Combinando Diferentes Estratégias de Mineração de Dados na Detecção de Arritmia Cardíaca

Christian Reis, Alan Cardoso, Diego Dias and
Elisa Tuler, and Leonardo Rocha

Universidade Federal de São João del-Rei
Minas Gerais - Brasil
{christian,alanc,etuler,diegodias,lcrocha}@ufsj.edu.br

Resumo A detecção de Arritmia Cardíaca (AC) é realizada por meio da análise clínica do Eletrocardiograma (ECG) de um paciente e é utilizada na prevenção de doenças cardiovasculares. Algoritmos de Aprendizagem de Máquina vêm se apresentando como ferramentas promissores no auxílio de diagnósticos de ACs, com destaque àqueles relacionados à classificação automática. Entretanto esses algoritmos sofrem com dois problemas clássicos relacionados à classificação: (1) número excessivo de atributos numéricos gerados a partir da decomposição de um ECG; e (2) o número de pacientes diagnosticados com ACs é muito menor do que aqueles classificados como normais acarretando em bases de dados muito desbalanceadas. Nesse trabalho avaliamos vários algoritmos de mineração de dados (*clustering*, *feature selection* e *oversampling*) combinados com técnicas de classificação automática no intuito de criar modelos de classificação mais eficazes para auxiliar especialistas na detecção da doença. Em nossas avaliações, utilizando uma base de dados disponibilizada pela UCI, a combinação dessas estratégias utilizando o algoritmo de classificação *Random Forest*, conseguimos uma taxa de acerto superior a 88%, um valor superior ao melhor já reportado na literatura.

Keywords: detecção de arritimia cardíaca, classificação automáttica

1 Introdução

Doenças cardiovasculares ainda são uma das principais causas de morte no mundo. Uma das principais anormalidades associadas à essas doenças é a Arritmia Cardíaca (AC), que pode ser detectada pelo especialista por meio de uma análise clínica do Eletrocardiograma (ECG) do paciente. Uma detecção prematura da AC pode auxiliar o tratamento, diminuindo consideravelmente o risco de vida do paciente. Entretanto, sua descoberta no surgimento dos primeiros indícios é uma tarefa difícil, pois envolvem diversas variáveis presentes em um ECG.

Para auxiliar especialistas nos diagnósticos de doenças cardiovasculares, uma recente e promissora linha de pesquisa vem sendo adotada, o emprego de métodos baseados em Aprendizado de Máquina na detecção de arritimia [1]. A partir de um conjunto prévio de exames ECGs devidamente classificados por médicos

especialistas, uma técnica de aprendizagem é aplicada gerando como resultado um modelo de classificação. Esse modelo então pode ser utilizado pelo médico na avaliação/classificação de ECGs de novos pacientes. Entretanto, o processo de criação de modelos de classificação eficazes é um desafio por duas questões principais: (1) cada ECG é composto por um conjunto muito grande de atributos; e (2) bases de dados relacionadas a avaliações de ECG são muito desbalanceadas, uma vez que o número de pacientes diagnosticados com AC são muito menores do que aqueles classificados como normais. Enquanto a primeira questão está relacionada ao custo computacional, a segunda limita o processo de aprendizagem das classes menores, sendo essas justamente os alvos dos modelos deste cenário.

Com o intuito de resolver as questões acima mencionadas, estratégias de pré-processamento de dados são empregadas, sendo as mais comuns as técnicas de seleção de atributos (*Feature Selection (FS)*) [2,3] e Sobreamostragem *Oversampling* [4,5]. FS consiste de técnicas que são capazes de mensurar a importância de cada atributo na construção do modelo de classificação para uma determinada base, retornando aqueles atributos mais relevantes, visando resolver a primeira questão previamente apresentada. O *Oversampling* consiste em replicar/combinar amostras relacionadas a classes menores, gerando novas amostras para compor o conjunto de dados com um desbalanceamento menor, aumentando a quantidade de informação relacionada às classes menores, estando relacionada à segunda questão. Com relação às técnicas de *Oversampling*, apesar de encontrarmos na literatura resultados importantes relacionados à eficácia, em coleções de dados cujo desbalanceamento é ainda mais acentuado, como no cenário de detecção de AC, a geração excessiva de amostras artificiais pode gerar distorções que comprometem a eficácia do modelo de classificação gerado. A partir dessa constatação, recentemente em [6], os autores apresentam uma técnica denominada *Classification using lOcal clusterinG* (COG), que consiste, resumidamente, na aplicação de alguma técnica de *clustering* nas classes majoritárias, desmembrando-as em outras X classes menores e, posteriormente, aplicando-se técnicas de *Oversampling* considerando a nova distribuição de classes geradas. A premissa dos autores é que menos amostras artificiais precisam ser geradas, diminuindo assim as distorções na geração do modelo de classificação.

