artificial e Aprendizado de Maquina), segundo Freitas [13].

Observamos que tarefas como a busca de informações utilizando sistemas gerenciadores de banco de dados ou a busca de páginas da internet através da utilização de um mecanismos de busca não são consideradas tarefas de mineração de dados e sim de recuperação de dados.

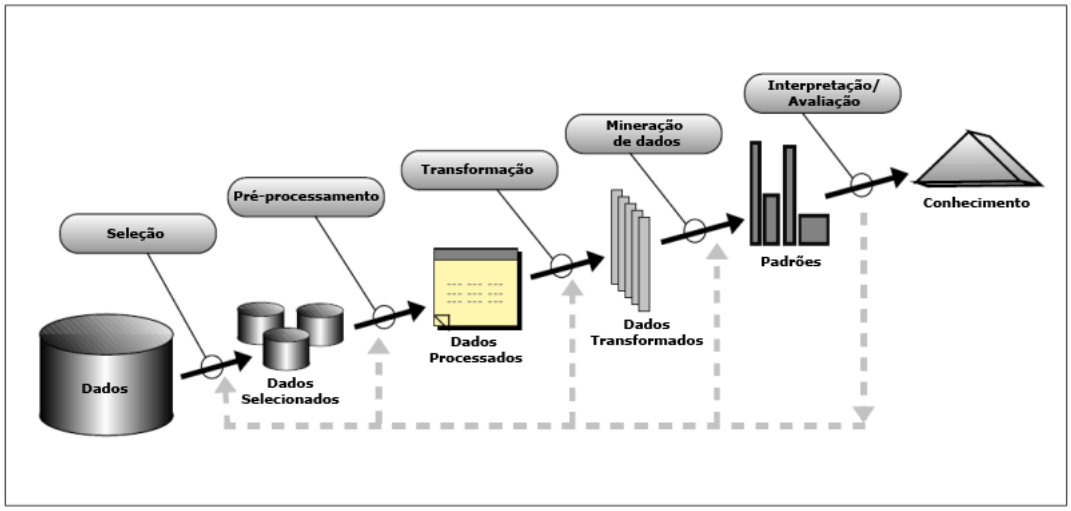

Figura 2.1: Processo de descoberta de conhecimento em bancos de dados.

As técnicas tradicionais de análise de dados encontravam dificuldades para poder superar os desafios que as bases de dados proporcionavam. Dados complexos e com alta dimensionalidade,

distribuição de dados e análises não tradicionais motivaram o surgimento desta área. As tarefas de mineração de dados estão divididas em dois grupos: 1) tarefas que tentam prever o valor de um atributo baseado em *n* atributos 2) tarefas descritivas que tentam derivar padrões que resumam o relacionamento subjacente dos dados. Integrando os grupos citados acima, encontramos tarefas como a Análise de Regras de Classificação, Análise de Padrões de Sequência, Análise de Cluster, Análise de Outlier e a Classificação e Predição. Especificamente dentro da tarefa de Classificação e Predição existem os Classificadores e a Combinação de Classificadores, que são parte do nosso estudo e serão apresentados nas próximas seções.

2.2 Classificadores

A tarefa dos Classificadores é tentar prever a classe de um objeto representado por uma instância, baseado no valor dos seus atributos. Para executar a tarefa de previsão, o classificador utiliza um conjunto de atributos, denominados previsores, e um atributo, denominado classe. Os atributos previsores são utilizados para definir uma classificação efetiva dos registros pertencentes ao conjunto de dados em estudo. O atributo classe, por sua vez, é utilizado com uma hipótese de classificação que será válida ou não pela análise resultante da classificação por meio dos atributos previsores, conforme descrito por Carvalho [9].

Para construção e teste do modelo, o algorimo utiliza dois subconjuntos de dados extraídos da base de dados original. O primeiro conhecido como conjunto de treinamento que é utilizado para construir o classificador. O segundo conhecido como conjunto de testes é utilizado para testar a precisão do modelo.

O conjunto de treinamento é percorrido analisando as relações existentes entre os atributos previsores e o atributo classe. Estas relações são então usadas para prever as classes das instâncias presentes no conjunto de teste, segundo Mitchell [19]. Para testar a precisão do modelo o algoritmo analisa o conjunto de testes e o atributo classe não é considerado. Após prever as classes das instâncias do conjunto de teste, estas são comparadas com as classes de hipótese definidas pelo classificador. O resultado gerado a partir do conjunto de teste é utilizado para obter a taxa de acerto, que é calculada comparando a quantidade de previsões corretas sobre a quantidade de instâncias do conjunto de teste. Como resultado, temos a taxa de classificação do algoritmo. Buscando melhorar esta taxa, surgiu a Combinação de Classificadores, que será apresentada na seção seguinte.

2.3 Combinação de Classificadores

A previsão por múltiplos classificadores é conhecida como técnica de Combinação de Classificadores. Um método de conjunto constrói um conjunto de classificadores básicos a partir de um conjunto de dados de treinamento. Para prever a possível classe de uma determinada instância, todos os classificadores votam por uma classe e a mais votada é selecionada.

Para entender como esta técnica de combinação de classificadores pode melhorar o desempenho, analisaremos o seguinte caso: sobre um conjunto de vinte e cinco classificadores binários, onde cada

um possui uma taxa de erro de $\epsilon = 0.35$, o rótulo da classe de uma instância de testes *D1* é selecionado escolhendo a classe mais votada sobre as previsões feitas pelos classificadores básicos. Se os classificadores forem idênticos, então o grupo classificará erroneamente, permanecendo a taxa de erro de 0.35. Se os classificadores forem independentes e seus erros não estiverem correlacionados, então o grupo fará uma previsão incorreta se mais da metade dos classificadores estiverem incorretos, como pode ser observado no cálculo da Equação 2.1. Cálculo da taxa de erro:

$$\hat{e}grupo = \sum_{i=13}^{25} \binom{25}{i} \epsilon^i (1-\epsilon)^{25-i} = 0.06 \tag{2.1}$$

A taxa é consideravelmente menor que a taxa dos classificadores básicos. Para um melhor entendimento a Tabela 2.1 resume o exemplo apresentado anteriormente. A Figura 2.2 mostra uma visão logica do método de aprendizagem de grupo. Para uma melhor compreensão do funcionamento dos classificadores de conjuntos, apresentaremos Bagging, Boosting e Random Forests métodos largamente utilizados pela comunidade científica.

Tabela 2.1: Exemplo da importância da baixa correlação dos classificadores.

classificadores	*taxa de erro* ϵ	*classificadores correlacionados*	*classificadores não correlacionados*
25	0.35	0.35	0.06

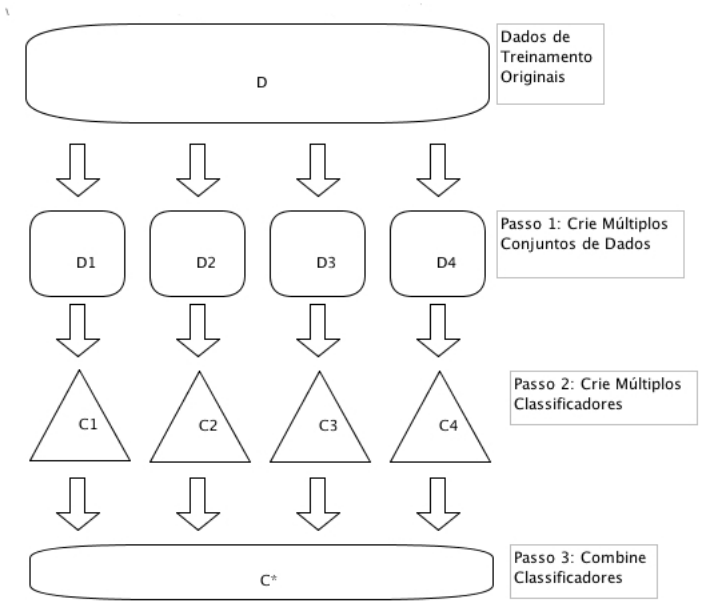

Figura 2.2: Uma visão logica do método de aprendizagem de grupo.

2.3.1 Bagging

Também conhecido como agregação de Bootstrap, foi proposto por Breiman [5], a idéia básica do algoritmo é criar classificadores a partir de uma base de dados de treinamento utilizando distri-

buição uniforme de probabilidades. Cada amostra de Bootstrap possui o mesmo tamanho dos dados originais, já que a amostragem é feita com substituição. Algumas instâncias podem aparecer repetidamente no mesmo conjunto de treinamento, enquanto que outras podem ser omitidas, segundo Quinlan [22]. Os classificadores são gerados independentemente, e a classificação é definida pelo voto majoritário sobre todos os classificadores. Bagging consiste em combinar T classificadores de N amostras geradas a partir do conjunto de treinamento M com R instâncias. Cada classificador contem m instâncias do conjunto de treinamento original M. Ao invés de utilizar todas as instâncias do conjunto original de treinamento, o método de amostragem escolhe instâncias uniformemente com repetição. O método de amostragem gerará K exemplos, que representam aspectos originais da base de dados. Para cada exemplo o classificador é gerado independentemente. A classificação de uma nova instância será executada sobre cada um desses T classificadores. Para cada tentativa $t = 1,2,...,T$, um conjunto de treinamento de tamanho N é amostrado do conjunto de treinamento original. Este conjunto será do mesmo tamanho do conjunto original. A cada tentativa, um classificador Ci será gerado e ao final um classificador $C*$ será formado através da geração de T classificadores obtidos em cada tentativa. Para classificar uma amostra desconhecida, cada classificador Ci retorna o seu voto. Ao final, o classificador $C*$ retorna a classe com maior número de votos. Perturbações no conjunto de treinamento podem causar mudanças significativas no classificador construído. Com isto, Bagging pode melhorar a sua precisão, segundo Breiman [4]. O pseudocódigo de Bagging é exibido na Figura 2.3.

1: Seja *k* o número de amostras de *bootstrap*.
2: **para** *i*=1 até *k* **faça**
3: Crie uma amostra *bootstrap* de tamanho *N*, *Di*.
4: Treine um classificador de base *Ci* sobre a amostra de *bootstrap Di*.
5: **fim do para**
6: $C^*(x) = argmax \sum_i \delta\,(C_i(x) = y)\{\delta(.) = 1$ se seu argumento for verdadeiro e 0 caso contrário$\}$

Figura 2.3: Algoritmo de Bagging.

2.3.2 Boosting

Boosting é um método de combinação de classificadores desenvolvido para oferecer uma classificação com maior eficiência. Este é um procedimento iterativo usado para alterar adaptativamente a distribuição de exemplos de treinamento, de modo que os classificadores de base foquem em exemplos que sejam difíceis de classificar. A partir deste método foram criados vários algoritmos, como por exemplo, *AdaBoost (Adaptative Boosting)*, proposto por Freund e Schapire [14]. Diferentemente de Bagging, Boosting atribui um peso para cada conjunto de treinamento, que pode ser utilizado para desenhar um conjunto de amostras de *Bootstrap* a partir dos dados originais. Podem também ser usados pelo classificador de base para descobrir um modelo que tenha tendência na direção de exemplos de peso, segundo Tan et al. [20].

A distribuição de amostragem do conjunto de treinamento funciona da seguinte forma: inicialmente os exemplos recebem peso *1/N*, assim todos possuem a mesma probabilidade de serem escolhidos para treinamento. Uma amostra é desenhada de acordo com a distribuição de amostras dos exemplos de treinamento, para obter um novo conjunto de treinamento. Logo após, um classificador é induzido a partir do conjunto de treinamento que é utilizado para classificar todas as instâncias do conjunto de dados original. Os pesos dos exemplos de treinamento são atualizados ao final de cada rodada de Boosting. Para forçar que o classificador foque nos exemplos que são de difícil classificação, os exemplos que são classificados incorretamente terão o seu peso aumentado, os que forem classificados corretamente terão seu peso diminuído. À medida que as rodadas de Boosting avançam, os exemplos mais difíceis de ser classificados se tornam ainda mais predominantes. O conjunto final é obtido agregando-se os classificadores de base resultantes de cada rodada de Boosting. A seguir uma explicação técnica do funcionamento de *AdaBoost* será apresentada para um melhor entendimento:

1: **W** = { $w_j = 1/N | j = 1,2,\ldots,N$}. Inicializa os pesos de todos os *N* exemplos.
2: Suponha que *k* seja o número de rodadas de *boosting*.
3: **para** *i*=1 até *k* **faça**
4: Crie o conjunto de treinamento D_i amostrado (com substituição) a partir de *D* de acordo com **W**.
5: Treine um classificador de base C_i sobre D_i.
6: Aplique C_i sobre todos os exemplos no conjunto de treinamento original *D*.
7: $\epsilon_i = \frac{1}{N}\left[\sum_j w_i \delta(C_i(x_j) \neq y_j)\right]$ {Calcula o erro ponderado}.
8: **se** $\epsilon_i > 0{,}5$ **então**
9: **W** = { $w_j = 1/N | j = 1,2,\ldots,N$}. {Reinicializa os pesos de todos os *N* exemplos.}
10: Volte para o Passo 4.
11: **fim se**
12: $\propto = \frac{1}{2} 1n \frac{1-\epsilon_1}{\epsilon_1}$.
13: Atualizar o peso de cada exemplo de acordo com a Equação (**Função 4.2**).
14: $C^*(x) = argmax \sum_{j=1}^{r} \alpha_j \delta(C_j(x) = y))$.

Figura 2.4: Algoritmo de Boosting.

$$w_i^{(j+1)} = \frac{w_i^{(j)}}{z_j} \times \begin{cases} exp^{-\alpha j} \ se \ C^j(x_i) = y_i \\ exp^{\alpha j} \ se \ C^j(x_i) \neq y_i \end{cases} \tag{2.2}$$

Inicialmente, *AdaBoost* inicializa os pesos de todos os *N* exemplos com o valor $1/n$. A primeira geração feita escolherá instâncias de uma maneira uniforme. Logo depois de criado o conjunto de treinamento *Di* a partir de *D*, um classificador *Ci* é treinado sobre *Di*. O conjunto de treinamento *D* é classificado pelo classificador *Ci*. A taxa de erro do classificador *Ci* (linha sete do Algoritmo de Boosting observado na Figura 2.4), é calculada levando em conta o número de instâncias da amostra. O peso de cada uma das instâncias é a efetividade do classificador *Ci*. Caso a taxa de erro seja superior ao palpite aleatório $(\epsilon_i > 0,5)$, descarta-se a amostra e outra amostra é gerada com os pesos em *N* reinicializados. Caso a taxa de erro ϵ_i seja satisfatória, a atualização dos pesos é feita de acordo com a importância do classificador, utilizando para tanto a Equação 2.2.

