Projeto de Ciência de Dados, Grupo 81

MATHEUS DE ALMEIDA ORSI E SILVA, 92238
JOÃO MARCELO FREZZATO OLIVEIRA, 95355
RENAN YUDI HAMADA NUNES, 95357

Este documento apresenta as abordagens do grupo para estudar os dois conjuntos de dados fornecidos pelos docentes, cobrindo o conteúdo da disciplina. O primeiro *dataset* trata da classificação de indivíduos para a Doença de Parkinson e o segundo da classificação de tipos de cobertura florestal baseado em dados cartográficos. São aplicados métodos de exploração e análise de dados, pré-processamento, aprendizagem supervisionada e não supervisionada. Os resultados e performances de diferentes abordagens são comparados e devidamente analisados.

1 INTRODUÇÃO

Nesse projeto, o grupo tem como objetivo estudar dois conjuntos de dados com características muito diferentes entre si.

O primeiro deles é um *dataset* feito em 2018 sobre a Doença de Parkinson. Estão presentes dados de 188 pacientes com a doença e de 64 pessoas saudáveis, bem como suas respectivas classificações para a presença ou não da doença. Tais dados são procedentes de diversos algoritmos de processamento de fala.

O outro *dataset*, de 1998, trata da classificação do tipo de cobertura florestal da *Roosevelt National Forest*, nos EUA. As observações representam áreas de 30 x 30 metros e as váriaveis independentes são informações geográficas e geológicas, como a elevação do terreno, distância até estradas, tipo do solo, entre outras.

Nesse projeto, os conjuntos de dados serão analisados em paralelo, desde a análise estatística até as conclusões tomadas a partir dos resultados obtidos com a aplicação de modelos de aprendizagem.

2 EXPLORAÇÃO DE DADOS

Ao deparar-se com um conjunto de dados, convém compreendê-lo melhor em aspectos gerais antes de aprofundar-se nas técnicas de pré-processamento e uso de métodos de machine learning. Isso envolve aspectos qualitativos, quantitativos e análise estatística.

O primeiro *dataset* (*Parkinson Desease Dataset* - PDD) tem 756 observações, das quais cada 3 correspondem a 3 amostras diferentes para um só indivíduo. As variáveis independentes são 755. Os dados foram colhidos de 2 grupos, um (de estudo) com 188 pacientes com a Doença de Parkinson e o outro (de controle), com 64 pessoas saudáveis.

As variáveis do PDD são todas numéricas, exceto o gênero e o diagnóstico, que são binárias, e não há valores ausentes. Em relação às suas distribuições, muitos valores encontram-se fora do intervalo interquartil (figura 1). Entretanto, devido ao pequeno tamanho do *dataset*, não se deve descartar esses objetos. Além disso, é possível que esses dados mais extremos sejam indicadores da presença ou ausência da doença, ponto que será explorado posteriormente.

Através da função *SelectKBest* da biblioteca *sklearn*, obtém-se o valor-p para cada variável, possibilitando assim verificar como a distribuição das classes (com e sem Parkinson) varia para essas

Authors' addresses: Matheus de Almeida Orsi e Silva, 92238; João Marcelo Frezzato Oliveira, 95355; Renan Yudi Hamada Nunes, 95357.

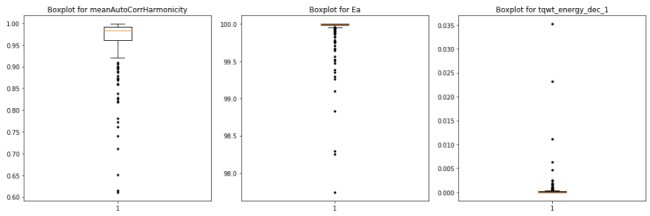


Fig. 1. Diagrama de extremos e quartis para algumas variáveis do PDD

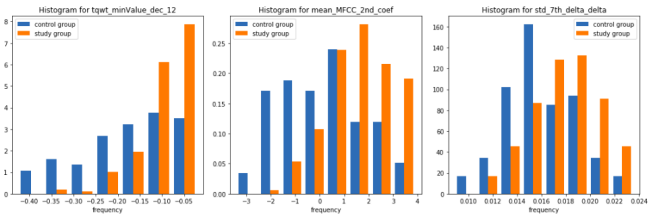


Fig. 2. Histograma diferenciando o grupo de estudo e de controle para algumas variáveis do PDD

características (figura 2). No total, 388 variáveis apresentam valor-p menor do que 0.05, o que indica que devem apresentar alguma informação a respeito do atributo alvo. Nos testes feitos e apresentados neste relatório, é possível verificar a influência do número das variáveis nas métricas.

Outro fator explorado foi a correlação entre as variáveis. O grupo analisou as correlações de forma gráfica, através de um mapa de calor (figura 3). Verificou-se que diversos atributos são altamente correlacionados, principalmente os que descrevem propriedades físicas parecidas, mas com escala ou algoritmos diferentes. Tais *features* muitas vezes estão em mesmos subgrupos, os quais são definidos no *dataset*. Nesses casos, é ineficiente manter todas elas, então podemos escolher apenas uma para representar o conjunto.

O segundo *dataset (Covertype - CT)* tem 581,012 observações e 55 variáveis. Dentre as 55, 40 são valores binários referentes ao "Tipo de solo", sendo que apenas um desses valores é igual a 1, ou seja, cada observação apresenta um único tipo de solo. O mesmo ocorre com os 4 atributos referentes às "Áreas selvagens". As restantes são númericas.

Ao contrário do primeiro *dataset*, o segundo tem como classe alvo 7 valores possíveis: 1 - Spruce/Fir; 2 - Lodgepole Pine; 3 - Ponderosa Pine; 4 - Cottonwood/Willow; 5 - Aspen; 6 - Douglas-fir; 7 - Krummholz. A frequência de cada observação está na figura 4.

O CT também não apresenta valores ausentes e possui um diagrama de extremos e quartis com muitos pontos fora da faixa interquartil (figura 5), que, novamente, serão mantidos.

