Análise de técnicas de aprendizagem em problemas de classificação

Gabriel dos Santos Sereno[1]

Abstract

Este artigo tem como objetivo analisar a aplicação de algoritmos classificadores em bases acadêmicas para construir um comparativo com a taxa de acurácia e demais dados. Nisso, foi utilizado os classificadores/ensembles do ScikitLearn. São eles: BaggingClassifier, AdaBoostClassifier, RandomForestClassifier e o Heterogeneous Pooling com os classificadores GaussianNB, DecisionTreeClassifier e o KNeighborsClassifier. Como os classificadores, as base de dados também foram utilizadas com o pacote ScikitLearn e foram selecionadas as bases: wine, breast cancer e digits.

Keywords: Classificadores, Base de dados, Algoritmos, *ScikitLearn*, *Ensembles*

1. Introdução

Saber utilizar o classificador correto para um problema é essencial para evitar uso demasiado de tempo e baixa acurácia. Dessa forma, comparativos e testes com os classificadores de maior preferência torna-se comum para resolver essa questão.

Este artigo, tem como objetivo comparar os classificadores *ensemble* para ser possível a analise de desempenho. Nisso, foi utilizados comparativos de acurácia, testes estatísticos, gráficos *Box plot* para ilustrar de forma clara as diferenças das técnicas. Os classificadores testados foram: *BaggingClassifier*, *AdaBoostClassifier*, *RandomForestClassifier* e o *Heterogeneous Pooling* com os

[1]Aluno do curso de mestrado com enfase em Inteligência Artificial na Universidade Federal do Espírito Santo

classificadores *GaussianNB*, *DecisionTreeClassifier* e o *KNeighborsClassifier* advindos do pacote *ScikitLearn*.

Todas as bases foram padronizadas com o *StandardScaler*, além de passarem pelo *Cross-validation* e o *Grid Search* para a busca de hiperparâmetros. Após a obtenção de dados, o comparativo foi feito com testes estatísticos como *T-Student* e o *Wilcoxon*, além de tabelas com as informações da mediana, desvio padrão, acurácia e os suportes superiores e inferiores de cada classificador testado.

2. Classificadores

O foco do artigo é voltado inteiramente na análise do poder de ensembles, bem como na assertividade e técnica em comparação com técnicas simples. Além disso, para engrandecer o conhecimento transmitido por este artigo, foi implementado a técnica de *Heterogeneous Pooling* com os classificadores *GaussianNB*, *DecisionTreeClassifier* e o *KNeighborsClassifier* e verificar seu desempenho diante dos ensembles padrões do pacote *ScikitLearn*.

2.1. Heterogeneous Pooling

O *Heterogenous Pooling* é estabelecido por um conjunto de classificadores, como o nome sugere, para potencializar a assertividade na escolha da classificação. De fato, é possível a partir da votação da maioria dos classificadores em uma determinada classe. Apesar do classificador ser mais difícil de ser implementado em comparação aos fornecidos pelo *ScikitLearn*, o ensemble se destaca pela diversificação dos resultados a partir da gama de classificadores, reduzindo os vícios na classificação e obtendo maior chance de acerto em uma classe que geralmente um único classificador erraria.

Hiperparâmetro. O único hiperparâmetro fornecido pela classe, foi o *n_Samples*, que determina quantos classificadores irão atuar na base fornecidos pelo *Grid Search*.

fit():. Primeiramente para a construção do *Heterogenous Pooling* e da votação majoritária, é necessário verificar a frequência das classes para guardar no momento da decisão. Além disso, outra técnica utilizada no método *fit*, foi o *resample* que permite embaralhar e gerar novos dados a partir da base original, criando maios exemplos para treinar os classificadores. Com a quantidade determinada de classificadores e das bases, inicia-se o treinamento de todos os classificadores com suas respectivas bases para serem preditas no próximo método.

predict():. Com os classificadores treinados, o predict recebe a base de treinamento e verifica qual foi a classe mais votada pelos classificadores, caso aconteça empate, é escolhido a classe que mais apareceu na base original.

3. Comparativo entre os resultados

Para realizar o estudo dos classificadores, foi utilizado a técnica de validação cruzada estratificada (*Cross-validation*) de 10 *folds*, com 3 repetições internas.