2.3.3 Random Forests

Random Forests foi proposto formalmente por Breiman [6]. Este é um algoritmo de Combinação de Classificadores projetado especialmente para árvores de decisão. Consiste em uma coleção de classificadores estruturados em árvores $\{h(X,\Theta_k), k=1,....\}$, onde $\{\Theta_k\}$ são vetores independentemente e identicamente distribuídos, e cada árvore vota pela classe mais popular em *X*. Vetores aleatórios são gerados a partir de uma distribuição de probabilidade fixa sobre o vetor de entrada inicial. A precisão do Random Forests é medida probabilisticamente em termos de margem do classificador, dado um conjunto de classificadores $h1(X), h2(X), ..., hk(X)$, e um conjunto de treinamento aleatório a partir do vetor Y, X. A margem do classificador é medida através da Equação 2.3:

$$mg(X,Y) = av_k I(h_k(X) = Y) - max_{j \neq y} av_k I(h_k(X) = j) \tag{2.3}$$

A classe $h_k(X)$ é prevista de *X* de acordo com um classificador construído utilizando o vetor randômico *K*. A precisão do classificador para prever um determinado exemplo de *X* aumenta de acordo com o aumento da margem. A medida que a correlação das árvores aumenta, ou a força do conjunto diminui, o limite de erro de generalização tende a aumentar. O limite de erro de generalização converge para a expressão que pode ser observada na Fórmula 2.4 quando o número de árvores for suficientemente grande.

$$PE^* = P_{x,y}(mg(X,Y) < 0) \tag{2.4}$$

Para um número grande de árvores decorre da *Strong Law of Large Numbers* e da estrutura da árvore que denota na equação a seguir:

$$P_{x,y}\left(P_\Theta(h(X,\Theta) = Y) - max_{j \neq y} P_\Theta(h(X,\Theta) = j) < 0\right. \tag{2.5}$$

Random Forests não faz *overfit* cada vez que mais árvores são adicionadas, mas sim produz um valor limite de erro de generalização. O limite superior de erro de generalização pode ser derivado da integração de dois parâmetros: um é da medida individual da precisão de cada classificador, calculada pela equação da margem do classificador apresentada acima, e da dependência entre eles. A força do conjunto de classificadores $\{h(x,\Theta)\}$ é medida utilizando a equação 2.6 .

$$s = E_{x,y} mr(X,Y) \tag{2.6}$$

Para aumentar à aleatoriedade, Bagging pode ser utilizado em conjunto com a seleção aleatória de características. Cada novo conjunto de treinamento é criado por substituição a partir do novo vetor de entrada inicial. Uma nova árvore é induzida a partir de um novo conjunto de treinamento usando seleção aleatória de características. As árvores atingem o seu tamanho máximo e não é executada a poda sobre as mesmas. Podem ser enumeradas duas razões para a utilização de Bagging: a primeira é que a sua utilização melhora a precisão quando as características aleatórias são utilizadas;

a segunda é que Bagging pode ser usado para fornecer estimativas de erro de generalização da combinação de conjunto de árvores bem como as estimativas para força e correlação durante a execução.

Estas estimativas são feitas utilizando a técnica *Out-of-Bag*, que funciona da seguinte forma. Suponha um método para construção de um classificador a partir de um conjunto de treinamento qualquer, e um determinado conjunto de treinamento *T*. A partir destes, são formados os conjuntos de treinamento de *Bootstrap* T_k. Em seguida são construídos os classificadores $h(x, T_x)$. Este voto é dado para formar o *Bagged Predictor*. Para cada *y*, *x* no conjunto de treinamento são agregados votos somente para aqueles classificadores os quais T_k não contem *y*, *x*. Este classificador é conhecido como *Out-of-Bag*.

Em cada conjunto de treinamento de Bootstrap um terço das instâncias é deixado de fora. Portanto as estimativas de *Out-of-Bag* são baseadas em combinar somente um terço dos classificadores. A medida que a taxa de erro diminui e o número de combinações aumenta, as estimativas de *Out-of-Bag* tendem a superestimar a taxa de erro corrente. A força e correlação entre as árvores também pode ser estimada utilizando o método de *Out-of-Bag*. Estas estimativas internas ajudam a conhecer a precisão de cada classificador e como este pode ser melhorado, segundo Breiman [7].

Cada árvore de decisão usa um vetor aleatório, que é gerado a partir de distribuição de probabilidade fixa. Um vetor aleatório pode ser incorporado ao processo de desenvolvimento da árvore de diversas maneiras.

Uma técnica conhecida e a *Floresta-RI*, onde *RI* refere-se à seleção aleatória de entrada, consiste em selecionar aleatoriamente *F* características de entrada para dividir em cada nodo da árvore de decisão. Ao invés de examinar todas as características disponíveis, o nodo é dividido a partir destas *F* características selecionadas. A árvore é desenvolvida utilizando a técnica de CART que denota no crescimento integral das árvores sem poda. Logo depois da construção das árvores, as previsões são combinadas usando um esquema de votação por maioria. A força e a correlação do Random Forests podem depender do tamanho de *F*. Se *F* for suficientemente pequeno, as árvores tendem a se tornar menos correlacionadas. A força do classificador tende a melhorar com um número maior de características *F*. Para balancear o número de características, é comumente escolhido como $F = log_2 d + 1$, onde *d* é o número de características de entrada. Se o número de *d* características originais for muito pequeno, é difícil escolher um conjunto independente de características aleatórias para construir uma árvore de decisão.

Para aumentar o espaço das características, são criadas combinações lineares das características de entrada. Esta abordagem é conhecida como *Floresta-RC*, conforme apresentado por Breiman [6]. Em cada nodo uma nova característica é gerada selecionando aleatoriamente *L* características de entrada. Estas são combinadas linearmente, usando coeficientes gerados a partir de uma distribuição uniforme na faixa de [-1,1]. Em cada nodo são geradas *F* novas características combinadas aleatoriamente, e a melhor é selecionada para dividir o nodo. Suponha que existam *M* variáveis de entrada. Logo depois que cada árvore é construída, o valor da variável *mth* no exemplo de *Out-of-Bag* é randomicamente permutado, e o conjunto de dados classificado pela árvore correspondente. A clas-

sificação dada por cada X_n que está fora do *Out-of-Bag* é salva. Este procedimento é repetido por m=1,2,...M. No final da execução as classes diferentes do *Out-of-Bag* votam por X_n, a variável *mth* é comparada com o rótulo da classe verdadeira de X_n para calcular a taxa de erro de classificação. A saída é o aumento percentual da taxa de classificação incorreta em relação à taxa de *Out-of-Bag*. Para um melhor entendimento do Random Forests, a Figura 2.5 mostra o processo de construção do modelo, enquanto que a Figura 2.6 mostra o processo de classificação das instâncias. A Figura 2.7 mostra o funcionamento do Random Forests. O Capítulo 3 apresenta o experimento realizado com o Random Forests, além das características e variações do mesmo.

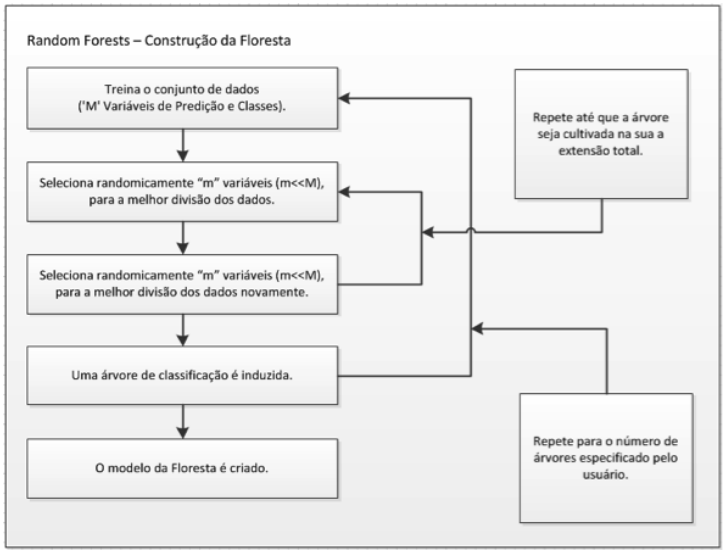

Figura 2.5: Processo de construção do modelo.

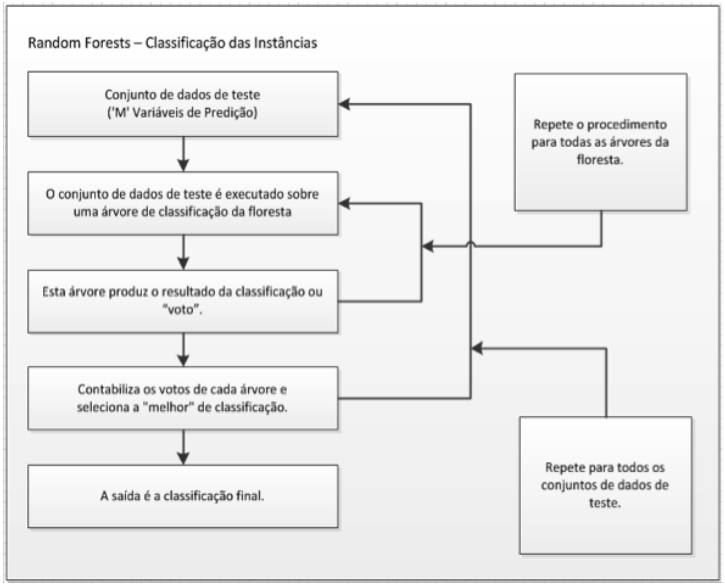

Figura 2.6: Processo de classificação das instâncias.

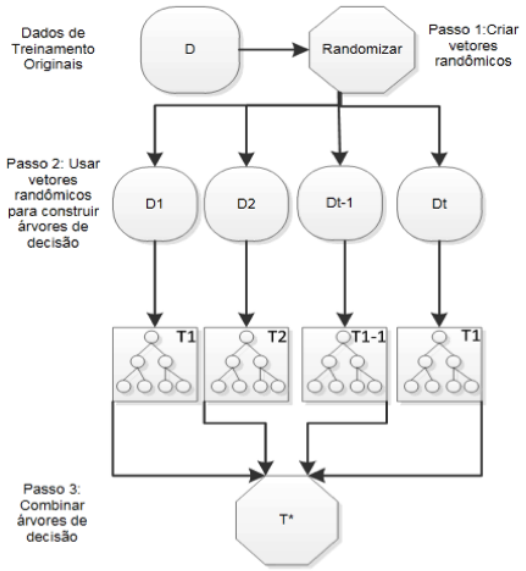

Figura 2.7: Random Forests.

2.4 Trabalhos Relacionados

Esta seção apresenta trabalhos relacionados com o aqui proposto. Estes objetivam melhorar a performance do Random Forests em algum aspecto, da mesma forma que o presente trabalho.

O estudo realizado por Tsymbal et al. [26] demostrou que a precisão do Random Forests pode ser melhorada substituindo a função de combinação do resultado dos classificadores pela integração dinâmica, que é baseada no desempenho local das estimativas de performance dos preditores básicos. O experimento conduzido no trabalho demostrou que a integração dinâmica aumentou a precisão em algumas bases de dados.

Conforme Robnik-Šikonja [23], individualmente, cada um dos classificadores base não são preditores efetivos ou seja, a seleção randômica dos atributos faz com que, individualmente, cada um dos classificadores seja mais fraco. O primeiro objetivo do trabalho foi fortalecer individualmente cada um dos classificadores, sem sacrificar a variedade entre eles. O segundo objetivo tratou de aumentar a variância sem sacrificar a força de cada um. Foram alcançadas melhorias no primeiro objetivo usando o algoritmo *ReliefF* para a estimativa de atributos e no segundo objetivo usando diferentes medidas de avaliação de atributos para seleção de divisão. Os classificadores, em certas situações, não obtiveram o mesmo sucesso ao classificar instâncias, tendo sido utilizadas estimativas internas para identificar instâncias mais similares ao do rotulo desejado. Em seguida, os votos das árvores foram ponderados com a força que eles demostraram para aquela instância. Com isso, foram alcançadas melhorias em várias bases de dados.

Boinee et al. [3] notaram que algoritmos de aprendizado como AdaBoost e Bagging melhoraram o resultado da classificação. No trabalho, o autor experimentou utilizar estas técnicas de meta-aprendizado combinando com o Random Forests. Um experimento foi conduzido e observou-se em média vantagem em favor do que ele chamou de Bagged Random Forests, em relação ao Random

Forests.

Thongkam et al. [25] propuseram uma combinação de AdaBoost e Random Forests para construir um modelo de predição para o câncer de mama. O Random Forests foi usado como um *weak learner* de AdaBoost para selecionar as instâncias com um peso maior durante o processo de Boosting, procurando melhorar a precisão, estabilidade e problemas de *overfiting*. A capacidade do método foi avaliada medindo a taxa de acerto, sensibilidade e especificidade. O resultado do experimento determinou que o método proposto superou C4.5 [21], AdaBoost e o próprio Random Forests.

Os trabalhos investigados aqui tem em comum a preocupação em melhorar a precisão do Random Forests. Notamos apenas em [25] a preocupação em observar o comportamento da sensibilidade. Percebemos também que, em nenhum dos trabalhos, questões relacionadas à aleatoriedade foram levadas em conta, diferentemente do trabalho apresentado aqui, cujo objetivo é melhorar a precisão e a variabilidade dos resultados do Random Forests quando submetido à aleatoriedade.

3. RANDOM FORESTS

O presente capítulo apresenta um estudo sobre o comportamento do Random Forests quando submetido à aleatoriedade. A Seção 3.1 descreve o experimento executado, os artefatos utilizados, configurações aplicadas e técnicas empregadas durante o experimento. A Seção 3.2 desenvolve um estudo comparativo utilizando como base os resultados experimentais. Em um primeiro momento, os resultados do Random Forests são analisados, mostrando as tendências encontradas. No final, uma segunda análise, comparando os resultados do Random Forests com relação a resultados de outros algoritmos encontrados na literatura, é apresentada.

3.1 Definição do experimento utilizando aleatoriedade

Uma série de estudos foram realizados anteriormente comparando Bagging e Boosting [22] [19] [24] [11] [28] [17] [16] [18], o que nos auxilia na construção do experimento. Para executar o experimento, foi utilizada a ferramenta WEKA (*Waikato Enviroment for Knowledge Analysis* 3.4.6) [27], e o resultado foi armazenado em uma base de dados. A classificação foi executada utilizando a técnica de *ten-folds stratified cross-validation*, que utiliza 90% do conjunto de dados para treinamento e 10% para testes. Mais detalhes sobre esta técnica podem se encontrados em [10] [15]. Para execução dos testes foram utilizadas trinta bases de dados com diferentes características exibidas na Tabela 3.1, esta foi retirada de [12], maiores informações podem ser consultadas na fonte. As bases de dados pertencem à *Universidade da California Irvine* [1] e da *Universidade de West Virginia* [2]. Os resultados comparados de Bagging e Boosting foram extraídos do estudo realizado por Fernandes et al. [12]. No experimento em [12] Bagging e Boosting foram configurados na ferramenta *Weka* com a opção -*Q*, que utiliza *bootstrap* para construir os classificadores. Random Forests constrói os classificadores utilizando esta técnica, conforme encontrado em [8]. Para configurar a semente a ser utilizada, o parametro -*S* foi configurado de forma aleatória conforme apresentaremos na especificação da configuração.

Procurando alcançar uma maior precisão no resultado do experimento ao igual que em [12], foram removidos os 5 maiores e os 5 menores resultados de cada base de dados considerando o número de classificadores utilizados (10, 30 e 50). Este procedimento evita que algum resultado com um desvio considerável exerça um peso grande no cálculo do percentual de média de acerto. O experimento foi executado em duas rodadas, nas quais foram aplicadas as seguintes configurações para cada base de dados:

- Configuração 1;
  - Sementes: 1 até 100 (Os resultados de todas as execuções foram agrupados em um único resultado).
  - Classificadores 10, 30 e 50.

- Configuração 2;
  - Sementes: *Default* = 1.
  - Classificadores 10, 30, 50.