Dessa forma, neste artigo é proposta e avaliada a combinação de técnicas de pré-processamento de dados visando a geração de modelos de classificação mais eficientes (menor custo computacional) e eficazes (melhor qualidade da classificação) para o problema de Detecção de AC. Mais especificamente, foram avaliados diferentes algoritmos de classificação, combinados com técnicas de FS, *Clustering* e *Oversampling*. Foi utilizada uma das coleções de dados relacionadas à AC mais referenciadas na literatura, provida pela UCI[7]. Em nossa avaliação experimental demonstramos que tratam-se de estratégias complementares, que quando combinadas, resultam em um modelo de classificação eficiente. Por exemplo, enquanto um modelo de classificação construído a partir um algoritmo *Random Forest* utilizando a coleção de dados sem nenhum pré-processamento resulta em uma Acurácia de 63%, o modelo gerado após a aplicação de uma técnica de FS resulta em uma Acurácia de 72%. Além disso, o modelo que combina *Clustering*

e *Oversampling* resulta em uma Acurácia de aproximadamente 82%. Por fim, o modelo que combina todas essas estratégias alcança uma Acurácia de 88,8%.

O restante do artigo está organizado do seguinte modo. Na Seção 2 são apresentados alguns trabalhos correlatos. A metodologia de trabalho é apresentada na Seção 3. Na Seção 4 os resultados da avaliação experimental são discutidos e, por fim, as conclusões e trabalhos futuros são apresentadas na Seção 5.

2 Trabalhos Relacionados

Nos últimos anos várias pesquisas relacionadas à classificação de AC têm sido realizadas, tendo como principal objetivo a detecção da arritmia utilizando modelos de classificação. Felipe et al.[8] desenvolveram modelos de classificação de AC utilizando oito conjuntos diferentes de variáveis relacionadas ao surgimento de AC em pessoas. Essas variáveis foram coletadas em tempo real de pacientes internados no Centro Hospitalar do Porto, tais como sinais vitais, resultados de laboratórios, entre outros. Tratam-se de dados bem controlados (não públicos) e relacionados somente a pacientes internados, resultado em uma coleção bastante balanceada, diferente da coleção considerada em nosso trabalho. Utilizando o algoritmo de classificação SVM, os autores conseguiram uma taxa de acerto de 95%.

Samad et al.[9] compararam três classificadores com base em sua acurácia para a detecção da arritmia na base de dados da UCI[7]. Os algoritmos de classificação kNN, Naive Bayes e Árvore de Decisão foram utilizados. O resultado mais relevante foi obtido pelo kNN, tendo alcançado 66,96% de acurácia. Este trabalho fornece uma explicação detalhada sobre a conversão de um ECG em valores numéricos para serem usados em tarefas de aprendizagem de máquina. Shivajirao et al.[10] criaram um sistema inteligente baseado em redes neurais artificiais para determinar a classificação da presença ou não da AC, também utilizando a base de dados da UCI. Os autores utilizaram o modelo *Multilayer Perceptron* com a técnica *Backpropagation*, alcançando uma acurácia de 86,67% o melhor resultado reportado na literatura para essa coleção. Conforme apresentaremos na Seção 04, combinando técnicas de *Feature Selection*, *Clustering* e *Oversampling*, alcançamos resultados superiores (i.e. 88,8% de acurácia).