Junto com a validação cruzada, foi utilizado a técnica de *Grid Search* para buscar o melhor hiperparâmetro e em consequência o melhor resultado. Com isso, foi divido os classificadores em dois tipos de hiperparâmetros como mostra a tabela a seguir:

Classificador	**Hiperparâmetro**
BaggingClassifier	n_estimators = [10, 25, 50, 100]
AdaBoostClassifier	n_estimators = [10, 25, 50, 100]
RandomForestClassifier	n_estimators = [10, 25, 50, 100]
Heterogeneous Pooling	n_Samples = [1, 3, 5, 7]

Table 1: Distribuição dos hiperparâmetros dos classificadores

3.1. Digits

A base *Digits* são uma das maiores bases em comparação com as demais, contendo mais de 1700 linhas com 64 colunas, além de ter 10 classes que torna o

trabalho do classificador um pouco mais árduo. Os resultados dos classificadores utilizados estão na tabela a seguir:

Classificador	**Acurácia**	**Desvio padrão**	**Lim. inf.**	**Lim. sup.**
Bagging	0,9517	0,0134	0,9469	0,9565
AdaBoost	0,2698	0,0222	0,2619	0,2778
RandomForest	0,9756	0,0103	0,9719	0,9793
Het. Pooling	0,9560	0,0130	0,9513	0,9607

Table 2: Resultados dos classificadores na base *Digits*

O classificador *Bagging* utiliza a técnica de ensemble, separando os dados aleatoriamente em pequenas bases para que suas árvores de decisão defina a classe correta a partir de cada modelo. Apesar da dificuldade de classificar a base *Digits*, o classificador teve uma ótima taxa de acerto com a acurácia de 95%, além do desvio baixo de 1%. Com isso, o classificador *Bagging* mostrou-se eficaz com grandes quantidades de dados e muitas classes.

Apesar da grande utilização da técnica de *Boosting* e de sua alta confiabilidade e precisão, essa técnica ficou a desejar, tendo resultados totalmente adversos do ótimo desempenho do método de *Bagging*, tendo apenas 26% de acurácia e consequentemente o limite inferior e superior ficaram com 26% e 27% respectivamente, com 2% de desvio padrão. Esses resultados mostram como é importante a comparação ao utilizar métodos de predição e de classificação, pois até métodos renomados podem não ter uma alta acurácia ou utilizando muito tempo para classificar corretamente, enquanto classificadores simples como o *KNeighborsClassifier* podem apresentar resultados melhores com pouco tempo.

Já o *RandomForest* e o *Heterogeneous Pooling* tiveram resultados semelhantes ao *Bagging*, mas com destaque ao *RandomForest* que costumeiramente consegue resultados melhores em diversas situações com 97% de acurácia e com 1% de desvio padrão.

Esses dados ficam mais expressivos no Boxplot a seguir:

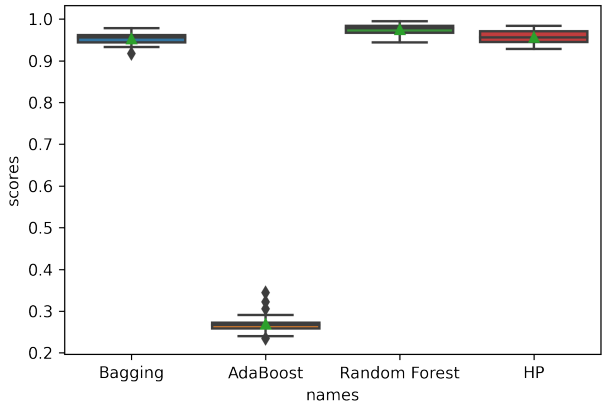

Figure 1: Gráfico *Boxplot* a partir dos resultados dos classificadores na base *Digits*.

A discrepância entre o *AdaBoost* e os demais classificadores é grande, tornando o gráfico desequilibrado por sua instabilidade. Para melhor comparação, nesse gráfico foi retirado o *AdaBoost*:

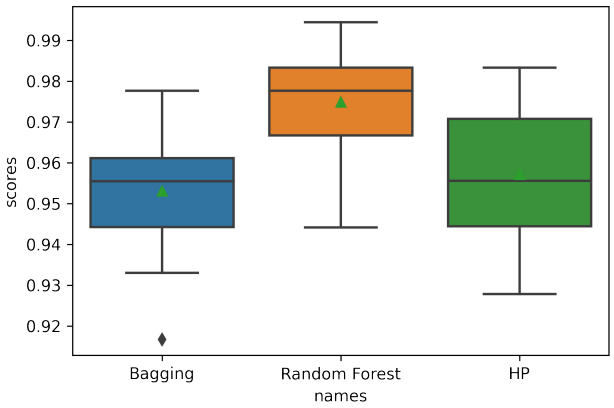

Figure 2: Gráfico *Boxplot* sem o *AdaBoost* na base *Digits*.

Com isso, em ambos os gráficos mostram que os três classificadores tiveram relativamente o mesmo desempenho. Para comprovar isso, foi feito o teste de T pareado e de *Wilcoxon* para mostrar se realmente esses classificadores não tem

uma grande disparidade. A Tabela 2 mostra os resultados dos testes:

Bagging	0,0000	0,0000	0,4255
0,0000	**AdaBoost**	0,0000	0,0000
0,0000	0,0000	**Random Forest**	0,0001
0,5214	0,0000	0,0000	**Het. Pooling**

Table 3: Gráfico dos testes na tabela pareada da base *Digits*.