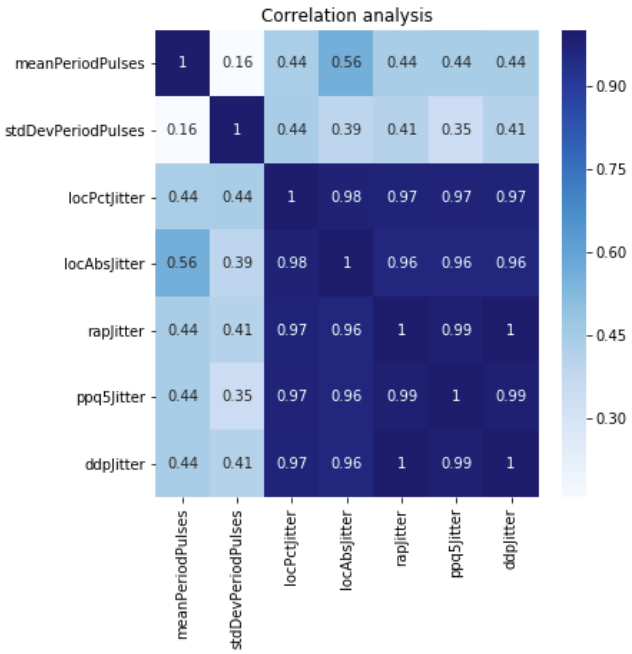


Fig. 3. Segmento do mapa de calor das variáveis do PDD

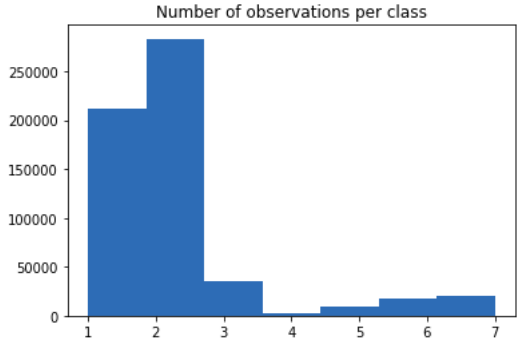


Fig. 4. Histrograma da frequência de cada classe do CT

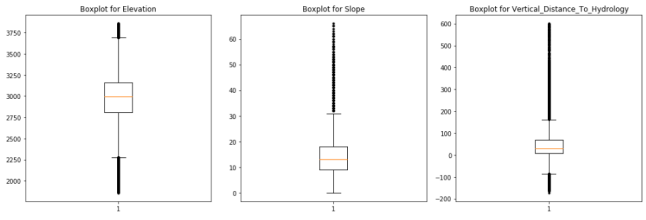


Fig. 5. Diagrama de extremos e quartis para algumas variáveis do CT

As classes apresentam distribuições diferentes para determinadas variáveis, porém a quantidade de possibilidades da variável alvo dificulta a visualização mais detalhada num gráfico (figura 6).

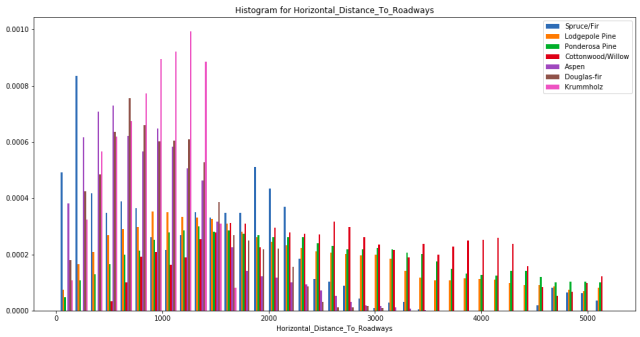


Fig. 6. Histograma diferenciando cada tipo de floresta para uma variável do segundo *dataset*

Por fim, o maior valor absoluto de correlação entre as variáveis foi de 0.78, portanto o grupo decidiu não alterar o segundo *dataset* quanto às correlações.

3 PRÉ-PROCESSAMENTO

No PDD, ter 3 amostras para um mesmo indivíduo poderia provocar *overfit* caso a divisão dos *datasets* de treino e de teste não fosse feito com muito cuidado. Esse problema poderia ocorrer, por exemplo, ao treinar um modelo com duas amostras de um paciente e testar com a terceira, gerando um resultado enviesado. A solução escolhida para isso foi calcular a média entre as 3 amostras do mesmo indivíduo.

O grupo testou essa hipótese separando os dados, ainda com 3 amostras de cada, em dois conjuntos, tomando o cuidado de deixar todas as amostras do mesmo indivíduo num mesmo conjunto. No primeiro conjunto, os dados foram usados para treino de um modelo kNN com validação cruzada, sendo possível haver observações do mesmo indivíduo em divisões internas diferentes, entre treino e teste. Em seguida, o modelo foi avaliado no segundo conjunto, a fim de detectar se houve *overfit*. O resultado do modelo no primeiro conjunto, para diferentes k's e métricas, encontra-se na figura 7. A validação no segundo conjunto, para apenas $k = 1$, foi de precisão de 74.56%, muito abaixo dos 98.15% do modelo com o mesmo k, o que indica que ocorreu *overfit*.

O número de variáveis do PDD foi reduzido devido a sua alta dimensionalidade. Os atributos altamente correlacionados (índice de correlação > 0.9) foram resumidos em um só, escolhendo-se somente o primeiro, pela ordem de leitura do *data frame*. Após isso, restaram 351 atributos. Para testar o impacto de um critério menos exigente, o processo foi repetido para correlações acima de 0.7 e 0.8, resultando em 208 e 256 atributos, respectivamente. Alguns desses índices de correlação já foram ilustrados na figura 3.

Como indicado na Exploração de Dados, ambos os *datasets* possuem apenas atributos numéricos e não apresentam valores ausentes, logo não foi necessário tratar dados faltantes ou efetuar *dummification*.