Uma métrica de FS é utilizada para designar uma pontuação para cada atributo, a fim de avaliar a sua importância na tarefa de aprendizagem. Em [11], os autores comparam o desempenho de diversas métricas, tais como Ganho de Informação (*Information Gain*, ou IG), χ^2, *Odds Ratio* (OR) e Coeficiente de Correlação (*Correlation Coefficient*, ou CC). Em nosso trabalho avaliamos todas essas métricas. No que se refere a *clustering*, existem na literatura diversas propostas [12]. Tratam-se de técnicas simples e utilizáveis em diversos cenários, como o K-Means [13], até algumas mais elaboradas e específicas para determinados contextos, tais como clusterização por subespaços [14] e clusterização por particionamento [12]. Em nosso trabalho, consideramos apenas o K-Means, mas outras estratégias podem ser avaliadas no futuro, conforme detalhado da Seção 05.

Wu et al.[6] criou um método para tratar o problema de desbalanceamento entre classes que supera as técnicas de *Oversampling* para prever classes raras.

O método, intitulado *Classification using lOcal clustering* (**COG**), aplica uma técnica de *clustering* para dividir classes majoritárias em subclasses menores. Observou-se nos resultados uma melhoria significante na eficácia de algoritmos de classificação supervisionada. Também foi mostrado uma variação do COG, aplicando-se o método de *clustering* local junto a uma técnica de *Oversampling*. Essa variação recebeu o nome de *Classification using lOcal clustering with Over-Sampling* (**COG-OS**), sendo uma das técnicas adotas em nosso trabalho.

3 Metodologia de Avaliação

Nesta seção apresentamos a metodologia de avaliação utilizada. Conforme mencionado, nosso trabalho consiste na aplicação de técnicas de pré-processamento a fim de melhorar a qualidade dos algoritmos de classificação supervisionada aplicados ao problema de detecção de arritmia cardíaca. Na Figura 1 apresentamos a sequência seguida pela metodologia, que consiste, basicamente, em aplicar os algoritmos de classificação preprocessando os dados de diferentes formas.

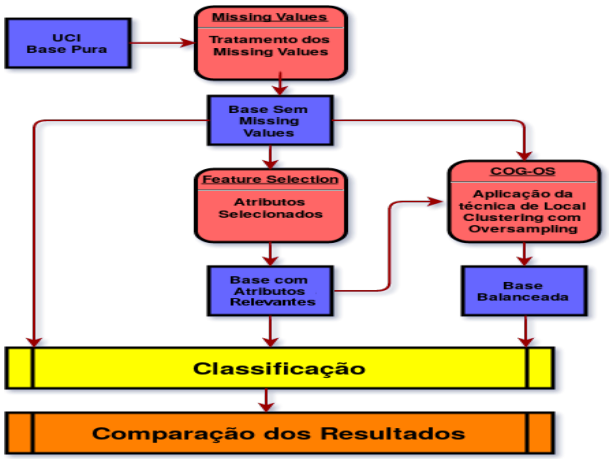

Figura 1. Sequencia Metodológica Seguida

1. **Tratamento de** *Missing Values***:** consiste na remoção/substituição de todos *missing values* dos atributos da base de dados que serão utilizadas nas demais etapas.
2. **Classificação Sem Pré-processamento:** nessa etapa é feita a classificação utilizando os algoritmos de classificação selecionados sem a utilização de nenhum pré-processamento, apenas o tratamento de *missing values*.
3. **Classificação com FS:** nessa etapa o objetivo é aplicar uma técnica de FS para remover os atributos que não agregam valor para a classificação, utilizando apenas um subconjunto de atributos relevantes. Após isso, geramos novos modelos de classificação e reavaliamos a qualidade alcançada em comparação com os resultados sem aplicação de nenhuma técnica de pré-processamento.