Os dados resultantes do presente experimento estão disponíveis em formato digital na biblioteca da Pontifícia Universidade Católica do Rio Grande do Sul.

Tabela 3.1: Características das Bases de dados.

Base de Dados		Informações sobre os dados			Informações sobre as classes	
	id	*atributos*	*instâncias*	*faltando*	*classes*	*desbalanceamento*
Abalone	B01	8	4,177	0.00%	28	0.071
Audiology	B03	69	226	2.03%	24	0.103
Balance	B04	4	625	0.00%	3	0.146
Breast cancer	B05	9	286	0.39%	2	0.165
Car Evaluation	B06	6	1,728	0.00%	4	0.390
CM1 software defect	B07	21	498	0.00%	2	0.645
Datatrieve	B08	8	130	0.00%	2	0.690
Desharnais	B09	11	81	0.00%	3	0.150
Ecoli	B10	8	336	0.00%	8	0.168
Echo cardiogram	B11	11	132	5.10%	3	0.054
Glass	B12	10	214	0.00%	6	0.116
Heart(Cleveland)	B13	13	303	0.38%	2	0.008
Heart statlog	B14	13	270	0.00%	2	0.012
Hepatitis	B15	19	155	5.70%	2	0.345
JM1 software defect	B16	21	10,885	0.00%	2	0.376
Kr-vs-kp	B17	36	3,196	0.00%	2	0.002
MW1 software defect	B18	37	403	0.00%	2	0.716
Pima-diabetes	B19	8	768	0.00%	2	0.091
Post-operative	B20	8	90	0.42%	3	0.366
Primary-tumor	B21	17	339	3.92%	21	0.066
Reuse	B22	27	24	0.93%	2	0.063
Solar Flare	B23	12	1,389	0.00%	8	0.682
Tic-Tac-Toe Endgame	B24	9	958	0.00%	2	0.094
Thyroid(Allhyper)	B25	29	2,800	5.61%	4	0.928
Thyroid(Hypothyroid)	B26	29	3,772	5.54%	4	0.807
Thyroid(Sick euthyroid)	B27	25	3,163	6.74%	2	0.664
Wbdc	B28	30	569	0.00%	2	0.065
Wisconsin breast cancer	B29	9	699	0.25%	2	0.096
Yeast	B31	9	1,484	0.00%	10	0.137
Zoo	B32	17	101	0.00%	7	0.114

3.2 Análise numérica dos métodos

O resultado do experimento é apresentado na Tabela 3.2. Esta lista o resultado experimental para cada uma das configurações descritas anteriormente. Estes resultados servirão como fonte para nossas observações e estudos.

A primeira observação sobre os resultados do experimento é uma comparação no comportamento do algoritmo quando submetido à seleção randômica da semente. Observamos em alguns casos a existência de diferença na média de acerto entre a classificação utilizando a semente *Default* e a classificação utilizando 100 sementes randômicas. Em alguns casos, a diferença encontrada ultrapassou os 5%. Ao aumentar o número de classificadores, identificamos em algumas situações que a diferença diminuiu. Para a base de dados B05, por exemplo, encontramos um comportamento igual para as configurações de 10, 30 e 50 classificadores. Utilizando 100 sementes randômicas, o percentual de acerto variou entre 48.1992% e 50.9578%, e para a semente *Default* entre 55.1724% e 51.7241%. Para a base de dados B09, o percentual de acerto utilizando a semente *Default* com 10 classificadores foi de 55.5556% e para 100 sementes randômicas de 67.6544%. Para 30 classificadores este valor se manteve em 77.7778% para a semente *Default* e 67.9013% para 100

Tabela 3.2: Resultado da classificação para Random Forests com 10, 30 e 50 classificadores.

Base de Dados	100 Sementes									Semente Default		
	10 classificadores			30 classificadores			50 classificadores			10	30	50
id	*média de acerto*	*desvio padrão*	*distância*	*média de acerto*	*desvio padrão*	*distância*	*média de acerto*	*desvio padrão*	*distância*	classificadores		
B01	22.5678	1.2002	4.7847	23.5088	1.2150	4.5455	23.6124	0.9382	3.8278	21.7703	24.8804	24.6411
B03	76.5700	3.2389	13.0435	78.8889	2.2161	8.6957	79.0822	1.7114	4.3478	73.9130	82.6087	82.6087
B04	6.1552	0.9466	3.1746	6.2963	0.3724	3.1746	6.3845	0.2353	1.5873	4.7619	4.7619	6.3492
B05	50.9578	4.6745	17.2414	48.2376	3.6732	13.7931	48.1992	3.6176	13.7931	55.1724	55.1724	51.7241
B06	93.4746	1.0692	4.6243	94.2710	0.8822	3.4682	94.5536	0.6985	2.8902	93.6416	94.2197	93.0636
B07	86.1556	0.9107	4.0000	86.0444	0.2965	2.0000	86.0000	0.0000	0.0000	86.0000	86.0000	86.0000
B08	84.2735	3.4422	15.3846	84.6154	0.0000	0.0000	84.6154	0.0000	0.0000	84.6154	84.6154	84.6154
B09	67.6544	7.9261	22.2222	67.9013	7.1534	22.2222	68.5186	5.3326	22.2222	55.5556	77.7778	66.6667
B10	67.8431	4.7445	17.6471	66.0784	3.3346	11.7647	65.719	3.1536	11.7647	73.5294	64.7059	64.7059
B11	57.4603	5.6569	21.4286	58.8095	5.6710	21.4286	58.8889	6.3628	21.4286	57.1429	64.2857	57.1429
B12	99.7475	1.0470	4.5455	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000	100.0000	100.0000	100.0000
B13	87.7061	3.5828	12.9032	88.8172	2.2265	9.6774	88.4946	1.6786	6.4516	90.3226	87.0968	87.0968
B14	81.6461	3.5512	14.8148	83.0041	3.1355	11.1111	83.7037	2.5923	11.1111	81.4815	81.4815	81.4815
B15	92.1528	3.9262	12.5000	92.6389	3.4523	12.5000	92.6389	2.7440	12.5000	93.7500	87.5000	87.5000
B16	81.2050	0.4500	1.7447	81.7151	0.3329	1.2856	81.8406	0.3079	1.1938	81.3591	81.7264	81.8182
B17	99.0451	0.3049	1.2500	99.3542	0.2237	0.6250	99.3299	0.2018	0.6250	99.0625	99.0625	99.3750
B18	91.0027	1.7010	7.3171	91.5718	1.3264	4.8780	91.9783	1.1117	2.4390	85.3659	92.6829	90.2439
B19	76.3204	2.4783	9.0909	77.0852	1.5612	6.4935	77.3016	1.3934	5.1948	80.5195	79.2208	76.6234
B20	55.4321	7.1631	22.2223	55.8025	7.0623	22.2223	57.0371	6.9078	22.2223	55.5556	66.6667	66.6667
B21	44.1176	3.4152	14.7059	46.5359	2.9766	11.7647	47.6470	2.3414	11.7647	41.1765	41.1765	41.1765
B22	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000	100.0000	100.0000	100.0000
B23	81.9664	0.8199	2.8777	81.9025	0.6552	2.1583	81.9185	0.5727	2.1583	81.2950	82.0144	81.2950
B24	91.3889	2.2672	9.3750	94.1898	1.7033	6.2500	95.3704	1.4987	5.2084	95.8333	94.7917	95.8333
B25	98.7096	0.3717	1.5873	99.0094	0.2463	0.7937	99.0682	0.1992	0.7937	99.2064	99.2064	99.2064
B26	98.2500	0.2456	1.0715	98.3016	0.1543	0.3571	98.2619	0.1333	0.7143	98.2143	98.2143	98.2143
B27	96.8384	0.3442	1.2618	96.9471	0.2403	0.9463	96.9891	0.2419	0.9463	97.1609	97.1609	96.8454
B28	94.7173	1.8124	7.0175	95.0487	1.4059	7.0175	95.3996	1.3038	5.2631	96.4912	96.4912	96.4912
B29	99.9206	0.3291	1.4286	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000	100.0000	100.0000	100.0000
B31	46.7562	6.6485	24.8322	50.1641	4.7940	16.1074	51.1186	4.2018	15.4363	36.9128	33.5571	32.8859
B32	84.0404	6.5797	27.2727	86.8687	5.2976	18.1818	86.5657	4.5664	9.0909	81.8182	100.0000	81.8182
média	77.1359	2.6949	10.0457	77.78694	2.0536	7.4487	78.0079	1.8015	6.4991	76.7209	78.5693	77.0696

sementes randômicas. Para 50 classificadores o resultado foi de 66.6667% para a semente *Default* e 68.5186% para 100 sementes randômicas. Para a base de dados B10 a assertividade foi de 73.5294% para a semente *Default* e 67.8431% para 100 sementes e 10 classificadores, 64.7059% para a semente *Default* e 66.0784% para 100 sementes e 30 classificadores e 64.7059% para a semente *Default* e 65.719% para 100 sementes e 50 classificadores. Os resultados apresentados na Tabela 3.2 mostram a variabilidade causada pela aleatoriedade no Random Forests. Isto é constatando comparando a distância entre os resultados encontrados nas duas configurações do experimento. A continuação observaremos os resultados do Random Forests em comparação com Bagging e Boosting. Observaremos se a tendência da instabilidade pode ser identificada nas próximas comparações.

Para alguns pesquisadores, como Quinlan [22] um método pode ser considerado mais preciso do que o outro se a diferença na média de acerto for igual ou superior a 2%. Utilizaremos este percentual como base para comparar os algoritmos e ressaltar caso algum resultado encontrado alcance este valor. Para observar o efeito da utilização de 100 sementes randômicas, utilizaremos os valores da coluna de média de acerto que se encontram na Tabela 3.2. A seguir, compararemos Random Forests, Bagging e Boosting com base nos valores da coluna citada anteriormente. Considerando a margem de 2% ou mais de diferença, observamos a seguir em quantas situações o Random Forests foi mais preciso que Bagging e Boosting.

Os resultados para Bagging e Boosting foram retirados de [12]. Para fins de comparação, levaremos em conta a quantidade de classificadores utilizados (10, 30 e 50) e a utilização da semente *Default* ou a seleção aleatória das sementes. Observando os resultados da média de acerto, identifi-

camos que Random Forests foi melhor em 20/90, Bagging em 14/90 e Boosting em 10/90 dos casos. O resultado foi formado pela soma dos resultados da classificação com 10, 30 e 50 classificadores utilizando a semente *Default*. Este resultado mostra uma ampla vantagem para o Random Forests quando utilizado com a semente *Default*. Apesar do resultado indicar vantagem para o Random Forests, os resultados de média de acerto variam quando aplicada a seleção randômica da semente. Utilizando a mesma quantidade de classificadores da observação anterior, e 100 sementes randômicas, o resultado encontrado foi o seguinte: Boosting foi melhor em 12/90, Bagging em 10/90 e Random Forests 16/90 dos casos. Mesmo permanecendo dentro da margem de 2% em 52/90 dos casos, este resultado mostra uma queda na vantagem do Random Forests quando utilizadas as 100 sementes randômicas, quando comparado ao uso da semente *Default*.

Buscando confirmar a tendência da instabilidade ao utilizar a seleção randômica da semente, observaremos o comportamento do Random Forests com relação as bases de dados B05, B31, B13 e B17. Consideraremos as características das bases de dados com o número de instâncias e desbalanceamento. A base de dados B05 possui 286 instâncias, e o resultado da média de acerto de classificação para a semente *Default* foi de 54.0229%, enquanto que para 100 sementes randômicas foi 49.1315%, o que totaliza uma diferença de 4.8914%. A base de dados B31 possui 1484 instâncias e a média resultante foi de 34.4519% para a semente *Default* e 49.3463% para a seleção randômica da semente, o que proporciona uma diferença de 14.8944%. Esta observação mostra que independente da quantidade de instâncias, ao utilizar 100 sementes randômicas, Random Forests apresenta variabilidade nos seus resultados.

A observação sobre as bases de dados B13 e B17 verifica o comportamento do Random Forests quando a base de dados possui um bom balanceamento em relação à seleção de 100 sementes randômicas. A base de dados B13 possui duas classes, 303 instâncias e um ótimo balanceamento de 0.0080. O resultado para esta base de dados foi de 88.1720% utilizando a semente *Default* e de 88.3393% para a utilização de 100 sementes. A base de dados B17 possui duas classes, 3196 instâncias e um balanceamento de 0.0020. O resultado para esta base de dados foi de 99.1667% para a semente *Default* e de 99.2430% utilizando 100 sementes randômicas. Esta última observação mostra que para estes dois casos, o balanceamento da base de dados e a utilização de 100 sementes randômicas não causou instabilidade no algoritmo.

O desvio padrão encontrado em [12] foi de 0.5% para Bagging e 0.75% para Boosting. A comparação destes resultados mostra uma vantagem para Bagging de 0.25%. Calculamos o desvio padrão para Random Forests utilizando o resultado do nosso experimento e o valor encontrado foi de 2.1833%. Comparamos o resultado do Random Forests com relação aos de Bagging e Boosting, e os resultados encontrados mostram uma vantagem de 1.6833% para Bagging e 1.4333% para Boosting.

Outro ponto observado foi a média da distância, que representa a distância entre o maior e o menor valor de classificação para uma mesma base de dados. O valor encontrado para Random Forests foi de 7.9978%, o que mostra que dependendo da semente, são gerados valores muito distantes do desvio padrão. Para Bagging, encontramos 1.9259%, e 2.9167% para Boosting. Estes

resultados proporcionam uma ampla vantagem de 6.0719% em favor de Bagging e de 5.0808% em favor de Boosting. A Tabela 3.3 exibe 9 casos nos quais a distância entre o valor mínimo e o valor máximo é maior a 10.00%. Estes exemplos nos mostram que o efeito da aleatoriedade causa variabilidade nos resultados do Random Forests.

A outra comparação realizada refere-se aos valores de distância citados anteriormente. Foram analisados valores encontrados dentre todas as bases de dados, não necessariamente os mais altos. Utilizando dez classificadores, o maior valor encontrado de distância entre as bases de dados selecionadas foi de 22.2223% na base de dados B20. Para Bagging o valor da distância foi de 4.4444%, o que representa uma diferença de 17.7779% em seu favor. Para Boosting o valor encontrado foi de 10.0000%, o que representa uma vantagem de 12.2223% em favor deste. Utilizando trinta classificadores destacamos o valor encontrado de 22.2223% para a base de dados B20. O percentual encontrado em Bagging para esta mesma base de dados foi de 3.3333%, o que representa uma vantagem de 18.8889% em favor deste, enquanto que o valor encontrado para Boosting foi de 8.8889%, o que resulta em uma vantagem de 13.3334% em seu favor. O Random Forests utilizando 50 classificadores obteve um resultado de 22.2222% na base de dados B09. Para esta mesma base de dados encontramos para Bagging o valor de 4.9383%, o que proporciona uma diferença de 17.2839% em favor de Bagging, e para Boosting o valor encontrado foi de 7.4074%, resultando em uma diferença de 14.8148%. Podemos concluir que em todos os casos analisados o Random Forests encontrou-se em desvantagem com relação aos demais algoritmos

Tabela 3.3: Casos excepcionais com distância maior que 10%.

classificadores	*base de dados*	*características*	*instâncias*	*média de acerto*	*desvio padrão*	*mínimo*	*máximo*	*distância*
10	B20	8	90	55.4321	7.1631	44.4444	66.6667	22.2223
	B11	11	132	57.4603	5.6569	42.8571	64.2857	21.4286
	B10	8	336	67.8431	4.7445	58.8235	76.4706	17.6471
30	B20	8	90	55.8025	7.0623	44.4444	66.6667	22.2223
	B09	11	81	67.9013	7.1534	55.5556	77.7778	22.2222
	B11	11	132	58.8095	5.6710	50.0000	71.4286	21.4286
50	B09	11	81	68.5186	5.3326	55.5556	77.7778	22.2222
	B11	11	13	58.8889	6.3628	50.0000	71.4286	21.4286
	B14	13	27	83.7037	2.5923	77.7778	88.8889	11.1111

A primeira observação realizada sore o Random Forests foi uma comparação entre o comportamento do algoritmo utilizando a semente *Default* e a seleção randômica da semente. Encontramos uma diferença que chegou a ultrapassar os 10% utilizando as duas configurações possíveis do experimento. Observamos que ao aumentar o número de classificadores, a diferença diminuiu. Considerando a margem 2% citada anetriormente, encontramos um comportamento instável do Random Forests em comparação a Bagging e Boosting, conforme descrevemos anteriormente. Utilizando a semente *Default*, observamos que Random Forests foi superior em 20/90 dos casos, Bagging 14/90 e Boosting 10/90, o que nos indica uma superioridade do Random Forests quando utilizada apenas uma semente. Quando observado o comportamento do Random Forests utilizando a seleção randô-

mica da semente, encontramos o seguinte resultado: Boosting foi melhor em 12/90 casos, Bagging em 10/90 e Random Forests em 16/90.