A maioria das variáveis dos dois *datasets* apresentam escalas de valores diferentes, o que influencia os resultados de alguns algoritmos. O caso mais evidente do problema é no KNN, porque a distância/similaridade entre objetos é de crucial importância no modelo. Logo escalas diferentes indicariam relevâncias falsas para

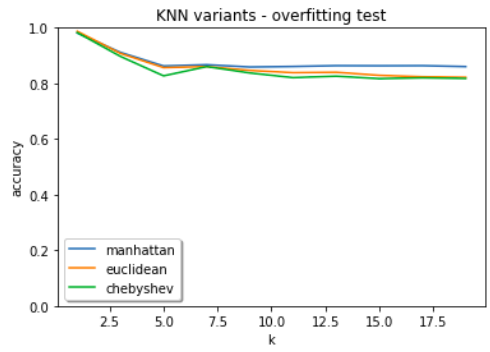


Fig. 7. Teste para verificar *overfit* devido às 3 amostras por indivíduo no PDD

id	gender	PPE	DFA	RPDE	numPulses	stdDevPeriodPulses	locPctJitter	locShimmer	meanAutoCorrHarmonicity	minIntensity	...	tqwt_kurto
0	1.0	0.916232	0.475707	0.590235	0.284974	0.022323	0.075900	0.271284	0.944323	0.692855	...	
1	0.0	0.182314	0.806469	0.630075	0.242228	0.689703	0.247423	0.228818	0.876238	0.882924	...	
2	1.0	0.877689	0.216841	0.516437	0.448187	0.033848	0.087255	0.117413	0.974092	0.871010	...	
3	0.0	0.925807	0.237611	0.542708	0.785622	0.010547	0.021814	0.161006	0.983185	0.781731	...	
4	0.0	0.930451	0.757013	0.842307	0.532383	0.754983	0.148962	0.722861	0.734980	0.497109	...	

	Elevation	Aspect	Slope	Horizontal_Distance_To_Hydrology	Vertical_Distance_To_Hydrology	Horizontal_Distance_To_Roadways	Hillshade_9am	Hillsh
0	0.368684	0.141667	0.045455	0.184681	0.223514	0.071659	0.870079	
1	0.365683	0.155556	0.030303	0.151754	0.215762	0.054798	0.866142	
2	0.472736	0.386111	0.136364	0.191840	0.307494	0.446817	0.921260	
3	0.463232	0.430556	0.272727	0.173228	0.375969	0.434172	0.937008	
4	0.368184	0.125000	0.030303	0.109520	0.222222	0.054939	0.866142	

Fig. 8. Dados normalizados dos *datasets* (PDD acima e CT abaixo).

variações de certos atributos. Por esse motivo, foi feita a normalização das *features*, para que ficassem no intervalo de 0 a 1. A figura 8 mostra parte do resultado.

O processo de *Feature Selection* foi feito com diferentes números de variáveis para avaliar a variação das métricas dos modelos e averiguar "maldição da dimensionalidade" em um conjunto de modelos (*ensemble*) e em um modelo único. Também foi feito *Feature Extraction* com PCA, gerando variáveis que explicassem 85% da variação da base de dados para comparar com os outros resultados (foram necessárias 56 variáveis no primeiro *dataset* e 18 no segundo). A figura 9 mostra o quanto de variação que as 10 primeiras variáveis extraídas de cada base explica.

O primeiro *dataset*, apresenta cerca de 75% dos pacientes com a Doença de Parkinson. Isso significa que uma "previsão" para um número qualquer de pessoas que informasse que todas elas têm Parkinson teria precisão de 75%, que é uma acurácia razoável. Entretanto, não queremos um modelo que tenha uma "previsão" constante, e sim que aprenda com as características das observações. Para mitigar esse problema, é feito o balanceamento do conjunto de dados. O método escolhido para esse caso foi o *oversampling*, porque a quantidade de observações não é grande, então prefere-se gerar mais dados através dos atuais do que amostrar dentre os existentes. Foi utilizado o SMOTE para o efeito, gerando assim novas observações da classe minoritária baseando-se nas existentes. O resultado foram 2 grupos de 188 amostras, um com e outro sem a doença.

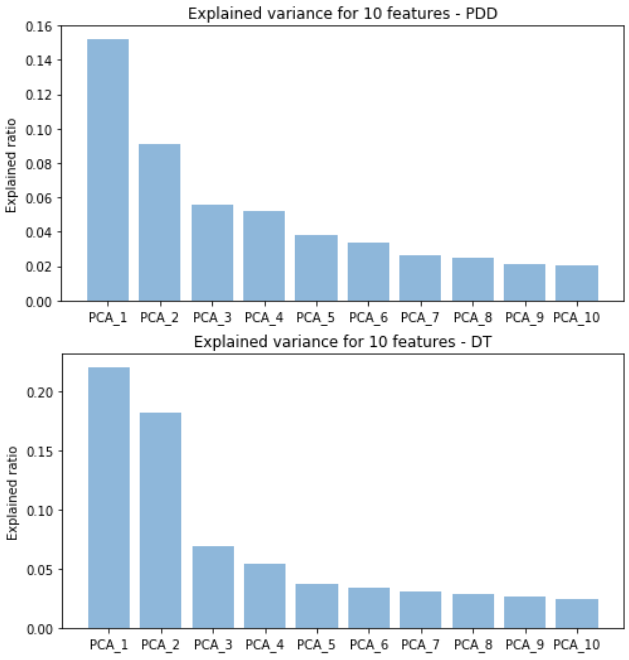


Fig. 9. Variação explicada por 10 variáveis extraídas de ambos os *datasets*

O CT apresenta a distribuição já apresentada na figura 4, ou seja, também encontra-se desbalanceado. A técnica utilizada para seu balanceamento foi diferente do PDD devido à grande quantidade de instâncias, sendo um volume suficiente para o treino dos modelos. O grupo optou por amostrar 2747 (número de objetos da classe minoritária) observações aleatórias em todas as classes (*Random Undersampling*).

4 APRENDIZADO NÃO SUPERVISIONADO

As duas bases de dados apresentam problemas de classificação, por esse motivo, a abordagem escolhida pelo grupo foi aplicar técnicas de aprendizado não supervisionado para encontrar padrões e regras que ajudem a entender os dados.

4.1 Regras de Associação

Para aplicar a busca de padrões e relações nos dados, as seguintes técnicas foram utilizadas: seleção de variáveis; discretização dos dados (por faixas de valores de mesmo tamanho - cut - ou por quantidade de elementos - qcut); *dummification* dos valores. Depois desses passos, foram aprendidos padrões para as quantidades de atributos selecionados: 5, 8, 10, 20, 30, e para as quantidades de divisões na discretização: 3, 4, 5, 8. A figura 10 apresenta uma das métricas avaliadas. Em ambos os casos, o "qcut" apresentou resultados melhores. Isso deve ter ocorrido porque discretizar mantendo o mesmo número de elementos por divisão diminui a influência de *outliers* e evita que a maioria dos dados recebam o mesmo valor discretizado (que fazia o suporte ser maior, mas o *lift* menor).