4. **Classificação com a Técnica COG-OS:** para essa parte da metodologia, aplica-se a técnica de COG sobre a base com os *missing values* tratados, porém considerando todos os atributos. Após a clusterização da classe majoritária (ECG's normais) e redistribuição da mesma em classes menores, aplica-se uma técnica de *Oversampling* nas classes minoritárias (ECG's arritmia), com objetivo de alcançar um balanceamento das classes na base de dados. Por fim, aplicam-se novamente os algoritmos de classificação para uma nova rodada de avaliação de resultados.
5. **Classificação com a Combinação da Técnica de FS com COG-OS:** nessa etapa será aplicado na base de dados as duas técnicas (FS e COG-OS) em conjunto. Utilizando-se a base de dados com *missing values* tratados, aplica-se uma técnica de FS para selecionar o subconjunto de atributos mais relevantes e, sobre a base de dados resultante, aplica-se o COG-OS. Com a base totalmente tratada, é feito a classificação com todos os algoritmos de classificação e comparam-se os resultados mais uma vez.

4 Avaliação Experimental

Nessa seção apresentamos os resultados experimentais referentes a classificação da base de dados referentes a detecção AC considerando os diferentes cenários definidos como etapas da metodologia.

4.1 Ambiente Experimental

4.1.1 Base de Dados A base de dados utilizada foi criada por Guvenir et al. [15] e disponibilizada pela UCI [1], sendo caracterizada por uma transformação de ECG's em atributos numéricos para a aplicação de técnicas de Mineração de Dados. Essa base possui *missing values* (valores faltantes) e amostras ambíguas que precisam ser tratadas para uma utilização mais eficiente dos algoritmos de classificação. A base de dados original possui 280 atributos. A base possui 16 classes, sendo que a classe 01 refere-se aos ECG's normais, a classe 13 refere-se aos ECG's que não possuem classificação e as demais referem-se aos ECG's com presença de algum tipo de arritmia. Três dessas classes foram desconsideradas por não possuirem nenhuma instância associada. A Figura 2 apresenta a distribuição das ocorrências entre as classes. Como podemos observar, trata-se de uma base de dados extremamente desbalanceada, de modo que algumas classes de arritmia possuem 2 instâncias, enquanto a classe de ECG's normais possui 245 instâncias.

4.1.2 Tratamento dos Missing Values Em uma análise prévia da base identificamos que um dos atributos (V14) possuía 390 instâncias com *missing values*, o qual foi removido de nossas análises. Para o restante dos atributos, o tratamento dos *missing values* foi efetuado utilizando o pacote *mice* disponibilizado junto com a Linguagem R. Esse pacote possui uma função para substituir valores incompletos por valores plausíveis sintéticos de acordo com todas as colunas da base, sem perder a consistência dos dados. Em todas as etapas de nossa avaliação experimental utilizamos a coleção de dados resultante desse tratamento.

[1] https://archive.ics.uci.edu/ml/datasets/Arrhythmia

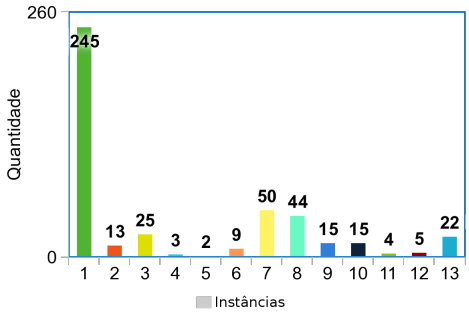

Figura 2. Distribuição de instâncias entre as classes da base de dados da UCI.

4.1.3 Algoritmos de Classificação Em nossas análises foram escolhidos algoritmos de classificação supervisionada considerados *estado-da-arte*, que tratam do problema por meio de diferentes abordagens, utilizando as implementações disponibilizadas pelo *Weka* [2] São eles:

- **Naive Bayes:** algoritmo probabilístico que calcula a probabilidade de uma determinada instância nova pertencer a cada uma das classes disponíveis de uma coleção. Trata-se de um dos métodos de máquina de aprendizagem mais utilizados que combina eficiência e simplicidade [16].
- **Random Forest:** trata-se de um algoritmo baseado na abordagem *bagging*, em que um conjunto de m árvores de decisão são treinadas considerando diferentes amostras do conjunto de treinamento. Posteriormente, cada uma dessas árvores é considerada na tomada de decisão final do algoritmo para se classificar uma nova instância [17].
- **Support Vector Machine (SVM**:) esse algoritmo mapeia o conjunto de treinamento como pontos em um espaço vetorial procurando definir os limites do espaço que separam cada uma das classes. Novas instâncias são mapeadas nesse espaço vetorial e atribuídos a classe de acordo com sua localização. Trata-se do algoritmo considerado como mais eficaz pela literatura [18].
- **k-Nearest Neighbor (kNN**:) trata-se de um algoritmo de classificação não-linear postergada (*lazy*) em que a classificação consiste em assinalar uma nova instância para a classe majoritária relacionada às k instâncias mais próximas em um espaço vetorial.[19].

Métricas Em nossas avaliações consideramos duas métricas: Acurácia e Macro F-Measure (**Macro F1**). A Acurácia é calculada como a porcentagem de instância corretamente classificadas. Já a Macro F1 é a média dos valores de *F-Measure* obtidos para cada classe possível na base de dados. Para definir a métrica *F-Measure* é necessário dois conceitos principais:

- Precision (Exatidão): quantidade itens classificados como positivos são realmente positivos.

[2] http://www.cs.waikato.ac.nz/ml/weka/

– Recall (Completude): quantidade de itens relevantes selecionados.

A F-Measure (**F1**) é a média harmônica entre *precision* e *recall*:

$$F1 = 2 * \frac{precision * recall}{precision + recall}. \quad (1)$$

Foi utilizada a estratégia de *10-fold Cross Validation*, que consiste em dividir o conjunto total de dados em 10 subconjuntos mutuamente exclusivos do mesmo tamanho e, a partir disso, um subconjunto é utilizado para teste e os 9 subconjuntos restantes são utilizados para o treino do modelo. Esse processo é repetido 10 vezes, alternando o subconjunto de teste. Ao final, os resultados reportados referem-se à média das acurácias e MacroF1 obtidas nas 10 repetições.

4.2 Análise dos Resultados

O primeiro resultado foi obtido por meio da avaliação dos algoritmos de classificação sem o uso das técnicas de pré-processamento. Na Tabela 1 é mostrado os valores de acurácia e MacroF1 obtidos por cada algoritmo de classificação avaliado.

Tabela 1. Resultados Obtidos na Classificação com a Base Original Desbalanceada

Algoritmo	Acurácia	MacroF1
Naive Bayes	$62,0\%$	$61,0\%$
Random Forest	$69,9\%$	$62,3\%$
k-NN	$58,1\%$	$45,6\%$
SVM Linear	$54,2\%$	$38,1\%$

Como podemos observar, o algoritmo *Random Forest* foi que o obteve o melhor valor de MacroF1 e Acurácia na base desbalanceada, sendo que, o *Naive Bayes* obteve um valor aproximado. O valor obtido é baixo, pois na base desbalanceada a maioria das classes de arritmia não é classificada corretamente. Isso se deve ao fato de que os modelos criados foram treinados em cima de uma base de dados desbalanceada, enviesada para classes normais (mais frequente).

O segundo conjunto de resultados diz respeito a combinação dos algoritmos de classificação com técnicas de FS. Foram testados diversos algoritmos de FS mencionados na Seção 02 e disponibilizados pelo Weka. Cada algoritmo de classificação foi testado considerando-se os atributos selecionados por diferentes algoritmos de FS, sendo o *CfsSubsetEval* [20] aquele que obteve melhores resultados. O *CfsSubsetEval* calcula para cada subconjunto de atributos a correlação de cada um deles com as classes da base de dados. Nesse caso, é desejável o subconjunto que possua uma alta correlação com uma classe em que cada atributo do subconjunto possua uma baixa correlação entre si. Assim, ele vai adicionando/removendo atributos até que se atinja um subconjunto que possua somente os atributos mais relevantes para prever a classe desejada.

Assim, o algoritmo de FS conseguiu diminuir a quantidade de atributos, selecionando somente **23 atributos** dos 280 contidos na base de dados original.

A seguir, foram aplicados os algoritmos de classificação usados anteriormente na base de dados com somente 23 atributos. A Tabela 2 apresenta os resultados obtidos nesta etapa.