O resultado da observação anterior nos mostra que a seleção randômica da semente causa instabilidade no algoritmo. Este resultado pode ser confirmado quando comparamos o desvio padrão do resultado da classificação e a distância em comparação com os dos demais algoritmos. Os resultados das observações apresentadas acima nos mostram que o Random Forests possui um bom rendimento utilizando a semente *Default*. Quando analisamos o algoritmo com relação a seleção randômica da semente, notamos instabilidade no seu comportamento. Esta instabilidade causou uma grande variação nos percentuais de média de acerto, refletindo por consequência no valor de desvio padrão e na distância. Visando explorar a sensibilidade detectada no Random Forests, este trabalho propõe a implementação e análise do método Random Forests Estocástico, a fim de tentar superar o problema da aleatoriedade.

4. RANDOM FORESTS ESTOCÁSTICO

O presente capítulo apresenta o desenvolvimento e experimentação do Random Forests Estocástico. Na Seção 4.1, o seu funcionamento é descrito, assim como a estratégia para superar o problema causado pela aleatoriedade. A seção 4.2 descreve o experimento realizado para analisar a precisão do Random Forests Estocástico. A Seção 4.3 apresenta um estudo sobre os resultados experimentais. Procurando alcançar um espectro maior na análise dos resultados, estes são observados por aspectos diferentes. Para tanto, serão comparados os resultados do Random Forests Estocástico, do Random Forests, de Bagging e de Boosting. Por fim, comparações quantitativas serão feitas considerando características das bases de dados.

4.1 Método Random Forests Estocástico

Apoiados pela literatura investigada na Seção 2.4, e pelos resultados encontrados nos experimentos da Seção 3.2, apresentaremos a nossa proposta. Esta tem como objetivo melhorar a precisão e diminuir a variabilidade dos resultados do método Random Forests quando este é submetido à aleatoriedade.

Desenvolvimento do método

Propomos implementar uma extensão do Random Forests, o qual chamamos de Random Forests Estocástico. Destacamos que a nossa implementação não realizou nenhuma alteração no funcionamento básico do Random Forests. Não foi feita nenhuma alteração no processo de construção do modelo, o qual será igual ao apresentado na Figura 2.5. Funcionalidades como seleção de características, geração aleatória dos vetores, cálculo do valor limite do erro de generalização, não execução de poda não foram modificadas. O Random Forests Estocástico adiciona um novo método o qual implementa a classificação estocástica, servindo esta como resultado da classificação. A Figura 4.1 exibe o funcionamento sistêmico do Random Forests Estocástico. O método de classificação estocástica, ao invés de retornar um único voto, como o fazem os conjuntos de classificadores, retorna uma matriz de probabilidades. As informações providas como resultado da classificação são a classe votada, percentual de votos e a quantidade de votos. As informações citadas anteriormente são retornadas para cada elemento do conjunto de resultados. O percentual de votos é calculado utilizando a quantidade de votos que esta classe obteve sobre o total de classificadores da floresta. A Fórmula 4.1 é utilizada para calcular o percentual de votos e conta com os seguintes elementos: P é o percentual resultante de uma classe x, e é calculado multiplicando-se v que é a quantidade de votos da classe, por cem. O resultado desta operação é dividido por T, que é o numero de árvores da floresta. O pseudocódigo do método, é apresentado no Algoritmo 4.1.

$$P_x = \frac{(\nu * 100)}{T} \tag{4.1}$$

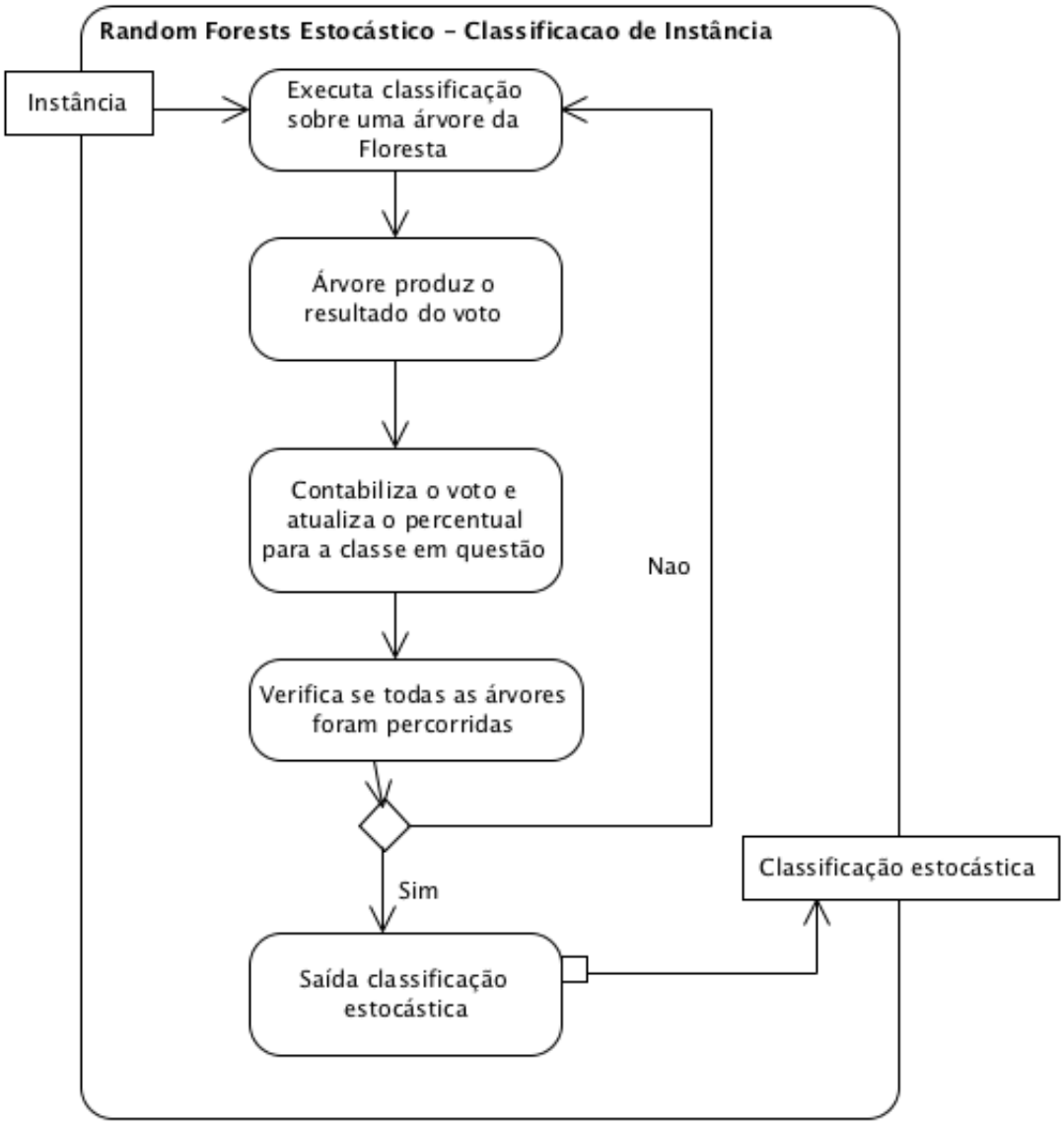

Figura 4.1: Processo de Classificação Estocástica.

Para um melhor entendimento do funcionamento do Random Forests Estocástico descreveremos o mesmo a seguir. Na linha 2 é classificada a instância *I*, armazenando o resultado em *D*. O código executado desde a linha 4 até a linha 11 verifica se a classe votada encontra-se dentro da matriz de resultados, e computa mais um voto caso verdadeiro. O percentual de votos da classe em questão é atualizado. Na linha 12 verifica-se se a classe votada foi localizada na matriz de resultados. Caso não tenha sido encontrada, será adicionada uma nova posição na matriz de resultados. Um voto será computado e o percentual de votos é atualizado para a classe em questão. Como podemos observar nas linhas 8 e 16, a Fórmula 4.1 é utilizada para calcular o percentual de votos para a classe em questão. Finalmente, na linha 19, o resultado de classificação estocástica é retornado. O código fonte do Random Forests Estocástico está disponível em formato digital na biblioteca da Pontifícia Universidade Católica do Rio Grande do Sul.

Algoritmo 4.1: Algoritmo do método de classificação do Random Forests Estocástico.

```
{I : Instância a ser classificada.}
{MC : Matriz de classificadores.}
{MR : Matriz de resultados.}
{D : Resultado da classificação.}
1: for all (i ∈ MC) do
2:    D = MC(i).ClassifyInstance(I) {Árvore classifica instância}
3:    F = false {Variável de controle de busca}
4:    for all (j ∈ MR) do
5:       if MR(j).Class == D then
6:          F = true
7:          MR(j).Votes++
8:          MR(j).Percent = ((MR(j).Votes*100)/MC.Lenght)
9:          return {Sai do loop}
10:      end if
11:   end for
12:   if F == false then
13:      MR.Insert
14:      MR(MR.Lenght-1).Class = D
15:      MR(MR.Lenght-1).Votes = 1
16:      MR(MR.Lenght-1).Percent = (MR(MR.Lenght-1).Votes*100)/MC.Lenght)
17:   end if
18: end for
19: return MR {Retorna a matriz de resultados}
```

Diminuindo a variabilidade dos resultados

A chave para diminuir a variabilidade dos resultados causada pela aleatoriedade consistem em combinar diferentes resultados de classificação do Random Forests Estocástico. Estas séries de classificações devem ser executadas utilizando diferentes configurações de sementes, em todos os casos. Estes resultados são combinados, e a classe mais votada dentre todos os resultados é selecionada como resultado da classificação. Combinar os resultados é agrupar o resultado de cada classificação estocástica em um único resultado. O campo agrupador é o *Class* e o campo *Percent* é utilizado para compor a soma. A seguir apresentaremos um exemplo de como o problema de sensibilidade é solucionado.

Suponha uma instância *I* de um determinado conjunto de teste *T*, onde o rótulo da classe esperado para esta instância seja *A*. Suponha um determinado conjunto de treinamento *R* e suponha também que um grupo de quatro conjuntos de classificadores nos quais a semente foi configurada com os valores 5, 10, 15 e 20 respectivamente em cada um, tenham sido construídos utilizando o conjunto de treinamento citado anteriormente. Em seguida a instância *I* é classificada sobre cada um dos modelos criados (o resultado da classificação pode ser observado na Tabela 4.1). Por causa da variação da semente a precisão foi afetada e a classe *A* foi selecionada apenas quando a semente foi configurada com o valor 15, o que representa 25% de acerto para este pacote de sementes.

Observamos que no conjunto de 5, 10 e 20 sementes, a classe *A* recebeu no mínimo 10% e no máximo 30% dos votos. Quando observado o resultado para a semente com o valor 15, na qual a classificação foi efetiva, a classe *A* recebeu 100% dos votos como esperado.

Combinando os resultados do pacote de 5, 10, 15 e 20 sementes computamos o seguinte resultado que pode ser observado na coluna *total*: 160/400% dos votos para a classe *A*, 90/400% para a classe *B* e 150/400% para a classe *C*. Como a classe *A* foi a mais votada sobre todos os resultados está será selecionada como resultado da votação de este pacote de sementes. Este exemplo nos mostra como o problema da sensibilidade pode ser solucionado combinando uma série de resultados.

Tabela 4.1: Resultado da classificação estocástica.

classe	*5 sementes*	*10 sementes*	*15 sementes*	*20 sementes*	*total*
A	10%	20%	100%	30%	160/400%
B	40%	30%	0%	20%	90/400%
C	50%	50%	0%	50%	150/400%

Na próxima seção executaremos um experimento utilizando o Random Forests Estocástico. O experimento será executado nos moldes do experimento realizado com o Random Forests apresentado anteriormente, na seção 3.1. Estudaremos o comportamento do Random Forests Estocástico quando submetido à aleatoriedade e compararemos estes resultados com os do Random Forests.

4.2 Definição do experimento utilizando aleatoriedade

Para estudar o comportamento do Random Forests Estocástico, planejamos e executamos um novo experimento, que será apresentado a seguir. Especificamos o experimento atual com base no experimento da Seção 3.1. Para adaptar o experimento ao nosso método algumas características do experimento foram alteradas.

Para a construção dos modelos e execução dos testes, utilizamos as mesmas bases de dados utilizadas no experimento da Seção 3.1. A classificação foi executada utilizando a técnica de *ten-folds stratified cross-validation*. Utilizando uma quantidade de classificadores *C* e um número de sementes *S*, que varia dependendo do pacote de sementes. Cada base de dados foi classificada utilizando 10, 30 e 50 classificadores. No experimento com Random Forests foram utilizadas 100 configurações de sementes, variando de 1 até 100. No presente experimento utilizaremos 100 pacotes compostos de 10 sementes cada. Os resultados da classificação das 10 sementes de cada pacote serão combinados formando um único resultado. Os pacotes de sementes serão compostos por sequência de números como por exemplo, *[Pacote1 = 1..10, Pacote2 = 2..11, ... Pacote100 = 100..109]*. A Tabela 4.2 mostra como o resultado de um pacote será consolidado.

Por uma questão estatística foram removidos os 5 maiores e os 5 menores resultados de cada base de dados. Com isso tratamos de evitar que resultados distantes causem uma variação muito grande na hora de calcular a média de acerto. Os resultados do experimento podem ser observados na tabela 4.3. Estes servirão como base para análise e posterior comparação com os resultados obtidos

no experimento do Random Forests. O código fonte e os dados resultantes do experimento aqui apresentado estão disponíveis em formato digital na biblioteca da Pontifícia Universidade Católica do Rio Grande do Sul. Na próxima seção apresentaremos os resultados do experimento aqui descrito.