Para a primeira base de dados, foram fixadas 10 variáveis e 8 divisões com a discretização "qcut" para avaliar as regras geradas. Pela figura 11, podemos notar que valores "_7", ou seja, no grupo de

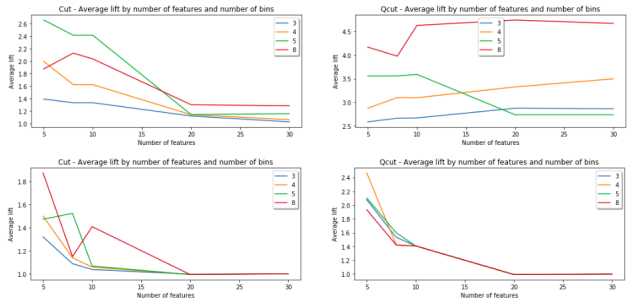


Fig. 10. *Lift* médio para diferentes valores de parâmetros para o primeiro *dataset* em cima e para o segundo *dataset* embaixo

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	antecedent_len
8	(std_7th_delta_7, std_6th_delta_7)	(class_1.0)	0.093085	0.50000	0.093085	1.000000	2.000000	0.046543	inf	2
9	(std_6th_delta_7, std_8th_delta_7)	(std_9th_delta_7)	0.095745	0.12500	0.090426	0.944444	7.555556	0.078457	15.750000	2
10	(std_6th_delta_7, std_8th_delta_7)	(class_1.0)	0.095745	0.50000	0.095745	1.000000	2.000000	0.047872	inf	2
11	(std_6th_delta_7, std_9th_delta_7)	(class_1.0)	0.103723	0.50000	0.103723	1.000000	2.000000	0.051862	inf	2
12	(std_6th_delta_7, std_10th_delta_7)	(class_1.0)	0.098404	0.50000	0.098404	1.000000	2.000000	0.049202	inf	2
13	(std_11th_delta_7, std_6th_delta_7)	(class_1.0)	0.095745	0.50000	0.095745	1.000000	2.000000	0.047872	inf	2
14	(std_6th_delta_7, std_12th_delta_7)	(class_1.0)	0.093085	0.50000	0.093085	1.000000	2.000000	0.046543	inf	2

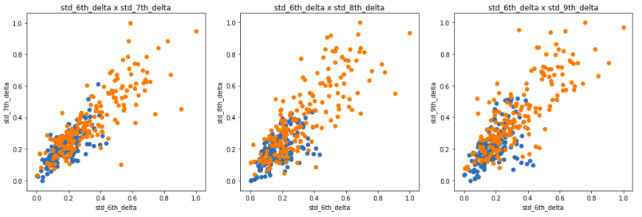


Fig. 11. Algumas regras relacionadas ao primeiro *dataset* e um gráfico de dispersão ilustrando as regras

valores mais altos, resultam na presença de Parkinson (classe 1). Isso ocorre para diferentes combinações de delta (variáveis), então valores altos nos mesmos podem ser um sinal de alerta. Tais atributos são os desvios padrões dos coeficientes da Transformada Wavelet Discreta; se eles são altos, isso pode indicar mais ruído/tremor na voz, e portanto uma dificuldade maior de descrever o sinal em coeficientes exatos. Apesar do suporte ser relativamente baixo (em torno de 10%), a confiança é bastante alta (chegando até a 100%), e o *lift* é maior do que 1, indicando uma correlação positiva. Isso indica que os atributos não acontecem juntos com tanto frequência, mas, quando acontecem, são indícios da doença. Muitos outros padrões aprendidos descrevem regras já conhecidas pela correlação entre atributos, verificada no mapa de calor.

Para a segunda base de dados, foram utilizadas as 10 variáveis não binárias e 9 divisões por qcut. Foram gerados 368 padrões, mas apenas 13 com 2 ou mais elementos em antecedentes. O suporte médio foi de 0.033, a confiança média foi de 0.968, o *lift* médio foi de 0.0286 e o *leverage* médio foi de 7.448. Já é possível notar um *lift* maior, indicando maior correlação entre as variáveis. A figura 12 mostra algumas regras que resultam na classe. As que apresentam resultados mais interessantes são: elevação mediana e pouca distância de rodovias indicando o tipo Aspen (o suporte indica que

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(Elevation_0, Vertical_Distance_To_Hydrology_0)	(Horizontal_Distance_To_Hydrology_0)	0.041240	0.183889	0.038640	0.936948	5.095186	0.031056	12.943522
1	(Elevation_4, Horizontal_Distance_To_Roadways_0)	(Cover_Type_5)	0.028239	0.142857	0.027251	0.965009	6.755064	0.023216	24.496241
2	(Elevation_8, Horizontal_Distance_To_Roadways_8)	(Cover_Type_7)	0.027875	0.142857	0.026574	0.953358	6.673507	0.022592	18.377143
3	(Horizontal_Distance_To_Fire_Points_8, Elevati...	(Cover_Type_7)	0.029643	0.142857	0.027823	0.938596	6.570175	0.023588	13.959184
4	(Aspect_2, Hillshade_Noon_0)	(Hillshade_3pm_0)	0.028239	0.112382	0.027251	0.965009	8.586840	0.024077	25.367178

Fig. 12. Algumas regras relacionadas ao CT

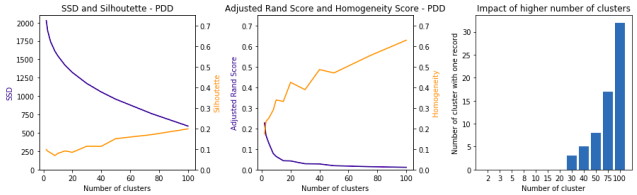


Fig. 13. Métricas do modelo *K-means* aplicado ao PDD

não é tão comum a ocorrência dos dois juntos, mas a confiança da regra é alta); grande distância de rodovias e elevação alta indicando o tipo Krummholz (mesma relação de suporte e confiança); elevação baixa, distância vertical e horizontal à hidrologia baixas indicando o tipo Cottonwood/Willow (mesma relação de suporte e confiança). Também é interessante verificar que elevação e distância vertical à hidrologia baixas resulta em distância horizontal à hidrologia baixa. Isso verifica-se, por exemplo, num rio ou lago.