Tabela 2. Resultados Obtidos na Classificação com Aplicação da Técnica de FS

Algoritmo	Acurácia	MacroF1
Naive Bayes	68, 4%	66, 2%
Random Forest	75, 7%	72, 7%
k-NN	63, 9%	55, 2%
SVM Linear	54, 2%	38, 1%

Analisando a Tabela 2, podemos perceber que quase todos os classificadores, com exceção do SVM, obtiveram melhoras na qualidade de classificação considerando apenas os 23 atributos mais relevantes. É importante ressaltar que, além da melhora dos modelos de classificação, o uso de técnicas de FS também pode melhorar a eficiência no processo de criação dos modelos de classificação.

A terceira etapa consistiu, primeiramente, na utilização da técnica de COG como pré-processamento para a classificação. Na base de dados de arritmia, somente a classe normal possui um grande número de objetos, logo, é aplicado o *clustering* para que ela seja transformada em subclasses de tamanhos menores. O algoritmo de *clustering* escolhido foi o **K-Means**, que consiste em particionar os objetos em K grupos onde cada objeto pertença a um grupo. O algoritmo cria K centros no espaço de objetos e segue mudando a localização de seus centros até que a quantidade de objetos em cada centro de uma iteração para outra não se altere. Para determinar o número K foi analisado o valor referente ao *Within-Cluster Sum of Squares* (WCSS), que é a soma feita dentro de cada *cluster* entre seus objetos e seu centro elevado ao quadrado. Foi observado o valor de WCSS de K variando de 0 a 10, sendo que, 4 foi o melhor número de *clusters* encontrado. O resultado dessa estratégia na distribuição das classes é representada na Figura 3.

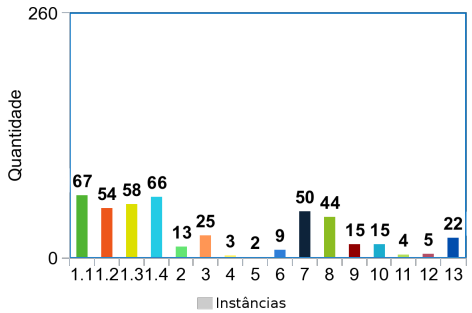

Figura 3. Base de Dados Resultante da Aplicação do COG.

A base resultante se tornou quase balanceada, sendo necessária a aplicação de alguma técnica de *Oversampling* para se obter um balanceamento mais relevante. Nesse caso, optamos pelo algoritmo **SMOTE** [21], disponível na Linguagem R.

O SMOTE consiste na criação de instâncias sintéticas das classes raras. Para cada classe que se deseja criar objetos para tornar a base de dados balanceada, o SMOTE usa k objetos vizinhos para criar uma instância sintética que seja próxima desses k objetos. A base resultante é apresentada na Figura 4.

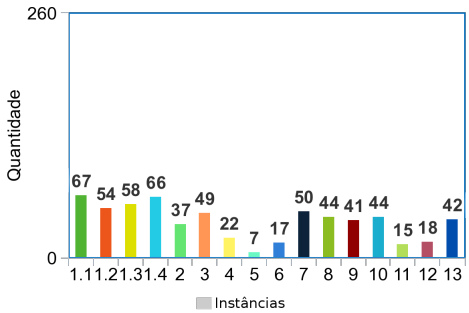

Figura 4. Base de Dados Resultante da Aplicação do COG-OS.

Com a base resultante da aplicação do COG–OS foi realizada a classificação para comparação com os resultados obtidos anteriormente. A Tabela 3 apresenta o resultado, utilizando a base de dados com 280 atributos.

Tabela 3. Resultados Obtidos na Classificação Depois da Aplicação do COG-OS

Algoritmo	Acurácia	Macro F1
Naive Bayes	70, 1%	70, 0%
Random Forest	82, 6%	81, 9%
k-NN	65, 6%	62, 5%
SVM Linear	30, 4%	32, 2%

A quarta e última etapa consistiu em combinar as técnicas de *clustering* local, *oversampling* e *FS* antes da geração do modelo de classificação. Assim, foi aplicado o COG-OS na base de dados com somente 23 atributos e foi obtida a base balanceada mostrada na Figura 5.