Tabela 4.2: Resultado da classificação estocástica para um pacote de sementes.

semente	*classe A*	*classe B*	*classe C*
1	60%	30%	10%
2	50%	30%	20%
3	40%	10%	50%
4	50%	30%	20%
5	70%	10%	20%
6	50%	30%	20%
7	40%	30%	30%
8	60%	10%	30%
9	30%	40%	30%
10	50%	30%	20%
total	500‰	250‰	250‰

4.3 Análise dos Resultados Experimentais

Na presente seção analisaremos o comportamento do Random Forests Estocástico utilizando como base o resultado do experimento da Seção 4.2, que pode ser observado na Tabela 4.3. O estudo analisa os resultados por diferentes aspectos, comprando os resultados do Random Forests Estocástico com os obtidos no experimento utilizando Random Forests. Na sequência, compararemos estes resultados com os de Bagging e Boosting. Analisaremos os resultados subdividindo as bases de dados, utilizando como critério a quantidade de instâncias. Outro critério de subdivisão para análise será o desbalanceamento das bases de dados. Por último, estudaremos casos onde comportamentos excepcionais foram identificados.

4.3.1 Análise comparativa: Random Forests Estocástico versus Random Forests

Observando os resultados da Tabela 4.4, a qual consolida o resultado do experimento para 10, 30 e 50 classificadores, identificamos que o resultado da distância foi diminuído consideravelmente em favor do Random Forests Estocástico com relação ao Random Forests. A distância para o Random Forests foi de 7.9978% e o do Random Forests Estocástico de 3.8363%, o que representa uma diferença de 4.1613%. Para o desvio padrão também encontramos uma diferença significativa: para o Random Forests foi de 2.1833%, enquanto que para o Random Forests Estocástico foi de 1.0523%, o que resulta em uma diminuição de 1.1310%. Este resultado mostra que o Random Forests Estocástico conseguiu, neste caso, menor variabilidade nos resultados quando a semente foi alterada. Observamos também a diminuição no percentual de acerto, sendo este no Random Forests de 77.6436% e no Random Forests Estocástico de 76.2364%, o que resultou em uma vantagem de

Tabela 4.3: Resultado da classificação para Random Forests Estocástico com 10, 30 e 50 classificadores.

	100 Pacotes com 10 sementes cada								
Base de Dados	10 classificadores			30 classificadores			50 classificadores		
id	*média de acerto*	*desvio padrão*	*distância*	*média de acerto*	*desvio padrão*	*distância*	*média de acerto*	*desvio padrão*	*distância*
B01	23.0197	3.5885	0.7790	23.4476	2.8708	0.7078	23.5433	2.1531	0.5759
B03	78.2609	8.6957	1.3035	78.2609	0.0000	0.0000	78.2609	0.0000	0.0000
B04	6.4550	1.5873	0.3982	6.3492	0.0000	0.0000	6.3492	0.0000	0.0000
B05	50.9962	10.3448	2.2823	53.0651	10.3448	2.1744	52.8352	10.3448	2.4698
B06	93.7123	4.0462	0.7084	93.6095	2.3121	0.4449	93.5645	1.7341	0.4083
B07	86.0000	0.0000	0.0000	86.0000	0.0000	0.0000	86.0000	0.0000	0.0000
B08	84.6154	0.0000	0.0000	84.6154	0.0000	0.0000	84.6154	0.0000	0.0000
B09	65.8025	22.2222	6.4997	67.1605	22.2222	2.8419	66.6667	0.0000	0.0000
B10	46.0131	14.7059	4.3044	44.6405	11.7647	2.4381	44.1176	0.0000	0.0000
B11	61.1905	21.4286	5.5690	62.0635	21.4286	4.4978	61.1111	7.1429	3.5692
B12	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000
B13	87.9928	6.4516	1.5313	87.0968	0.0000	0.0000	87.0968	0.0000	0.0000
B14	82.5926	11.1111	3.5311	84.0329	7.4074	2.3973	84.5267	7.4074	2.0458
B15	91.1806	12.5000	3.2313	92.8472	6.2500	2.2094	92.9861	6.2500	2.0586
B16	81.9549	0.9183	0.2425	81.9886	1.0101	0.2765	81.9631	0.6428	0.1646
B17	99.3576	0.6250	0.1836	99.2813	0.6250	0.2220	99.2500	0.3125	0.1540
B18	91.4905	2.4390	1.2260	92.6829	0.0000	0.0000	92.6829	0.0000	0.0000
B19	77.5613	3.8961	1.0844	77.8788	3.8961	0.9224	78.0087	2.5974	0.7232
B20	45.9259	11.1111	3.7982	44.4444	0.0000	0.0000	44.4444	0.0000	0.0000
B21	50.4902	5.8824	1.9841	50.8497	5.8824	1.6640	51.6993	2.9412	1.4608
B22	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000
B23	82.2862	1.4388	0.4892	82.6059	0.7194	0.2766	82.6459	0.7194	0.2370
B24	95.0000	6.2500	1.6760	95.8565	4.1667	1.0237	96.3194	4.1667	0.8438
B25	98.9830	0.7937	0.2095	99.1358	0.2646	0.1176	99.1681	0.2646	0.0935
B26	98.3413	0.3571	0.1719	98.2937	0.3571	0.1493	98.2262	0.3571	0.0645
B27	97.0838	0.9464	0.1789	97.0032	0.9464	0.2231	96.9751	0.9464	0.2312
B28	95.5361	1.7544	0.8786	96.4912	0.0000	0.0000	96.4912	0.0000	0.0000
B29	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000	100.0000	0.0000	0.0000
B31	32.4385	3.3557	0.7185	32.3117	2.0134	0.3839	32.2148	0.0000	0.0000
B32	77.0707	18.1818	4.9565	79.1919	9.0909	4.1435	76.9697	9.0909	4.5608
média	76.0450	5.8210	1.5978	76.3734	3.7857	0.9038	76.2910	1.90237	0.6553

1.4072% em favor do Random Forests. A diferença encontrada é inferior a 2%, margem que utilizamos para considerar que um algoritmo é melhor do que o outro.

Tabela 4.4: Consolidação dos resultados para 10, 30 e 50 classificadores.

método	*média de acerto*	*distância*	*desvio padrão*
Random Forests Stochastic	76.2365	3.8363	1.0523
Random Forests	77.6436	7.9978	2.1833
Bagging	80.1165	1.9259	0.5000
Boosting	79.3448	2.9170	0.7500

Como podemos observar na Tabela 4.5, em nenhum dos três casos (10, 30 e 50 classificadores), a diferença na média de acerto superou os 2%. Utilizando 10 classificadores, a média de acerto para o Random Forests foi de 77.1359% e de 76.045% para o Random Forests Estocástico, o que resulta em uma diferença de 1.0945%. Utilizando 30 classificadores, o resultado foi de 77.7869% para o Random Forests e 76.3734% para o Random Forests Estocástico o que resulta em uma diferença de 1.4135%. Finalmente, utilizando 50 classificadores, o valor encontrado para o Random Forests foi de 78.0079% e para o Random Forests Estocástico de 76.2910%, resultando em 1.7169% de diferença. Observamos que apesar da queda no percentual de média de acerto do Randon Forests Estocástico esta não é suficiente para considerar este menos preciso que o Random Forests, se considerada a margem de 2%.

Anteriormente mostramos que a diferença média entre a distância foi de 4.1613% a menos em favor do Random Forests Estocástico com relação ao Random Forests. Esta diferença nos permite afirmar que o Random Forests Estocástico para as bases de dados utilizadas nos experimento gerou valores mais próximos, mostrando que é menos sensível ao efeito causado pela seleção randômica da semente, problema encontrado no Random Forests.

Analisando os resultados individualmente pela quantidade de classificadores utilizados (podem ser observados na Tabela 4.5), notamos que com 10 classificadores a distância foi de 5.8210% para o Random Forests Estocástico e de 10.0457% para o Random Forests, o que resulta em uma diferença de 4.2247% a menos em favor do Random Forests Estocástico. Quando utilizados 30 classificadores, o valor foi de 3.7857% para o Random Forests Estocástico e de 7.4487% para o Random Forests, resultando em uma diferença de 3.6630% a menos em favor do Random Forests Estocástico. Finalmente, para 50 classificadores, o valor do Random Forests Estocástico foi de 1.9023% e de 6.4991% para o Random Forests resultando em uma diferença de 4.5968% a menos em favor do Random Forests Estocástico. A análise acima nos mostrou melhoria na estabilidade do Random Forests Estocástico, obtendo resultados mais próximos da média de acerto.

Os valores de desvio padrão para 10 classificadores foram de 1.5978% para o Random Forests Estocástico e de 2.6949% para o Random Forests, o que resultou em uma diferença de 1.0971% em favor do Random Forests Estocástico. Utilizando 30 classificadores, o desvio padrão foi de 0.9038% para o Random Forests Estocástico e de 2.0536% para o Random Forests, resultando em uma vantagem de 1.1498% em favor do Random Forests Estocástico. Finalmente, utilizando 50 classificadores, o desvio padrão para o Random Forests Estocástico foi de 0.6553% e para o Random Forests foi de 1.8015%, resultando em uma diferença de 1.1462% em favor do Random Forests Estocástico. Observamos que quando o número de classificadores aumentou de 10 para 30, a diferença entre o desvio padrão do Random Forests Estocástico e do Random Forests aumenta mais do que quando incrementado o número de classificadores de 30 para 50, onde este valor se manteve um pouco mais próximo. Esta observação nos mostra que em todas as diferentes configurações de classificadores, o Random Forests Estocástico obteve resultados significativos no desvio padrão se comparado com o Random Forests.

Tabela 4.5: Resultado consolidado pelo número de classificadores.

	10 classificadores			30 classificadores			50 classificadores		
método	*média de acerto*	*distância*	*desvio padrão*	*média de acerto*	*distância*	*desvio padrão*	*média de acerto*	*distância*	*desvio padrão*
Random Forests Estocástico	76.0450	5.8210	1.5978	76.3734	3.7857	0.9038	76.2910	1.9023	0.6553
Random Forests	77.1359	10.0457	2.6949	77.7869	7.4487	2.0536	78.0079	6.4991	1.8015
Bagging	79.8339	2.4648	0.6534	80.2213	1.8116	0.4611	80.2944	1.5013	0.3927
Boosting	78.8702	3.5389	0.9002	79.5269	2.6797	0.6880	79.6373	2.5323	0.6509

4.3.2 Análise comparativa: Random Forests Estocastico versus Random Forests com relação a Bagging e Boosting

A próxima avaliação analisa o comportamento do Random Forests Estocástico e do Random Forests em relação a Bagging e Boosting. Para tanto, observaremos as Tabelas 4.4, que possui os

resultados consolidados para 10, 30 e 50 classificadores, e 4.5, que apresenta os resultados separados pelo número de classificadores. Identificamos anteriormente que quando observada a média de acerto, o Random Forests Estocástico obteve resultados menos performáticos que o Random Forests. Se comparado com Boosting, que obteve uma média de acerto de 79.3448%, o Random Forests obteve 77.6436%, o que resultou em uma diferença de 1.6985%. Quando comparamos Boosting com o Random Forests Estocástico, o último obteve uma média de acerto de 76.2364%, resultando em uma diferença de 3.1084%. Quando comparado com Bagging, que obteve um percentual de acerto médio de 80.1165%, observamos que a diferença de média de acerto em comparação ao Random Forests foi de 2.4729%, e a diferença com relação ao Random Forests Estocástico foi de 3.8801%. Notamos que os resultados observados do Random Forests Estocástico são menos eficientes quando comparados com os resultados de Bagging e Boosting, enquanto que se comparados os resultados de Random Forests com os de Bagging e Boosting, obteve-se resultados mais próximos.

Identificamos anteriormente uma diferença considerável no valor da distância em favor do Random Forests Estocástico em comparação com o Random Forests. Em comparação aos de Bagging e Boosting, a distância para o Random Forests Estocástico foi de 3.8363%, o que resultou em uma diferença de 1.91045% em comparação ao Bagging, que obteve uma distância de 1.9259%. Diferença esta menor que a diferença de 6.0719%, encontrada com relação a Random Forests, que obteve uma distância de 7.9978%. Comparando estes resultados com Boosting, que obteve uma distância de 2.9170%, encontramos uma pequena diferença de 0.9193% a menos em favor deste com relação ao Random Forests Estocástico. Esta mesma comparação feita para o Random Forests resultou em uma vantagem de 5.8008% em favor de Boosting.

Observando a Tabela 4.4, identificamos que o desvio padrão para Bagging foi de 0.5000% e de 0.7500% para Boosting, enquanto que para o Random Forests o valor foi de 2.1833%. Quando observamos o resultado do Random Forests Estocástico, encontramos 1.0532% de desvio padrão, o que resulta em uma diferença menor com relação à Bagging e Boosting quando comparado com Random Forests.

Esta análise considerou três variáveis (média de acerto, distância e desvio padrão) para comparar os resultados do Random Forests Estocástico e Random Forests. Inicialmente, comparamos os dois algoritmos, e em um segundo momento comparamos os resultados destes com os resultados de Bagging e Boosting. Identificamos uma pequena desvantagem do Random Forests Estocástico na média de acerto com relação ao Random Forests, e uma significativa vantagem com relação à distância e ao desvio padrão. A seguir, analisamos os resultados contabilizando os casos nos quais um algoritmo foi mais performático que o outro.

4.3.3 Comparação quantitativa

Observando o resultado do experimento por outra perspectiva, analisaremos o número de vezes que o Random Forests Estocástico foi superior ao Random Forests, e vice-versa. Comparando os resultado de todas as bases de dados e considerando a margem de 2% utilizada anteriormente em outras observações, identificamos que na média de acerto o Random Forests foi superior em 12/90

casos, enquanto que o Random Forests Estocástico foi em 8/90. Observando o desvio padrão, o Random Forests Estocástico foi melhor em 16/90 casos e o Random Forests em 0/90. Por último, para a distância, contabilizamos que o Random Forests Estocástico foi melhor em 44/90 casos e o Random Forests em 0/90.

Em relação ao número de classificadores, utilizando 10 classificadores, encontramos que na média de acerto o Random Forests foi melhor em 4/30 casos, e o Random Forests Estocástico em 2/30. O resultado do desvio padrão indicou que o Random Forests Estocástico foi melhor em 5/30 casos e o Random Forests em 0/30. Na distância, o Random Forests Estocástico foi melhor em 16/30 casos contra 0/30 do oponente. Classificando com 30 classificadores, o Random Forests foi melhor em 4/30, e o Random Forests Estocástico em 3/30 dos casos para a média de acerto. No desvio padrão, o Random Forests Estocástico foi melhor em 6/30 casos e o Random Forests 0/30. Na distância, o Random Forests Estocástico foi melhor em 14/30 casos e o Random Forests em 0/30. Finalmente, utilizando 50 classificadores, notamos que a média de acerto do Random Forests foi melhor em 4/30 casos e o Random Forests Estocástico em 3/30. O resultado para o desvio padrão mostra que o Random Forests Estocástico foi melhor em 5/30, e o Random Forests em 0/30. O resultado da distância para 50 classificadores mostra que o Random Forests Estocástico foi melhor em 14/30 resultados e o Random Forests em 0/30.

Quando analisado o resultado da distância e do desvio padrão para estas bases de dados, concluímos que o Random Forests Estocástico é consideravelmente menos sensível que o Random Forests. Apesar do Random Forests Estocástico estar dentro da margem de 2% em 70/90 casos na média de acerto, para esta medida o Random Forests obteve vantagem. Analisando separadamente pelo número de classificadores, notamos um comportamento parecido para as três configurações, sendo que o Random Forests foi levemente mais preciso na média de acerto. Analisando o desvio padrão e a distância, notamos uma ampla vantagem em favor do Random Forests Estocástico, comprovando que este é mais estável que o Random Forests. A próxima comparação de resultado será nos moldes da apresentada aqui, porém não levaremos em conta a margem de 2%.