4.2 *Clustering*

O grupo utilizou o modelo *K-means* para criar diferentes quantidades de *clusters* para verificar como são feitos os agrupamentos e se eles contribuem para um melhor entendimento dos dados. Os resultados obtidos utilizando as quantidades de clusters 2, 3, 5, 8, 10, 15, 20, 30, 40, 50, 75, 100 no PDD estão na figura 13.

No PDD, a soma das distâncias quadradas (SSD) entre instâncias do mesmo cluster apresenta-se bastante elevada para poucos grupos, provavelmente pela alta dimensionalidade do *dataset* e pela grande quantidade de *outliers*, que são dois fatores que influenciam neste modelo. Isso resulta em agrupamentos não tão próximos entre si. O aumento do número faz com que seja possível agrupar observações mais semelhantes, tanto em questão dos atributos (SSD), quanto das classes (homogeneidade); e separar as mais diferentes (silhueta). Já o *Rand Score* diminui com o aumento da quantidade de *clusters*, pois considera no cálculo o número de observações de uma mesma classe em clusters diferentes. O último gráfico da figura 13 mostra a quantidade de agrupamentos com apenas 1 registro. Isso pode ser usado como uma técnica de detecção de *outliers*, mas também pode indicar que os grupos gerados não conseguiram generalizar bem os dados.

Para testar o problema da alta dimensionalidade citado, foram selecionadas 10 variáveis através do *SelectKBest*, resultando em SSD's menores do que 100 e comportamentos similares aos gráficos citados para as outras métricas. Para visualizar melhor o resultado, foi fixada a quantidade de clusters igual a 5, que aparenta ser o ponto onde a diminuição do SSD começa a suavizar, e que não é grande a ponto

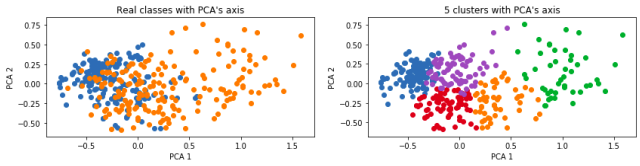


Fig. 14. Representação das classes à esquerda e dos *clusters* à direita, para o PDD

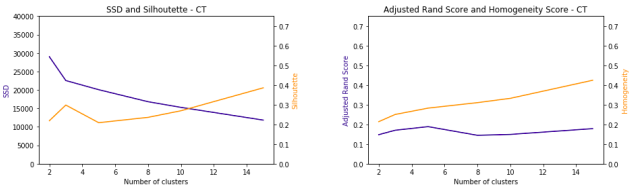


Fig. 15. Métricas do modelo *K-means* aplicado no CT

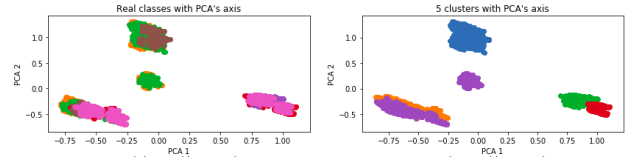


Fig. 16. Representação das classes à esquerda e dos *clusters* à direita, para o CT

de atrapalhar a visualização. Depois, foi criado um gráfico de dispersão utilizando as duas principais dimensões extraídas utilizando *Principal Component Analysis (PCA)* (figura 14).

Os pontos em verde e em azul conseguem separar muito bem as classes, podendo assim até serem utilizados como uma forma de classificação. Já os demais grupos poderiam ser analisados por técnicas de classificação diferentes para analisar se a separação é benéfica para as predições. Também é possível que com mais dimensões no PCA, a separação fosse mais clara.

A mesma técnica foi utilizada para o CT, porém com as quantidades 2, 3, 5, 8, 10, 15. As métricas encontram-se na figura 15. Podemos perceber que o SSD é muito maior, isso ocorre devido a existência de varíaveis binárias, aumentando muito a distância de observações com este atributo diferente. E a homogeneidade é menor, algo fácil de se visualizar através da figura 16

Portanto, devido a grande quantidade de classes relativamente próximas, a aplicação desta técnica no CT seria ineficiente.

5 CLASSIFICAÇÃO

Para os modelos de classificação, foi utilizado o método de validação cruzada estratificada do tipo *k-fold* com repetição, com 4 divisões (*folds*) e 3 repetições. Este método foi escolhido para evitar problemas com *overfitting*, principalmente no PDD, que possui apenas algumas centenas de dados.

As métricas que serão analisadas são acurácia, para saber a porcentagem de acerto dos modelos, e sensibilidade, para saber a relação entre verdadeiros positivos e todos os positivos. A sensibilidade é

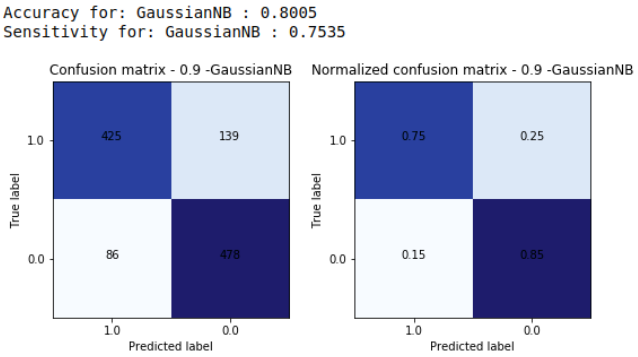


Fig. 17. Métricas do modelo *Naive Bayes* para o PDD (acima), matriz de confusão com valores absolutos contando repetições (à esquerda) e matriz de confusão com valores normalizados (à direita)

uma métrica muito importante para a área da saúde, pois evidencia doentes não identificados.

Uma observação importante é que, no CT, devido a variável alvo ser multiclasse, foi calculada uma métrica global baseada em verdadeiros positivos e falsos positivos (*average='micro'*), portanto os valores de acurácia e de sensibilidade apresentam-se iguais.