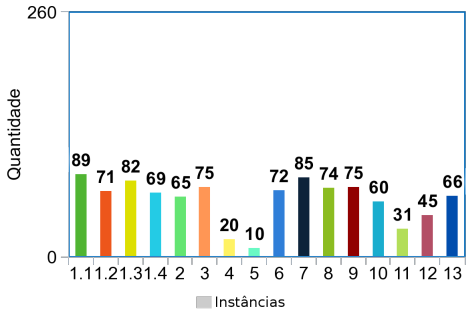

Figura 5. Base de Dados Resultante da Aplicação do COG-OS com 23 Atributos.

A Tabela 4 apresenta os resultados obtidos na aplicação dos algoritmos de classificação na base apresentada na Figura 5. Como podemos observar, a combinação das técnicas foi muito eficaz, aumentando ainda mais a qualidade das classificações obtidas.

Tabela 4. Resultados Obtidos na Classificação Depois da Aplicação do COG-OS na Base com 23 Atributos

Algoritmo	Acurácia	Macro F1
Naive Bayes	71, 9%	71, 3%
Random Forest	88, 9%	88, 8%
k-NN	71, 992%	70, 6%
SVM Linear	29, 424%	32, 2%

4.3 Discussão

As técnicas de FS e o método COG-OS mostraram uma excelente estratégia em aprimorar a eficácia dos classificadores escolhidos, com exceção do SVM. O algoritmo que obteve os melhores resultados foi o *Random Forest*, sendo alcançado uma MacroF1 de quase 90%, tornando o melhor resultado já reportado na literatura para a coleção de detecção de AC. A Tabela 5 mostra como foi possível aumentar, gradualmente, o valor da acurácia e, principalmente, do valor da MacroF1 do algoritmo *Random Forest*. Trata-se de um importante avanço científico mostrando que a combinação de diferentes estratégias de mineração de dados pode auxiliar significativamente na construção de modelos de classificação que auxiliem médicos especialistas na detecção de arritmia cardíaca.

Tabela 5. Resultado Obtido Pelo Random Forest em Cada Etapa

Técnicas de Preprocessamento Utilizadas	MacroF1
Nenhuma	62, 3%
FS	72, 7%
COG-OS	81, 9%
FS + COG-OS	**88,8%**

5 Conclusões e Trabalhos Futuros

Neste artigo foi demonstrado como que o desbalanceamento entre classes de uma base de dados relacionadas a detecção de AC influencia negativamente no processo de criação de modelos de classificação supervisionadas existentes. A grande maioria dos diagnósticos de arritmia é classificada como normal e os casos de incidência da doença são raros. Dessa forma, foram avaliadas várias estratégias de pré-processamento de dados combinadas com técnicas de classificação automática no intuito de criar modelos de classificação mais eficazes para auxiliar especialistas na detecção da doença.

Mais especificamente, os resultados deste artigo demonstraram que modelos de classificação construídos a partir de um subconjunto de atributos mais relevantes, selecionados por meio de uma técnica de FS, tendem a melhorar significativamente a qualidade dos modelos gerados. De maneira análoga e complementar, foi demonstrado que uma estratégia de *Oversampling*, combinada com uma abordagem de *clustering* (COG-OS), também resulta em modelos eficazes. Além disso, combinando ambas as estratégias foi alcançado um modelo de classificação ainda melhor, superando o melhor resultado reportado na literatura. Mais especificamente, utilizando o algoritmo de classificação **Random Forest**, considerando apenas os 23 atributos mais relevantes e aplicando a estratégia de COG-OS, foi obtida uma MacroF1 de 88,8%, superando os 86% alcançados em [10] para a mesma base de dados da UCI utilizada.

Como trabalho futuro, o objetivo é aumentar ainda mais a predição de arritmias utilizando outros algoritmos de classificação, *clustering* e *oversampling* nas etapas propostas no artigo. Além disso, uma analise detalhada dos 23 atributos selecionados pode facilitar a detecção de arritmias em um ECG, descobrindo quais as relações desses atributos no seu respectivo ECG.