4.3.4 Comparação quantitativa sem considerar margem de distância entre resultados

Diferentemente da desvantagem encontrada no percentual de média de acerto do Random Forests Estocástico, apresentada na análise da Subseção 4.3.3, nos resultados que apresentaremos a seguir encontraremos vantagens a favor. Observando a média de acerto, contabilizamos que o Random Forests Estocástico foi melhor em 48/90 casos, enquanto que e o Random Forests foi melhor em 32/90. Contrário ao resultado apresentado aqui, a Tabela 4.4 mostra desvantagem na média do Random Forests Estocástico, fato causado pelo baixo desempenho em algumas bases de dados. Para uma melhor visualização e entendimento do resultado apresentado nesta subseção, os gráficos exibidos nas Figuras 4.2, 4.3 e 4.4 servem como auxílio. Continuando com o nosso estudo, encontramos que a diferença do desvio padrão aumentou, sendo o Random Forests Estocástico melhor em 80/90 casos e o Random Forests em 0/90. Na contabilização da distância, notamos que o Random Forests Estocástico foi superior em 69/90 casos e o Random Forests em 2/90. Identificamos, para

este caso, que à medida que o número de classificadores aumentou, a diferença na contabilização dos resultados do percentual de acerto diminuiu. Por exemplo, comparando o percentual de media de acerto para 10 classificadores o Random Forests Estocástico foi melhor em 22/30 casos e o Random Forests em 7/30. Utilizando 30 classificadores, o Random Forests Estocástico foi melhor em 14/30 casos e o Random Forests em 12/30, e finalmente, para 50 classificadores, o Random Forests foi melhor em 13/90 casos e o Random Forests Estocástico em 12/30.

Analisando o desvio padrão e distância, considerando o número de classificadores, encontramos uma ampla vantagem em favor do Random Forests Estocástico. Observando o desvio padrão e utilizando 10 classificadores o Random Forests Estocástico foi melhor em 20/30 casos e o Random Forests em 0/30. Utilizando 30 classificadores, o primeiro foi melhor em 26/30 casos e o segundo em 0/30. Finalmente para 50 classificadores, o Random Forests Estocástico foi melhor em 25/30 casos e o Random Forests em 0/30 casos. Comparando a distância, o Random Forests Estocástico foi melhor em todos os casos. Utilizando 10 classificadores, o Random Forests Estocástico foi melhor em 26/30 casos contra 0/30 do oponente. Utilizando 30 classificadores, o Random Forests Estocástico foi melhor em 20/30, casos contra 1/30 do Random Forests. Utilizando 50 classificadores, o primeiro foi melhor em 23/30 casos e o segundo se manteve com 1/30 dos casos.

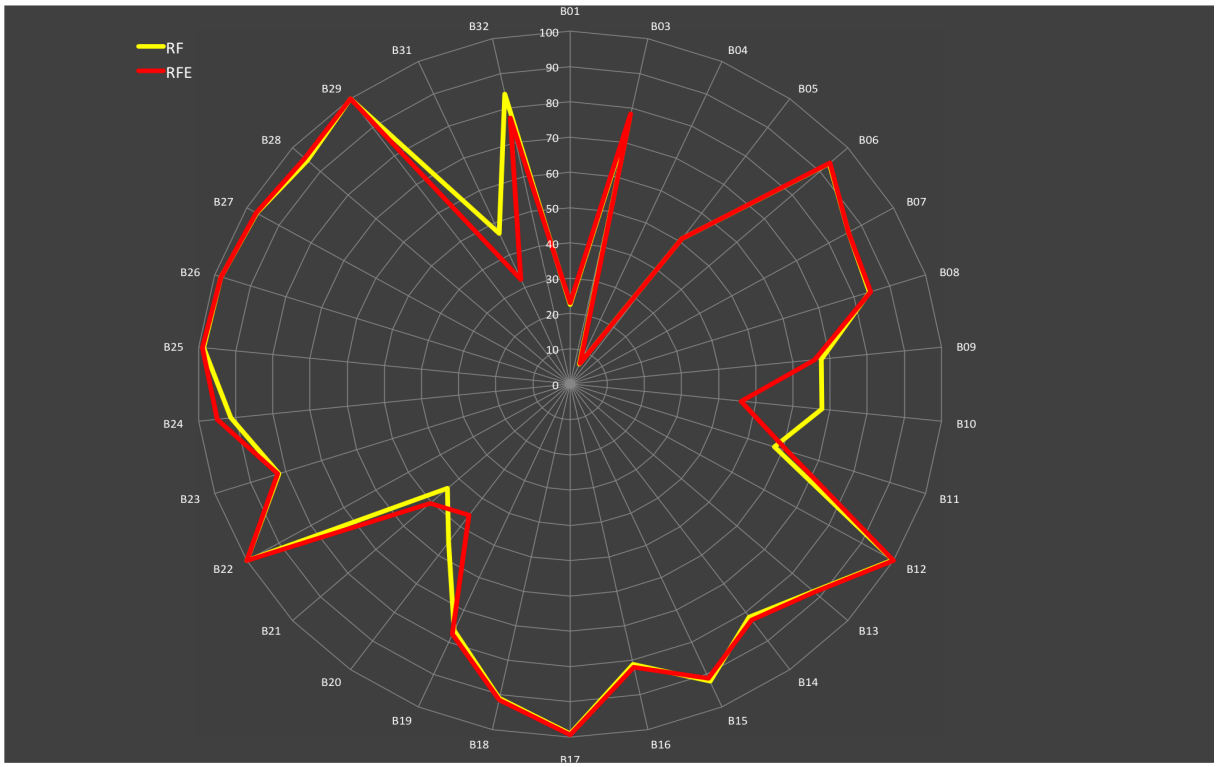

Figura 4.2: Gráfico comparativo da média de acerto dos resultados experimentais de Random Forests Estocástico versus Random Forests utilizando 10 classificadores.

Esta análise mostra que apesar da desvantagem do Random Forests Estocástico na média de acerto e quando considerado o percentual de 2%, no geral, este obteve um melhor rendimento. Quando comparados os resultados individualmente, notamos que quando não foi considerado o percentual de 2% o resultado da média de acerto foi melhor que o do Random Forests em todas as

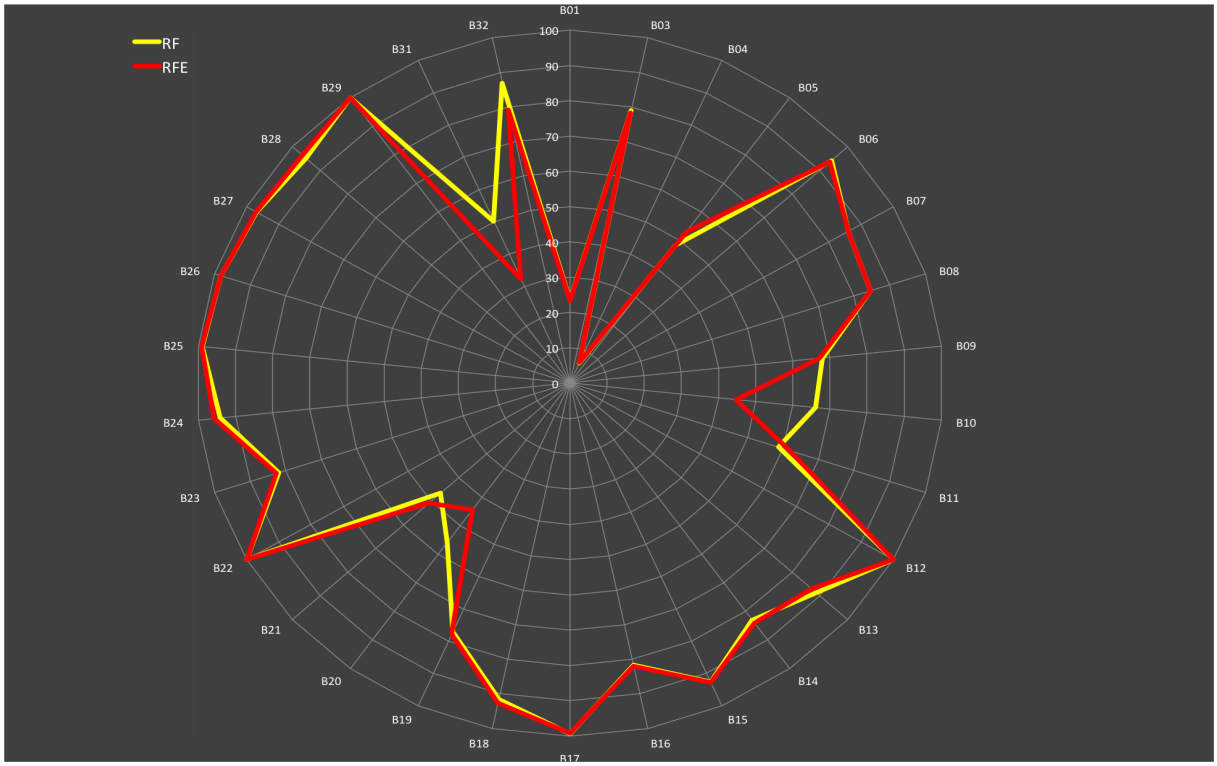

Figura 4.3: Gráfico comparativo da média de acerto dos resultados experimentais de Random Forests Estocástico versus Random Forests utilizando 30 classificadores.

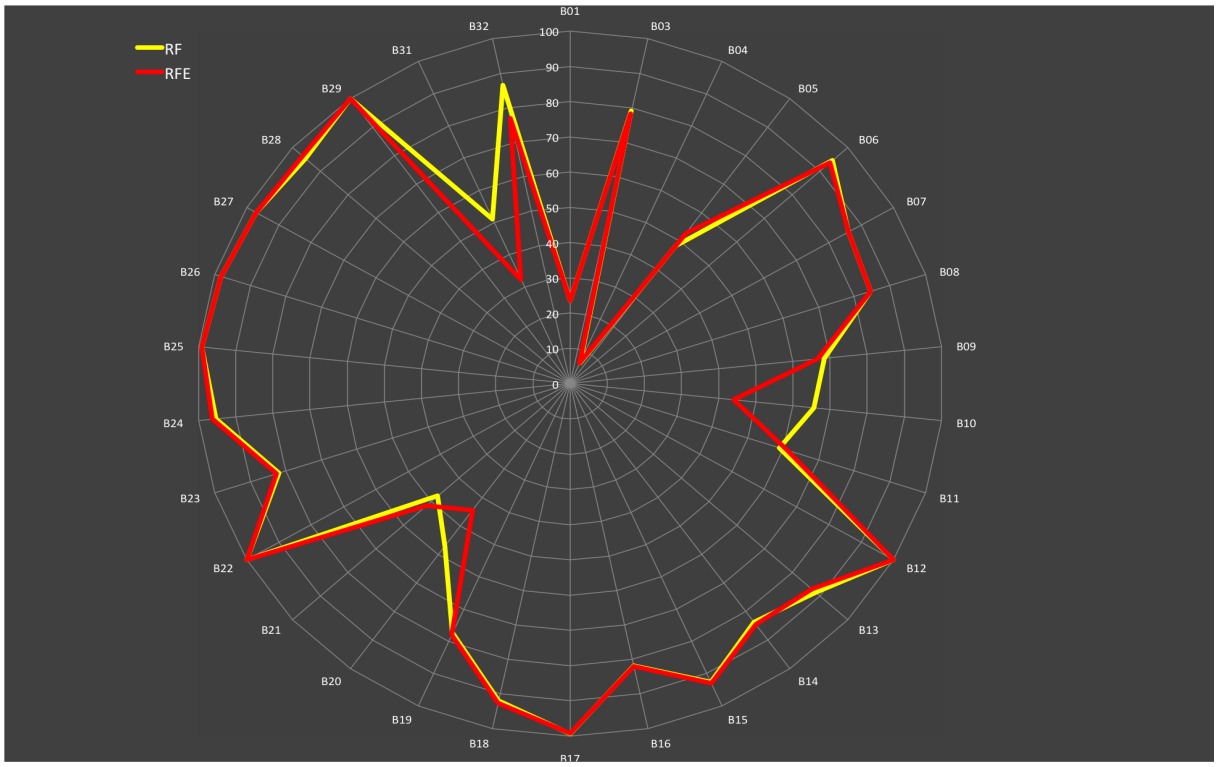

Figura 4.4: Gráfico comparativo da média de acerto dos resultados experimentais de Random Forests Estocástico versus Random Forests utilizando 50 classificadores.

comparações, assim como os resultados de desvio padrão e distância.

Na próxima seção observaremos os resultados experimentais que contemplam desde aspectos como desbalanceamento e quantidade de instâncias por base de dados. Buscamos com isto identificar se alguma destas características exerce alguma influência sobre os resultados.

4.3.5 Análise considerando características das bases de dados

O objetivo da análise seguinte é comparar o comportamento dos algoritmos considerando a quantidade de instâncias e o desbalanceamento. Esta comparação será elaborada separadamente. Para comparar os resultados pela quantidade de instâncias e pelo desbalanceamento, separamos as bases de dados em três grupos para cada observação:

- Grupos de bases de dados para comparar pelo desbalanceamento:
  - Grupo A: desbalanceamento de 0.0001 até 0.0999.
  - Grupo B: desbalanceamento de 0.1000 até 0.2999.
  - Grupo C: desbalanceamento de 0.3000 em diante.
- Grupos de bases de dados para comparação pelo número de instâncias:
  - Grupo D: de 1 até 300 instâncias.
  - Grupo E: de 301 até 1000 instâncias.
  - Grupo G: de 1001 em diante.

Desbalanceamento

A análise foi elaborada sobre o resultado geral e sobre o resultado individual, considerando o número de classificadores. Analisando o resultado geral das bases de dados do Grupo A, encontramos que tanto para o percentual de acerto, como para desvio padrão e distância, o Random Forests Estocástico foi melhor em todos os casos. Comparando a média de acerto do Random Forests Estocástico, esta foi de 79.6299%, e para o Random Forests de 78.5149%. A distância encontrada foi de 4.0602% para o primeiro e 7.4724% para o segundo. Apesar da diferença de 3.4122% encontrada na distância, a diferença do desvio padrão não foi tão alta. O desvio padrão do Random Forests Estocástico foi de 1.1523%, enquanto que o do e do Random Forests foi de 1.9100%.

Observando individualmente pelo número de classificadores, encontramos o seguinte resultado. Utilizando 10 classificadores, o Random Forests Estocástico foi melhor que o Random Forests na média de acerto, com 79.3400% contra 77.7173%, na distância foi de 5.5443% contra 8.7999%, e no desvio padrão foi de 1.5651% contra 2.2363%. Quando utilizados 30 classificadores, o Random Forests Estocástico manteve a vantagem em todos os resultados, com 79.7271% contra 78.7594% de percentual de acerto, o resultado da distância foi de 4.2070% contra 7.1739%, e o desvio padrão de 1.0395% contra 1.8290%. Utilizando 50 classificadores para este conjunto de bases de dados,

o resultado encontrado manteve-se parecido com os encontrados aqui anteriormente. A média de acerto para o Random Forests Estocástico foi de 79.8224% contra 79.0680%, a distância foi de 2.4292% contra 6.4431% do Random Forests e, finalmente, o desvio padrão foi de 0.8520% do primeiro contra 1.6646% do segundo. Para este conjunto de bases de dados, encontramos uma diferença não muito significativa em favor do Random Forests Estocástico no percentual de média de acerto e no desvio padrão. Observando a distância encontramos uma diferença um pouco maior de 3.4121%.