5.1 *Naive Bayes*

O modelo *Naive Bayes* foi testado com os estimadores *GaussianNB, MultinomialNB e BernoulliNB*. No PDD, o melhor estimador para a acurácia foi o *Gaussian* (figura 17). O grupo concluiu que, mesmo assumindo independência entre as variáveis (característica do modelo em questão), as métricas foram relativamente boas - acurácia: 80%, sensibilidade: 75%. O melhor estimador para a sensibilidade foi o *Multinomial*, com acurácia: 78%, sensibilidade: 78%. Logo, se considerarmos os dois indicadores igualmente importantes, as performances foram muito similares, e qualquer um pode ser utilizado como linha de base para os modelos seguintes, que são mais robustos.

No CT, a melhor perfomance foi para o estimador *Bernoulli* (figura 18), devido a grande quantidade de variáveis binárias (explicadas na Exploração dos dados). Porém, essas variáveis estão correlacionadas, contrariando o que o modelo assume e refletindo negativamente nas métricas. Outra dificuldade é dada pela maior quantidade de possibilidades de classe alvo, ou seja, é preciso diferenciar mais grupos entre si e não apenas 2, como no PDD. Portanto, novamente, este modelo servirá para apenas fornecer uma linha de base para os outros modelos.

5.2 Aprendizado baseado em instâncias

O modelo *KNN* foi testado com diferentes valores de vizinhos (para o PDD: k = 1, 3, 5, 7, 9, 11, 13, 15, 17, 19; para o CT: mesmos valores no intervalo de 1 a 13) e diferentes tipos de métricas de distância (*'manhattan', 'euclidean', 'chebyshev'*). O resultado para o PDD encontra-se na figura 19. O grupo notou que o resultado foi semelhante ao *Naive Bayes*, e formulou a hipótese de que a dimensionalidade da base de dados prejudicou o resultado ("maldição da dimensionalidade"). Por esse motivo, foi aplicada uma função para

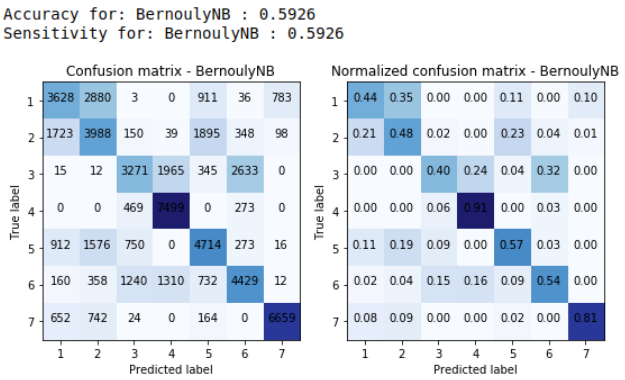


Fig. 18. Métricas do modelo *Naive Bayes* para o CT (acima), matriz de confusão com valores absolutos contando repetições (à esquerda) e matriz de confusão com valores normalizados (à direita)

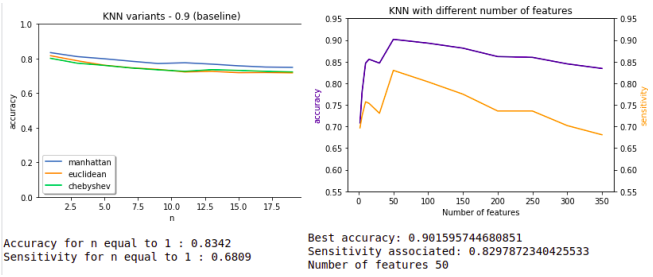


Fig. 19. Métricas do modelo *KNN* para o PDD sem seleção de atributos (à esquerda) e com (à direita)

selecionar atributos (explicada no Pré-processamento) e então foi retreinado o modelo para $k = 1$ e usando a distância *manhattan* (resultado também na figura 19). A perfomance obtida foi muito melhor, inclusive percebe-se que a partir de certo ponto, a acurácia e a sensibilidade do modelo começam a decrescer, sustentando a hipótese formulada.

Ao testar o impacto da extração de variáveis através de *PCA* (56 variáveis extraídas), o resultado foi semelhante ao com seleção de *features* (com 50 variáveis).

Para o CT, os resultados estão apresentados na figura 20, e o grupo percebeu que, por existirem menos atributos, o impacto da seleção é bem menos significativo. Além disso, a performance foi muito melhor do que com *Naive Bayes*. A justificativa formulada foi que não assumir independência entre as variáveis e verificar a proximidade entre as classes conseguiu generalizar melhor a base de dados.

Um detalhe interessante nos dois *datasets* é que o melhor resultado está relacionado ao $k = 1$. A hipótese do grupo é que cada atributo apresenta um pouco de informação em relação a classe alvo, portanto o vizinho mais próximo do registro a ser classificado, ou seja, a combinação de variáveis mais similares, representa a classe mais provável do mesmo.

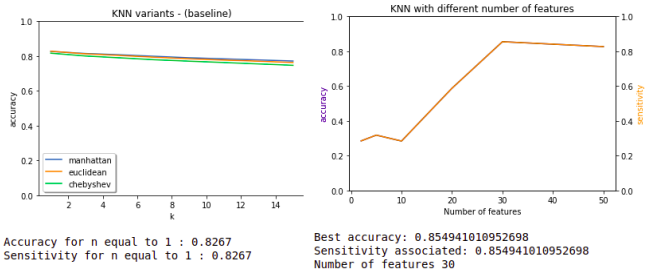


Fig. 20. Métricas do modelo *KNN* para o CT sem seleção de atributos (à esquerda) e com (à direita)

5.3 Árvores de decisão

Para o treinamento e avaliação de modelos de Árvores de Decisão nos *datasets*, decidiu-se variar os seguintes parâmetros: porcentagem mínima de observações para formar uma folha, profundidade máxima da árvore (5, 10, 25 ou 50) e o critério utilizado para selecionar a melhor variável a ser dividida numa ramificação. Os critérios avaliados foram o uso do ganho de informação (entropia) das variáveis e o uso do índice Gini de impureza.