Na Tabela 3, a representação dos classificadores que tiveram a hipótese nula rejeitada são os classificadores que tiveram o resultado abaixo de 5%, em outras palavras, onde há uma diferença significativa e assim, os demais classificadores aceitam a hipótese. Com essa tabela é possível verificar que há uma igualdade expressiva entre *Heterogeneous Pooling* e o *Bagging* que obtiveram um dos melhores resultados. Portanto, é possível afirmar que o *AdaBoost* não é recomendado para bases que contem muitas classes, já que tem um desempenho muito inferior aos demais. Além disso, o uso do *Random Forest* é altamente recomendado pela sua consistência e grande assertividade em diversas bases. E por fim, o *Heterogeneous Pooling* e o *Bagging* são semelhantes, não sendo possível definir qual é o melhor entre ambos.

3.2. Wine

A base *Wine* é menor comparado ao *Digits*, contendo apenas 3 classes e cerca de 180 linhas com 13 colunas. Dessa forma, os classificadores tendem a ter resultados mais precisos pelo menor número de classe, mas menos generalizado pela quantidade de linhas. Os resultados dos classificadores utilizados estão na tabela a seguir:

Classificador	**Acurácia**	**Desvio padrão**	**Lim. inf.**	**Lim. sup.**
Bagging	0,9643	0,0566	0,9444	0,9846
AdaBoost	0,9123	0,0715	0,8867	0,9378
RandomForest	0,9831	0,0385	0,9693	0,9969
Het. Pooling	0,9660	0,0541	0,9466	0,9853

Table 4: Resultados dos classificadores na base *Wine*

Novamente o *RandomForestClassifier* teve um maior desempenho pela adaptabilidade com 98% de acurácia, 3% de desvio padrão e os limite inferior e superior foram de 96% e 99% respectivamente, mostrando o poder de classificação. Vale ressaltar que é importante verificar se o classificador irá se comportar bem com novos dados e se ajustou demais a base. Dessa vez o *AdaBoostClassifier* melhorou o seu desempenho, mas novamente teve o pior resultado, distanciando dos demais, com alto desvio padrão de 7% e apenas 91% de acurácia.

O gráfico *Boxplot* mostra a diferença do *Heterogeneous Pooling* diante dos demais classificadores e a total dominância do *RandomForestClassifier*:

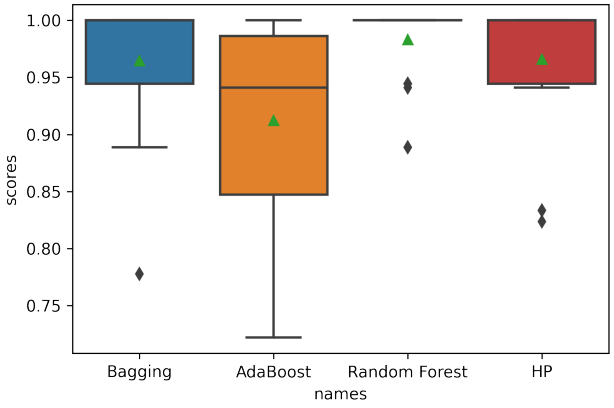

Figure 3: Gráfico *Boxplot* a partir dos resultados dos classificadores na base *Wine*.

Bagging	0,0035	0,1307	0,5892
0,0028	**AdaBoost**	0,0005	0,0029
0,1360	0,0002	**Random Forest**	0,0803
0,7556	0,0030	0,0683	**Het. Pooling**

Table 5: Gráfico dos testes na tabela pareada da base *Wine*.

De acordo com a Tabela 5, novamente os classificadores *Heterogeneous Pooling* e *Bagging* tiveram altíssima similaridade, além de serem um dos melhores em acurácia. Além disso, outros classificadores obtiveram pequenas similaridades que aceitaram a hipótese nula.

Para concluir a análise na base *Wine*, o classificador construído *Heterogeneous Pooling* mostra-se eficaz em bases com maior número de dados e com alta aceitabilidade em decorrência aos seus classificadores internos que possibilitam maior precisão no momento da classificação. Já o *AdaBoost*, é possível perceber que seu desempenho é mais eficaz quando há poucas classes para serem definidas, pois os seus pesos são ajustados de forma mais assertiva.