O resultado encontrado no Grupo A não se confirma no Grupo B, como mostraremos a seguir. Encontramos uma queda considerável na média de acerto e na distância entre elas. O Random Forests Estocástico obteve uma média de acerto de 57.3096%, e o Random Forests de 62.9061%. Observando individualmente o resultados das bases de dados do Grupo B, encontramos duas diferenças consideráveis que causaram esta vantagem em favor do Random Forests. Para a base de dados B32, a média de acerto do Random Forests foi de 85.8249% e para o Random Forests Estocástico de 77.7442%. Já para a base de dados B31 a média de acerto para o primeiro foi de 49.3463%, e para o segundo 32.3117%. Verificando as características das bases de dados B31 e B32 notamos que a diferença entre o desbalanceamento de ambas não é grande, já que o da primeira é de 0.1370 e o da segunda 0.1140. Para os demais resultados deste grupo encontramos um comportamento similar ao do Grupo A.

O Random Forests Estocástico obteve uma média de 6.4252% de distância, e o Random Forests de 12.5900%. O primeiro obteve um desvio padrão de 1.6448%, e o segundo de 3.5610%. Este último resultado mostra que o comportamento sobre estes resultados se manteve tanto para o Grupo A como para o Grupo B. Os resultados encontrados individualmente para os distintos conjuntos de classificadores mostram um resultado igual ao resultado geral, no qual o Random Forests obteve vantagem no percentual de média de acerto e o Random Forests Estocástico obteve vantagem na distância e desvio padrão. Quando 10 classificadores foram utilizados, o Random Forests obteve 62.4656% de média de acerto contra 57.1296% do Random Forests Estocástico. O Random Forests Estocástico obteve vantagem na distância, com 9.8867% contra 16.2474%, e no desvio padrão com 2.5579% contra 4.4757%.

O mesmo comportamento encontramos para 30 classificadores. O Random Forests obteve uma média de acerto de 63.0544% contra 57.6224% do Random Forests Estocástico. A distância do Random Forests Estocástico foi de 6.9296% contra 11.7424%, e o desvio padrão foi de 1.4978% contra 3.3552% do Random Forests. Finalmente, utilizando 50 classificadores encontramos uma vantagem em favor do Random Forests no percentual de média de acerto, de 63.1945% contra 57.1768%. No presente caso, o Random Forests Estocástico também obteve vantagem na distância, com 2.4295% contra 9.7893%, e no desvio padrão, com 0.8788% contra 2.8523%. Os resultados observados no Grupo B nos mostram que a vantagem em favor do Random Forests Estocástico se manteve na distância e no desvio padrão, porém encontramos uma vantagem considerável em favor do Random Forests no percentual de média de acerto. Analisando detalhadamente o resultado de cada base de dados, percebemos que ao classificar as bases B31 e B32 o Random Forests Estocástico

foi notoriamente menos preciso que o Random Forests. Este resultado exerceu um peso grande no resultado final do grupo na média de acerto.

Os resultados encontrados para o Grupo C mostram um comportamento similar ao do Grupo A. A única diferença, neste caso, é uma leve vantagem em favor do Random Forests no percentual de média de acerto. A média de acerto, considerando as três configurações do Random Forests Estocástico, foi de 86.6082%, e para o Random Forests, de 87.4903%. A distância para o primeiro foi de 1.7371%, e para o segundo de 5.1836%. O desvio padrão encontrado foi de 0.5215% para o Random Forests Estocástico, e de 1.4548% para o Random Forests. Observando individualmente pelo número de classificadores utilizados, encontramos, como em casos anteriores, o mesmo comportamento, no qual o Random Forests foi melhor no percentual de média de acerto, e o Random Forests Estocástico no desvio padrão e na distância. Classificando com um conjunto de 10 classificadores, o Random Forests obteve uma média de acerto de 87.2237% contra 86.5067% do Random Forests Estocástico. A distância foi favorável ao Random Forests Estocástico, com 3.1410% contra 6.7810%, assim como o desvio padrão, com 0.9323% contra 1.8585% do Random Forests. O mesmo observamos para 30 classificadores. Random Forests obteve uma média de acerto de 87.5290% contra 86.6569%. Na distância, o Random Forests Estocástico obteve um melhor resultado, com 1.0781% contra 4.6008%, e para o desvio padrão a vantagem deste foi de 0.3361% contra 1.3317%. Por último, utilizando 50 classificadores, na média de acerto o Random Forests foi melhor com 87.7183% contra 86.6610%. Comparando a distância, notamos que o Random Forests Estocástico foi melhor, com 0.99225% contra 4.1689%. Por último, este também foi melhor no desvio padrão com 0.2962% contra 1.7428%. No Grupo C notamos um comportamento parecido com o do Grupo A porém com uma notória queda no desvio padrão e na distância, além de perceber um aumento no percentual de média de acerto.

Analisando o resultado desta observação e considerando o desbalanceamento das bases de dados, percebemos um comportamento padrão: à medida que aumentamos o número de classificadores durante o experimento, o percentual de desvio padrão e da distância diminuíram. Em alguns casos, como o do Grupo C, a distância do desvio padrão entre o Random Forests Estocástico e o Random Forests também aumentou à medida que os classificadores aumentaram. Excepcionalmente no Grupo A, identificamos que o Random Forests Estocástico foi levemente superior que o Random Forests na média de acerto, o que não ocorre nos demais grupos. Na distância e no desvio padrão, o Random Forests Estocástico foi melhor em todos os casos.

Especificamente no Grupo B, encontramos uma queda significativa no percentual de média de acerto. Investigamos a possível causa e identificamos que para as bases de dados B31 e B32, o resultado do Random Forests Estocástico é consideravelmente inferior ao do Random Forests. Este resultado não nos permite deduzir que o desbalanceamento tem influência no resultado por dois motivos. O primeiro é que para as demais bases de dados do grupo as médias de acerto são similares com as demais médias de acerto do experimento, estando estas próximas. O segundo é que no resultado do Grupo C, no qual o desbalanceamento é maior, encontramos um comportamento similar ao do Grupo A, o que mostra que esta característica não tem influência direta sobre os resultados.

Identificamos, no caso do Grupo B, que a média de acerto foi inferior à 22.3203% com relação ao Grupo A, e 29.2986% com relação ao Grupo C. Finalmente, notamos uma diferença importante entre o Grupo A e o Grupo C: estes grupos não foram afetados por nenhum comportamento excepcional, e identificamos que em todos os casos para 10, 30 e 50 classificadores no Grupo C, no qual o desbalanceamento foi mais alto, obteve-se uma melhor média de acerto, e distância e desvio padrão menores. Este resultado mostra que o desbalanceamento para estes casos não é uma característica que pode afetar o resultado da classificação. Destacamos que a afirmação anterior é válida para as bases de dados apresentadas na Tabela 3.1.

Número de instâncias

Esta nova análise observa os resultados dividindo as bases de dados pelo número de instâncias. Uma nova observação será realizada sobre o resultado de cada grupo. O resultado médio do conjunto de bases de dados Grupo D nos mostra um resultado que se confirma na maioria das observações. O Random Forest é melhor no percentual de média de acerto com 77.7561% contra 76.5373% do Random Forests Estocástico. O desvio padrão do Random Forests Estocástico foi de 1.9436% e do Random Forests de 3.5970%. Já a distância do primeiro foi de 7.0477% e do segundo 12.6529%. Os resultados individuais para cada conjunto de classificadores mostram o mesmo comportamento que o resultado geral. Utilizando 10 classificadores o Random Forests obteve uma média de acerto de 77.2668% contra 76.1487% do Random Forests Estocástico. O desvio padrão do Random Forests Estocástico foi de 2.8339% contra 4.2924%. Já a distância do primeiro foi de 10.5087% contra 15.5160% do segundo. Para 30 classificadores a vantagem em favor do Random Forests no percentual de acerto foi mantida, com 77.8879% contra 77.8802%. Este número de classificadores nos mostra uma vantagem no desvio padrão em favor do Random Forests Estocástico, de 1.6604% contra 3.4238%, mantendo a distância com 6.9767% contra 11.8323%. Finalmente, utilizando 50 classificadores, notamos uma distância levemente superior em favor do Random Forests no percentual de média de acerto, com 78.1136% contra 76.5832% do Random Forests Estocástico. O desvio padrão do Random Forests Estocástico foi melhor com 1.3367% contra 3.0759% do Random Forests, e a distância no caso foi de 3.6578% para o primeiro e de 10.6105% para o segundo.

Para o Grupo E encontramos um comportamento similar ao encontrado no Grupo D: o Random Forests Estocástico foi melhor na distância e desvio padrão e o Random Forests na média de acerto. A média de acerto do Random Forests foi de 75.0430% contra 73.7717% do Random Forests Estocástico. O desvio padrão do Random Forests Estocástico foi de 0.7386% contra 1.6702% do Random Forests, e a distância do primeiro foi de 2.6127% contra 6.6451% do segundo. Individualmente para cada conjunto de classificadores o comportamento foi igual ao resultado geral, não ocorrendo nenhuma exceção neste caso. Com 10 classificadores, o Random Forests obteve vantagem na média de acerto, com 74.5327% contra 73.6539% do Random Forests Estocástico. O desvio padrão do Random Forests Estocástico foi melhor, com 1.3083% contra 2.2188% do Random Forests, a distância do primeiro foi melhor, com 4.2968% contra 8.6660%. Observando o resultado para 30 classificadores a vantagem na média de acerto em favor do Random Forests foi de 75.1668% contra

73.4856% do Random Forests Estocástico. O desvio padrão do Random Forests Estocástico foi de 0.6048% contra 1.5203%, e a distância do primeiro foi de 2.5710% contra 6.3020% do segundo. Por último, utilizando 50 classificadores, o Random Forests obteve uma média de acerto de 75.4295% contra 73.8765%. O desvio padrão do Random Forests Estocástico foi de 0.3028% contra 1.2717% do Random Forests e a distância do primeiro foi de 0.9705% e do segundo 4.9674%.

No Grupo F ,agrupando os resultados de todos os conjuntos de classificadores, o Random Forests obteve uma média de acerto de 80.3955% contra 78.6076% do Random Forests. Observando o desvio padrão encontramos que o Random Forests Estocástico foi melhor, com 0.3116% contra 1.0257% do Random Forests, e na distância com 1.2711% contra 3.8113%. Individualmente para cada grupo de classificadores, notamos o mesmo resultado, no qual o Random Forests foi melhor na média de acerto e o Random Forests Estocástico na distância e desvio padrão. Utilizando 10 classificadores, o Random Forests obteve uma média de acerto de 79.8681% contra 78.5753% do Random Forests Estocástico. No desvio padrão, o Random Forests Estocástico obteve 0.4090% contra 1.2727% do Random Forests. A distância do Random Forests Estocástico foi de 1.7855% contra 4.8927% do Random Forests. Com 30 classificadores, o Random Forests obteve uma média de acerto de 80.5749% contra 78.6308% do Random Forests Estocástico. Para o desvio padrão, o Random Forests Estocástico manteve a vantagem, com 0.3113% contra 0.9715% do Random Forests. O primeiro também manteve a vantagem na distância, com 1.2354% contra 3.3652% do segundo. Finalmente, utilizando 50 classificadores, a média de acerto do Random Forests foi melhor, com 80.7436% contra 78.6168% do Random Forests Estocástico. Por último, o desvio padrão do Random Forests Estocástico foi de 0.2143% contra 0.8328%, e a distância do primeiro foi de 0.7922% contra 3.1762% do segundo.

Notamos que o número de instâncias não é uma característica que afeta o desempenho e a estabilidade do Random Forests Estocástico. Esta afirmação se apoia na seguinte observação: o Grupo F é o que possui as bases de dados com o maior número de instâncias, e obteve a melhor média de acerto, distância e desvio padrão quando comparando com o resultado dos demais grupos de bases de dados. Notamos também que o número de instâncias não tem efeito sobre a relação de média de acerto, distância e desvio padrão, já que no Grupo D a média de acerto foi maior que a do Grupo E o que não acontece com a distância e desvio padrão, que são menores no Grupo E. Quando comparamos o resultado do Random Forests Estocástico com o do Random Forests, notamos no resultado geral o mesmo comportamento em todos os grupos e casos. O Random Forests é melhor no percentual de média de acerto, e o Random Forests Estocástico na distância e desvio padrão. Analisando os resultados mais especificamente, não identificamos nenhum comportamento excepcional, o que mostra que o número de instâncias não influencia na estabilidade do algoritmo.

4.3.6 Estudo de exceções

Identificamos, nas análises anteriores uma mesma tendência em favor do Random Forests Estocástico nos resultados da distância e desvio padrão, e uma pequena vantagem em favor do Random Forests no percentual de média de acerto. Aqui a nossa análise é focada sobre resultados individuais

de cada base de dados, nos quais um resultado excepcional seja identificado. Separamos vinte e três resultados identificados sobre os resultados do experimento: oito referentes à percentual de média de acerto, quatro sobre desvio padrão e onze referentes à distância. Para selecionar os resultados utilizamos o seguinte critério:

- Percentual de média de acerto: diferença maior ou igual a 9% entre os classificadores.
- Desvio padrão: diferença maior ou igual a 5%.
- Distância: diferença maior ou igual a 9%.

Observando o percentual de média de acerto, identificamos comportamentos excepcionais para as bases de dados B10, B20 e B31. Todos os resultados identificados aqui para o percentual de média de acerto são em favor do Random Forests. Para a base de dados B10, encontramos dois resultados significativos: utilizando 30 classificadores, a média de acerto do Random Forests foi de 66.0784% contra 44.6405% do Random Forests Estocástico, resultando em uma diferença de 21.4379%: utilizando 50 classificadores, o primeiro obteve uma média de acerto de 65.7190% contra 44.1176% do segundo, resultando em uma diferença de 21.6014%. Para a base de dados B20 encontramos significativa diferença nas três configurações. Utilizando 10 classificadores o percentual de acerto do Random Forests foi de 55.4321% contra 45.9259%, resultando em uma diferença de 9.5062%; para 30 classificadores, o primeiro obteve uma média de 55.8025% contra 44.4444%, resultando em uma diferença de 11.3581%; por último, utilizando 50 classificadores, o Random Forests obteve de média de acerto de 57.0371% contra 44.4444% do Random Forests Estocástico, resultando em uma diferença de 12.5927%. Notamos que para o caso da base de dados B20, à medida que o número de classificadores foi aumentando, a diferença entre os resultados também aumentou.

Para a base de dados B31, encontramos, como na base B20, diferença nos três conjuntos de classificadores. Utilizando 10 classificadores, o percentual de acerto do Random Forests foi de 46.7652% contra 32.4385%, resultando em uma diferença de 14.3177%; com 30 classificadores a média de acerto do primeiro foi de 50.1541% contra 32.3117%, resultando em uma diferença de 17.8524%. Por último, para a base de dados B32 utilizando 50 classificadores configurados a média de acerto para o Random Forests foi de 51.1186% contra 32.2148%, resultando em uma diferença de 18.9083%. Para a base de dados B31, assim como para a base de dados B20, notamos que à medida que o número de classificadores aumenta, a diferença entre os resultados também aumenta. Os casos específicos de diferença considerável na média de acerto destacados aqui mostram uma ampla vantagem em favor do Random Forest. Notamos também que nos casos em que uma diferença grande foi encontrada nos três conjuntos de classificadores, à medida que o número de classificadores aumenta, a diferença entre os resultados também aumenta.