O uso da entropia apresentou melhores resultados nos dois conjuntos de dados. O valor de profundidade máxima da árvore com melhores acurácias também foi comum aos dois, sendo 50, o que faz sentido uma vez que quanto maior a árvore, melhor ela consegue separar os dados. No PDD, ainda foi possível avaliar que a acurácia e sensibilidade aumentaram com o número de variáveis (maior índice de correlação para agrupar atributos). Para tal dataset, a melhor acurácia (84.04%) deu-se com porcentagem de amostras mínima igual a 0.5%, enquanto para o outro (76.43%) foi de 0.1%. Para o CT, os parâmetros que obtiveram melhor acurácia também apresentaram melhor sensibilidade, ao contrário do PDD, em que tal valor foi máximo (82.80%) para profundidade máxima de 25 e folhas com mínimo de 0.75% das amostras.

Com esses testes, foi possível verificar que, de facto, as árvores de decisão performam bem para conjuntos de dados com alta dimensionalidade, como o PDD. A baixa acurácia no CT talvez se explique pelo fato de que os dados estão muito próximos.

É relevante observar que para o CT, os parâmetros de qualidade do modelo não mudam tanto a partir da profundidade máxima de 10 e quase nada entre 25 e 50. Nesse caso, escolher uma profundidade maior pode estar só a adicionar uma complexidade desnecessária à *Decision Tree*.

Na figura 21, é apresentada a acurácia dos modelos pela porcentagem mínima de amostras por folha (eixo) e profundidade máxima das árvores (linhas), usando o critério de ganho de informação. Observa-se que o PDD apresenta um comportamento muito mais instável que o CT, provavelmente por ter menos dados, o que significa que a porcentagem mínima interfere mais.

5.4 *Random Forest*

Quanto ao modelo de *Random Forest*, foram feitos diversos testes para estudar o efeito do número de estimadores (árvores), a profundidade máxima das árvores e o número de *features* máxima por *Decision Tree*, que pode ser $\log_2(\#features)$ ou $\sqrt{\#features}$. Esses

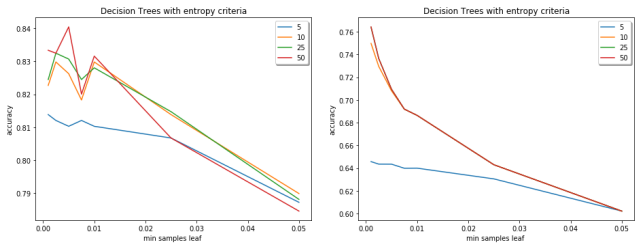


Fig. 21. Acurácia das árvores de decisão para o PDD (esquerda) e o CT (direita)

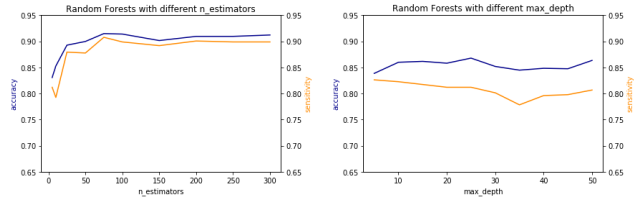


Fig. 22. Qualidade do *Random Forest* para diferentes números de estimadores e profundidade máxima da árvore (PDD)

testes são usados para estimar a performance do modelo para os parâmetros ideais.

Para os dois casos, a acurácia e a sensibilidade aumentam significantemente com o número de estimadores até certo ponto (por volta de 75) e depois passam a serem lineares ou aumentarem pouco, o que é natural. Em relação ao número de variáveis, ambas as opções performam igual, então escolheu-se a função de raiz quadrada para ser usada a seguir.

No PDD, as máximas acurácia e sensibilidade são para 75 estimadores, apesar de que esse valor varia entre iterações do programa. Quanto à profundidade máxima no PDD, os parâmetros de qualidade parecem lineares, então escolhe-se o valor intermédio de 25. Para este *dataset* e usando os parâmetros anteriores, a acurácia do *Random Forest* foi de 91.13% e a sensibilidade de 90.07%. Os gráficos para os primeiros dois fatores encontram-se na figura 22.

Quanto ao CT, o número de estimadores ótimo obtido é 250. Devido à estabilização citada anteriormente, o valor de 100 foi escolhido para a avaliação final, uma vez que que um valor maior só estaria adicionando atraso ao programa, sem ganhos práticos na performance. O aumento da profundidade máxima melhora a acurácia até chegar em aproximadamente 20. Depois disso, a melhoria é muito pequena, pelo que foi escolhido o valor 25, usando o raciocínio anterior. Os gráficos correspondentes a essas análises encontram-se na figura 23 Com tais parâmetros, obteve-se acurácia e sensibilidade de 86.57%. As aproximações feitas fizeram com que o modelo perdesse 0.3 ponto percentual de acurácia, mas o tempo foi reduzido em mais de metade, com 40s a menos.

Para avaliar o impacto da quantidade de atributos fornecidos ao modelo, foi efetuada seleção de variáveis para treinar o modelo para a base CT, utilizando os parâmetros citados no parágrafo anterior. A quantidade de variáveis testadas foi: 2, 5, 10, 20, 30, 40, 50. A figura 24 indica os resultados, e com isso, podemos concluir que a seleção

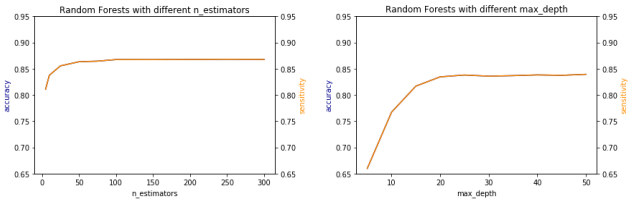
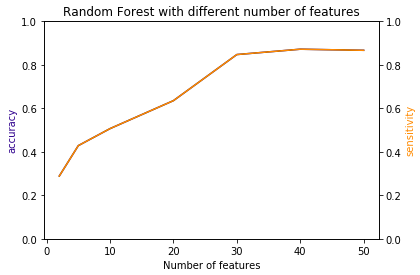


Fig. 23. Qualidade do *Random Forest* para diferentes números de estimadores e profundidade máxima da árvore (CT)




```
Best accuracy: 0.8719295251594387
Sensitivity associated: 0.8719295251594387
Number of features 40
```

Fig. 24. Métricas do modelo *Random Forest* (CT) com seleção de variáveis

de atributos é menos relevantes nesta base de dados, provavelmente devido a baixa dimensionalidade, que foi a mesma conclusão de quando aplicamos no *KNN*.