3.3. Breast cancer

A base *Breast cancer* contêm 2 classes apenas, significando se o paciente possui ou não o câncer. Além disso, é uma base mediana, contendo 569 linhas com 30 colunas. Com essas características, os classificadores tendem a ter um excelente desempenho, pois contem um bom número de linhas e colunas e o trabalho mais fácil de classificar apenas duas classes. Dessa forma, os classificadores podem ter resultados semelhantes. Os resultados dos classificadores utilizados estão na tabela a seguir:

Classificador	Acurácia	Desvio padrão	Lim inf.	Lim. sup.
Bagging	0,9548	0,0316	0,9435	0,9662
AdaBoost	0,9677	0,0235	0,9593	0,9762
RandomForest	0,9583	0,0233	0,9500	0,9667
Het. Pooling	0,9548	0,02388	0,9463	0,9634

Table 6: Resultados dos classificadores na base *Breast cancer*

De fato, os resultados foram bem semelhantes, mas com duas pequenas diferenças das demais tabelas que são: *AdaBoost* com o melhor resultado e o *RandomForest* que não obteve uma diferença significativa entre os demais classificadores. Essa características ficam expressivas ao ver o gráfico *Bloxpot*:

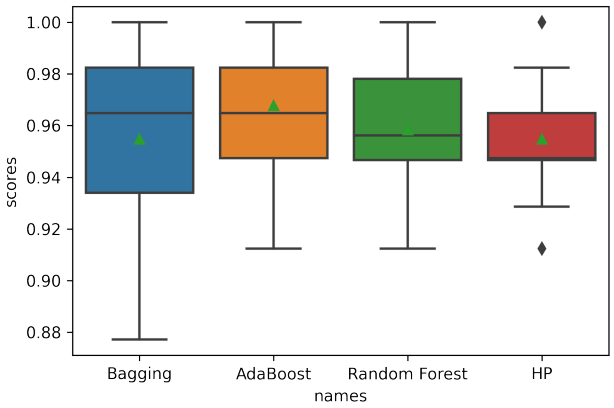

Figure 4: Gráfico *Boxplot* a partir dos resultados dos classificadores na base *Breast cancer*.

Infelizmente o *Heterogeneous Pooling* construído não obteve um dos melhores desempenhos, mas obteve uma boa consistência em comparação aos classificadores *Bagging* e o *RandomForest*, que obtiveram 95% de acurácia e 94% de limite inferior e 96% de limite superior.

O *AdaBoost* obteve o melhor desempenho nessa base com 96% de acurácia, 2% de desvio padrão e para o limite inferior e superior foram 95% e 97% respectivamente.

Bagging	0,0027	0,5014	0,9809
0,0017	**AdaBoost**	0,0178	0,0134
0,3847	0,0222	**Random Forest**	0,6904
0,9072	0,0121	0,5743	**Het. Pooling**

Table 7: Gráfico dos testes na tabela pareada da base *Breast cancer*.

Com os testes de *Wilcoxon* e de *T-test* torna-se possível evidenciar que o classificador *AdaBoost*, como previsto nas bases anteriores, teria maior assertividade em bases com poucas classes. Dessa vez, o *AdaBoost* foi o melhor classificador e rejeitando a hipótese nula. Vale ressaltar, que os classificadores *Heterogeneous Pooling*, *Bagging* e o *Random Forest*, apresentaram altas taxas nos testes, aceitando a hipótese nula em decorrência de sua igualdade.

4. Conclusões

Trabalhar com comparações é essencial para achar o melhor resultado e o classificador que encaixa melhor a um determinado problema. Com os testes feitos desse artigo, foi possível ver a diferença do *AdaBoost* em diferentes bases e a consistência do *Random Forest*.

Ademais, o classificador construído *Heterogeneous Pooling* demonstrou bem eficaz com todas as bases, tendo resultados semelhantes aos demais classificadores que tiveram os melhores resultados. Esse desempenho, foi possível graças aos classificadores *GaussianNB*, *DecisionTreeClassifier* e o *KNeighborsClassifier* que são altamente precisos e com grande reputação entre os desenvolvedores e estatísticos.

Com os testes de *Wilcoxon* e *T-test* foi possível analisar a diferença entre os classificadores, logo determinando qual classificador teve o melhor desempenho em conjunto com as tabelas e os gráficos de apoio apresentados. Geralmente, os melhores classificadores tiveram uma diferença significativa aos demais classificadores, determinando a sua superioridade.

Esse trabalho contribuiu com o aprofundamento das técnicas de classificação

e de testes, tornando o conhecimento aprendido na sala de aula em prática. Dessa forma, o conhecimento adquirido é consolidado para situações reais. Uma das possíveis contribuições futuras é a utilização de meta-heurística junto com os classificadores para verificar se há aumento de desempenho significativo, bem como o aumento de número de bases, utilizando até mesmo algumas bases reais disponíveis de forma gratuita na internet.

References

- Slides, artigos e notas transmitidas em aula das disciplinas tutoradas pelo professor.