Sobre todos os resultados de desvio padrão destacamos quatro casos de notória diferença. Para a base de dados B09 utilizando 50 classificadores encontramos um desvio padrão de 0% para o Random Forests Estocástico, e de 5.3326% para o Random Forests. A base de dados B20 obteve os

seguintes resultados: com 30 classificadores o desvio padrão do Random Forests Estocástico obteve 0% de desvio padrão e o Random Forests 7.0623%, e utilizando 50 classificadores o primeiro também obteve 0% de desvio padrão contra 6.9078%. Por último, para a base de dados B31, o resultado do desvio padrão para o Random Forests Estocástico foi de 0.7185% contra 6.6485% do Random Forests resultando em uma diferença de 5.9300%. Notamos que para estes casos ocorreu um ganho significativo em favor do Random Forests Estocástico, mostrando que este consegue alcançar uma ótima estabilidade.

Analisando os resultados do experimento e o comportamento do Random Forests Estocástico com relação à distância, encontramos valores que mostram uma grande queda nos resultados gerados pelo Random Forests Estocástico. Dentre todos estes resultados destacamos onze que superaram os 9% de diferença. Observando o resultado da base de dados B08, encontramos que para 10 classificadores a distância do Random Forests Estocástico foi de 0%, contra 15.3846% do Random Forests. Ao classificar a base de dados B09 utilizando 50 classificadores, o Random Forests Estocástico obteve 0% de distância contra 22.2222% do Random Forests. A performance anterior manteve-se para a base de dados B10: utilizando 50 classificadores, o Random Forests Estocástico obteve 0% de distância, e o Random Forests 11.7647%. Para a base de dados B13 utilizando 30 classificadores, o Random Forests Estocástico obteve 0% de distância contra 9.6774%. O resultado do base de dados B11 utilizando 50 classificadores é um pouco distante dos anteriores, no qual o Random Forests Estocástico obteve 7.1429% contra 21.4286% do Random Forests resultando em uma diferença de 14.2857%. Detectamos, na observação, resultados significativos para os três conjuntos de classificadores para a bases de dados B20 e B31. Para a base de dados B20 encontramos os seguintes resultados: com 10 classificadores o Random Forests Estocástico obteve 11.1111% de distância e o Random Forests 22.2223%, resultando em uma diferença de 11.1112%. Utilizando 30 classificadores, o Random Forests Estocástico obteve 0% de distância contra 22.2223% do Random Forests. Finalmente, utilizando 50 classificadores, o primeiro manteve 0% de distância contra 22.2223% do segundo.

Os resultados da base de dados B31 mostram que com 10 classificadores, o Random Forests Estocástico obteve uma média de 3.3557% contra 24.8322% do Random Forests, resultando em uma diferença de 21.4765%. Utilizando 30 classificadores, o Random Forests Estocástico obteve uma média de 2.0134% contra 16.1074%, resultando em uma diferença de 14.0940%. Classificando com 50 classificadores, o primeiro obteve uma distância de 0% contra 15.4363% do segundo. Os resultados apresentados aqui mostram uma grande diferença entre os resultados de distância gerados pelo Random Forests Estocástico e pelo Random Forests. Os casos analisados mostram que o Random Forests Estocástico alcançou baixos percentuais de distância sendo estes valores consideravelmente menores que os valores de distância gerados pelo Random Forests.

5. CONCLUSÃO

Este trabalho apresentou um método de mineração de dados que objetiva diminuir a variabilidade encontrada nos resultados do método Random Forests quando submetido à aleatoriedade. Para alcançar o objetivo principal deste trabalho, foi desenvolvido um método chamado Random Forests Estocástico na Seção 4.1, e sua performance experimentada e avaliada nas Seções 4.2 e 4.3.

Como parte do desenvolvimento do trabalho, foi implementado o algoritmo junto com a especificação e exemplificação de como é possível obter ganhos de precisão e estabilidade. Esta produção cumpre com o primeiro objetivo específico desta dissertação. O segundo objetivo específico foi alcançado nas seções nas quais o Random Forests Estocástico foi experimentado, e sua precisão e estabilidade foram avaliadas. Para avaliar o comportamento do Random Forests Estocástico com relação ao do Random Forests, foram realizadas uma série de comparações sobre os resultados do experimento. Comparamos os resultados considerando e não considerando o número de classificadores. Comparamos ambos resultados com relação à Bagging e Boosting.

Buscando uma nova perspectiva de análise, realizamos uma comparação quantitativa dos resultados levando e não levando em conta a margem de 2%. Outro ponto importante da análise foi a divisão das bases de dados por características. Separamos as bases de dados por seu desbalanceamento, e pelo número de instâncias, e observamos se estas características influenciaram na performance e estabilidade do algoritmo. Por último, um estudo de exceções foi feito, no qual destacamos casos nos quais uma diferença considerável no resultado foi encontrado. Com relação aos resultados encontrados, o Random Forests Estocástico obteve vantagem em algumas comparações. As observações sobre os estudos citados anteriormente mostram que o Random Forests Estocástico obteve notória vantagem em relação ao Random Forests quando analisados os valores de desvio padrão e distância. Comparando os resultados de percentual de acerto, encontramos vantagem em favor do Random Forests. Na comparação quantitativa, o Random Forests foi melhor quando considerada a margem de 2% ao comparar os resultados, enquanto que o Random Forests Estocástico foi superior quando não utilizada esta margem.

Todos os resultados apresentados até aqui nos permitem concluir que dentro das bases de dados experimentadas no presente trabalho, o Random Forests Estocástico conseguiu uma menor variabilidade nos resultados quando comparado ao Random Forests.

Realizando algumas adaptações como por exemplo, a possibilidade de processar este de forma distribuída, permitirão que este possa ser utilizado em sistemas de recomendação.

5.1 Lições aprendidas

Este trabalho nos deixou os seguintes aprendizados:

- Um bom entendimento sobre Mineração de Dados, Combinção de Classificadores, Bagging e Boosting.

- Uma compreensão mais profunda sobre o Random Forests e o entendimento de que, além de melhorar a estabilidade, como alcançado no presente trabalho, outras melhorias em performance e estabilidade podem ser alcançadas. Estas possíveis melhorias abrem caminho para trabalhos futuros.

5.2 Contribuição trabalho

A contribuição principal do trabalho foi mostrar através da implementação e experimentação do método Random Forests Estocástico que é possível adaptar o método Random Forests para que este seja menos vulnerável à aleatoriedade.

Os objetivos específicos, que serviram para responder a pergunta da hipótese do presente trabalho, somados ao estudo inicial sobre o Random Forests, produziram três outras menores contribuições que claramente podem ser apontadas:

- Um estudo através de experimentos que mostra a variabilidade dos resultados do Random Forests quando submetido à aleatoriedade;
- A proposta e implementação do novo método Random Forests Estocástico;
- Um exprimento que serviu para medir a precisão e a variabilidade dos resultados do Random Forests Estocástico.

A primeira das contribuições menores citada acima serviu como apoio e justificativa para o desenvolvimento desta dissertação. Experimentamos e estudamos os resultados por diferentes aspectos. Inicialmente comparamos os resultados experimentais do Random Forests, considerando o número de sementes utilizadas. Comparando os resultados utilizando a semente *Default*, notamos que o Random Forests obteve melhor desempenho se comparado com o alcançado com a seleção aleatória da semente. Buscando identificar a origem da instabilidade encontrada, comparamos os resultados com os de Bagging e Boosting. Esta comparação nos mostrou que observando o percentual de media de acerto do Random Forests, encontramos resultados levemente inferiores aos resultados destes algoritmos.

Na sequência, aplicamos uma comparação quantitativa. Contabilizamos o número de vezes que um algoritmo foi melhor que os demais. O resultado desta comparação mostrou um melhor desempenho em favor do Random Forests. Apesar do melhor desempenho, notamos uma queda de performance quando aplicada a seleção aleatória da semente. Decidimos observar então o resultado do experimento por outro aspecto, e o fizemos analisando a distância e o desvio padrão. Ao comparar os resultados de desvio padrão e distância, encontramos uma notória diferença à favor de Bagging e Boosting em relação à Random Forests. Notamos que este gerou resultados de classificação muito distantes de distância e um desvio padrão muito alto quando comparado com os demais algoritmos. Este estudo nos mostrou através dos seus resultados que o Random Forests é sensível à seleção aleatória da semente, servindo este como base para a realização deste trabalho.

A implementação do Random Forests Estocástico foi a segunda menor contribuição. O desenvolvimento do método não implicou em alterar nenhuma funcionalidade interna do Random Forests, e sim utilizar as funcionalidades já existentes para o seu desenvolvimento. Apresentamos a fórmula para calcular os percentuais de voto e descrevemos seus componentes. O funcionamento do algoritmo foi comentado passo a passo, para um melhor entendimento. No fim, a estratégia para diminuir a sensibilidade causada pela aleatoriedade combinando diferentes resultados de classificação foi explicada e exemplificada.

A última contribuição de este trabalho é um estudo comparativo entre os resultados obtidos no experimento do Random Forests com os do Random Forests Estocástico. Inicialmente um experimento foi planejado e executado utilizando como base o primeiro cenário apresentado no presente trabalho. Coletadas as informações deste experimento, iniciamos as distintas análises. Comparamos os resultados por perspectivas diferentes. Analisamos as médias e comparamos estas de forma consolidada, de acordo com o número de classificadores. Comparamos os resultados de ambos métodos relacionando seus resultados com os de Bagging e Boosting. Uma comparação quantitativa foi realizada para conhecer o número de casos em que um algoritmo foi melhor que o outro. Os resultados foram analisados levando em conta diferentes características das bases de dados . Por ultimo, as exceções foram destacadas. O resultado de todas estas diferentes análises nos mostraram no geral um melhor despenho do Random Forests Estocástico no que diz respeito à distância e desvio padrão e vantagem no percentual de acerto em favor do Random Forests. A última afirmação só não foi verdadeira na comparação apresentada na Subseção 4.3.4, na qual a vantagem em favor do Random Forests Estocástico foi encontrada na comparação da média de acerto. Isto mostra que mesmo não conseguindo uma vantagem expressiva, capaz de superar os 2%, este foi mais preciso que o Random Forests na maioria dos casos.

5.3 Trabalhos Futuros

Dentro do contexto deste trabalho, do desenvolvimento do Random Forests Estocástico e do experimento que mostra os ganhos obtidos, foi possível identificar o seguinte trabalho futuro:

- Desenvolver melhorias no Random Forests Estocástico;

O desenvolvimento do item citado acima adicionaria novos mecanismos ao funcionamento atual do Random Forests Estocástico. Uma melhoria significativa seria possibilitar a configuração de um pacote de sementes N, sendo $N >= 1$, possibilitando assim a paralelização da construção dos modelos. Outra funcionalidade a ser adicionada seria a paralelização da classificação de uma instância sobre as N florestas construídas. As melhorias citadas anteriormente dizem respeito a questões de performance de execução das tarefas.

Bibliografia

[1] A. Asuncion and D. J. Newman. Uci machine learning repository. http://www.ics.uci.edu/∼mlearn/MLRepository.html, 2007.

[2] G. Boetticher, T. Menzies, and T. Ostrand. Promise repository of empirical software engineering data. http://promisedata.org/, 2007.

[3] P. Boinee, A. D. Angelis, , and G. L. Foresti. Meta random forests. In *International Journal of Computational Intelligence*, volume 2, pages pp. 138–147, 2005.

[4] L. Breiman. Bagging predictors. Technical Report 421, University of California, Berkeley, 1994.

[5] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, Aug. 1996.

[6] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, Oct. 2001.

[7] L. Breiman. Random forests. Technical Report 200, University of California, Berkeley, 2001.

[8] L. Breiman. Random forests - classification description. Capturado em:. `http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm`, Maio 2011.

[9] L. A. V. de Carvalho. *Data Mining in Marketing, Medicine, Economy, Engineering, and Business*. Erica Publisher, 2001.

[10] N. A. Diamantidis, D. Karlis, and E. A. Giakoumakis. Unsupervised stratification of cross-validation for accuracy estimation. *Artif. Intell.*, 116(1-2):1–16, Jan. 2000.

[11] T. G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Mach. Learn.*, 40(2):139–157, Aug. 2000.

[12] P. Fernandes, L. Lopes, and D. D. A. Ruiz. The impact of random samples in ensemble classifiers. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, SAC '10, pages 1002–1009, New York, NY, USA, 2010. ACM.

[13] A. A. Freitas. *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Springer, 2002.

[14] Y. Freund and R. E. Shapire. Experiments with a new boosting algorithm. In *Proc. of the 13th. Int. Conf. on Machine Learning*, pages 148–156, 1996.

[15] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th international joint conference on Artificial intelligence*

- *Volume 2*, IJCAI'95, pages 1137–1143, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[16] S. B. Kotsianti and D. Kanellopoulos. Combining bagging, boosting and dagging for classification problems. In *Knowledge-Based Intelligent Information and Engineering Systems and the XVII Italian Workshop on Neural Networks on Proceedings of the 11th International Conference*, KES '07, pages 493–500, Berlin, Heidelberg, 2007. Springer-Verlag.

[17] L. Kuncheva, M. Skurichina, and R. Duin. An experimental study on diversity for bagging and boosting with linear classifiers. *Information Fusion*, 3(4):245 – 258, 2002.

[18] L. Lopes, E. E. Scalabrin, and P. Fernandes. An empirical study of combined classifiers for knowledge discovery on medical data bases. In *APweb 2008 Workshops (LNCS 4977)*, pages 110–121, 2008.

[19] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[20] V. K. Pang-Ning Tan, Michael Steinbach. *Introduction to Data Mining*. Addison-Wesley, 2006.

[21] J. R. Quinlan. *C4.5: Programs for Machine Learning*, volume 16. Morgan Kaufmann Publishers, Hingham, MA, USA, Sept. 1994.

[22] R. Quinlan. Bagging, boosting, and C4. 5. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference*, pages 725–730, Menlo Park, August4–8 1996. AAAI Press / MIT Press.

[23] M. Robnik-Šikonja. Improving random forests. In *Machine Learning: ECML 2004*, volume 3201 of *Lecture Notes in Computer Science*, pages 359–370, 2004.

[24] R. E. Shapire, Y. Freund, P. Bartlett, and W. S. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26:1651–1686, 1998.

[25] J. Thongkam, G. Xu, and Y. Zhang. Adaboost algorithm with random forests for predicting breast cancer survivability. In *IJCNN*, pages 3062–3069, 2008.

[26] A. Tsymbal, M. Pechenizkiy, and P. Cunningham. Dynamic integration with random forests. In *Proceedings of the 17th European conference on Machine Learning*, ECML'06, pages 801–808, Berlin, Heidelberg, 2006. Springer-Verlag.

[27] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., 2005.

[28] B. Zenko, L. Todorovski, and S. Dzeroski. A comparison of stacking with meta decision trees to bagging, boosting, and stacking with other methods. In *Proceedings of the 2001 IEEE International Conference on Data Mining*, ICDM '01, pages 669–670, Washington, DC, USA, 2001. IEEE Computer Society.