5.5 *XGBoost*

O *Gradient Boosting* foi treinado com a variação dos seguintes parâmetros: taxa de aprendizado (0.05, 0.1, 0.2, 0.25, 0.3, 0.5, 1); número de estágios de *boost* (5, 10, 25, 50, 75, 100, 150, 200, 250, 300); profundidade máxima dos estimadores (3, 5, 10, 25, 50); o número de variáveis a ser considerado('sqrt', 'log2').

A taxa de aprendizado representa o ajuste no modelo após cada iteração, portanto altos valores representam mudanças mais acentuadas, prejudicando a perfomance do modelo. Quanto ao número de estimadores, uma vez que o módulo é robusto quanto a overfitting, a maior quantidade de estimadores resultou em melhores métricas, aparentando convergir após certo número. Um baixo valor de profundida máximo se provou melhor no PDD, provavelmente, devido ao pequeno número de registros, um número maior poderia levar a *overfit*). Já no CT, as variações forem menores, com pico em 10. A figura 25 apresenta algumas métricas dos modelos.

No PD, utilizando os parâmetros: taxa de aprendizagem igual a 0.2, número de estimadores igual a 200, profundidade máxima igual a 3 e número de variáveis como 'log2', a acurácia foi de 91.67% e a sensibilidade de 89.72% No CT, utilizando os parâmetros: taxa de aprendizagem igual a 0.5, número de estimadores igual a 300, profundidade máxima igual a 10 e número de variáveis como 'sqrt', a acurácia/sensibilidade foram de 86%.

Este modelo apresentou a melhor perfomance para o PDD, portanto o grupo decidiu verificar o impacto da aplicação de seleção de variáveis. Fixamos os parâmetros baseado no resultados obtidos:

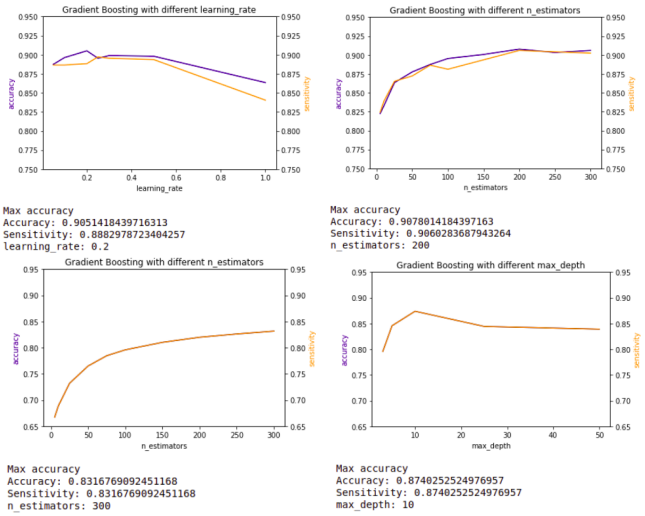


Fig. 25. Métricas do modelo *Gradient Boosting* (PDD acima e CT abaixo)

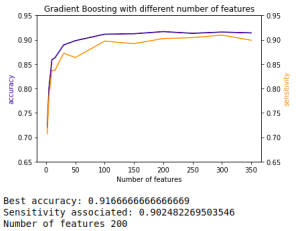


Fig. 26. Métricas do modelo *Gradient Boosting* (PDD) com seleção de variáveis

taxa de aprendizado em 250, número de estimadores em 250, profundidade máxima em 3 e número máximo de variáveis em 'sqrt'; e treinamos com os seguintes número de variáveis: 2, 5, 10, 15, 30, 50, 100, 150, 200, 250, 300, 350. Como indicado na figura 26, após atingir cerca de 100 variáveis, as métricas variam pouco. Esse comportamento indica que modelos *ensemble*, por utilizarem diversos classificadores independentes, conseguem lidar bem com uma maior quantidade de variáveis do que num modelo simples (exemplificado pelo *KNN*). Inclusive, podem apresentar melhores métricas com mais indicadores por terem a possibilidade de criarem mais estimadores diferentes. Portanto, a seleção de atributos se mostra mais relevante para esses outros modelos.

6 CONCLUSÃO

Ao longo do projeto, foi possível notar vantagens e desvantagens de cada modelo, principalmente por serem utilizados *datasets* com características tão diferentes (tamanho e dimensionalidade). Os resultados resumidos encontram-se na tabela 1:

Os modelos que envolvem apenas um classificador (*Naive Bayes, KNN*, Árvore de decisão) apresentaram-se mais suscetíveis a variações devido a técnicas de pré-processamento. Por exemplo, a normalização afetaria o *kNN* e a premissa de que as variáveis são

		NB	KNN	KNN-FS	DT	RF	RF-FS	GB	GB-FS
PDD	Acurácia	0.8005	0.8342	0.9016	0.8404	0.9113		0.9167	0.9167
	Sensibilidade	0.7535	0.6809	0.8298	0.7643	0.9007		0.8972	0.9025
CT	Acurácia	0.5926	0.8267	0.8549	0.8280	0.8657	0.8719	0.8600	
	Sensibilidade	0.5926	0.8267	0.8549	0.8280	0.8657	0.8719	0.8600	

Tabela 1. Resultados finais para os modelos

independentes resultou em performances ruins para o *Naive Bayes* com o CT, que tem quase todas as variáveis correlacionadas (como o tipo de solo). Enquanto isso, os que envolvem mais classificadores (*Random Forest e XGBoost*), apresentaram-se mais robustos, performando bem mesmo sem os mesmos cuidados.

Uma forma de tentar melhorar a perfomance ainda mais seria juntar os resultados dos estudos não supervisionados na classificação, por exemplo tentando gerar novas variáveis utilizando padrões conhecidos (*pattern mining*) ou aplicando diferentes modelos para agrupamentos diferentes (*clustering*). Contudo, a combinação dessas técnicas não foi tão explorada no trabalho.

Por fim, dado os métodos aplicados pelo grupo durante o projeto, os modelos que performaram melhor e que seriam escolhidos para uso, caso necessário, seriam os métodos *ensemble*, por obterem as melhores métricas apresentadas. Entretanto, a diferença entre os dois não foi significativa, não sendo possível escolher entre eles com argumentos estatísticos. Seria preciso testar com mais dados, ou outros dados, para concluir.