Agradecimentos

Esse trabalho foi parcialmente financiado por CNPq, CAPES, FINEP, Fapemig, e INWEB.

Referências

1. Özçift, A.: Random forests ensemble classifier trained with data resampling strategy to improve cardiac arrhythmia diagnosis. Computers in biology and medicine **41**(5) (2011) 265–271
2. Alelyani, S., Tang, J., Liu, H.: Feature selection for clustering: A review. Data Clustering: Algorithms and Applications **29** (2013) 110–121
3. Liu, H., Motoda, H.: Feature selection for knowledge discovery and data mining. Volume 454. Springer Science & Business Media (2012)
4. Barua, S., Islam, M.M., Yao, X., Murase, K.: Mwmote–majority weighted minority oversampling technique for imbalanced data set learning. IEEE Transactions on Knowledge and Data Engineering **26**(2) (2014) 405–425
5. Douzas, G., Bacao, F.: Self-organizing map oversampling (somo) for imbalanced data set learning. Expert Systems with Applications **82** (2017) 40–52
6. Wu, J., Xiong, H., Wu, P., Chen, J.: Local decomposition for rare class analysis. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM (2007) 814–823
7. Lichman, M.: UCI machine learning repository (2013)
8. Portela, F., Santos, M.F., Silva, Á., Rua, F., Abelha, A., Machado, J.: Preventing patient cardiac arrhythmias by using data mining techniques. In: Biomedical Engineering and Sciences (IECBES), 2014 IEEE Conference on, IEEE (2014) 165–170
9. Samad, S., Khan, S.A., Haq, A., Riaz, A.: Classification of arrhythmia. International Journal of Electrical Energy **2**(1) (2014) 57–61

10. Jadhav, S.M., Nalbalwar, S., Ghatol, A.: Artificial neural network based cardiac arrhythmia classification using ecg signal data. In: Electronics and Information Engineering (ICEIE), 2010 International Conference On. Volume 1., IEEE (2010) V1–228
11. Zheng, Z., Wu, X., Srihari, R.: Feature selection for text categorization on imbalanced data. **6** (2004) 80–89
12. Berkhin, P.: A survey of clustering data mining techniques. Grouping Multidimensional Data (2006) 25–71
13. Farivar, R., Rebolledo, D., Chan, E., Campbell, R.H.: A parallel implementation of K-means clustering on GPUs. In: Proc. of PDPTA'08, USA (July 2008) 340–345
14. Agrawal, R., Gehrke, J., Gunopulos, D., Raghavan, P.: Automatic subspace clustering of high dimensional data for data mining applications. In: Proc. of SIGMOD '98, New York, USA, ACM (1998) 94–105
15. Guvenir, H.A., Acar, B., Demiroz, G., Cekin, A.: A supervised machine learning algorithm for arrhythmia analysis. In: Computers in Cardiology 1997, IEEE (1997) 433–436
16. Viegas, F., Gonçalves, M.A., Martins, W., Rocha, L.: Parallel lazy semi-naive bayes strategies for effective and efficient document classification. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. CIKM '15, New York, NY, USA, ACM (2015) 1071–1080
17. Breiman, L.: Random forests. Machine Learning **45**(1) (Oct 2001) 5–32
18. Joachims, T.: Advances in kernel methods. MIT Press, Cambridge, MA, USA (1999) 169–184
19. Rocha, L., Ramos, G., Chaves, R., Sachetto, R., Madeira, D., Viegas, F., Andrade, G., Daniel, S., Gonçalves, M., Ferreira, R.: G-knn: An efficient document classification algorithm for sparse datasets on gpus using knn. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing. SAC '15, New York, NY, USA, ACM (2015) 1335–1338
20. Hall, M.A.: Correlation-based Feature Subset Selection for Machine Learning. PhD thesis, University of Waikato, Hamilton, New Zealand (1998)
21. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: Synthetic minority over-sampling technique. J. Artif. Int. Res. **16**(1) (June 2002) 321–357