PROJETO CIÊNCIA DE DADOS 2018

GRUPO 64

Projeto Data Mining

Kevin Batista Corrales, ist194131

Conteúdo

1. INTRODUÇÃO

Técnicas de aprendizagem supervisionada e sem supervisão foram usadas para a exploração de dois datasets. Utilizamos algoritmos como KNN, Nayve-Bayes, Decision Tress e Random Forests.

Várias técnicas de pré-processamento foram utilizadas, dependendo não só do algoritmo mas também do dataset em exploração.

2. PRÉ-PROCESSAMENTO

O pré-processamento é uma etapa importante no processo de Data Mining. É vital encontrar os erros antes de investir demasiado tempo na análise dos dados porque pode produzir resultados enganosos.

Geralmente, procuramos as respostas para as seguintes perguntas:

- Existe algo de errado nos dados?
- Existe algumas peculiaridades nos dados?
- Preciso de reparar ou remover dados?

2.1 Digital Colposcopies

A informação que conseguimos na análise do dataset Digital Colposcopies é que existe 69 atributos para cada dataset (green, hinselmann, schiller).

Os datasets green, hinselmann e schiller tem 6762, 6693, 6348 entradas respectivamente.

Através dos comandos anteriores foi verificado que não existe dados vazios/perdidos nos três datasets.

No entanto será necessário a divisão dos datasets por training e test sets para evitar *overfitting*.

2.2 APS failure at Scania trucks

A informação que conseguimos na análise do dataset APS é que existe 171 atributos e 2 classes: negativo e positivo.

O dataset já está dividido em train e test set: O train set tem 60000 entradas, 59000 das quais são da classe negativo e as restantes 1000 são da classe positive, o que indica que o dataset não é balanceado. O test set tem 16000 entradas .

Temos então aproximadamente 73% de dados para treino e os restantes 27% para teste.

Os atributos são numéricos, não é necessário fazer labeling.

Existem duas classes:

Negativo - Não existe problema com o APS

Positivo - Existe problema com o APS

Devido ao facto de existir grandes quantidades de dados vazios/perdidos (850015) será necessário eliminar as colunas com percentagem superior a 45% de dados vazios.

O valor vazio/perdido nos datasets APS failure at Scania trucks são representados por ‘na’, mas como ‘na’ é visto pela linguagem de programação python como string e não um valor NaN, foi necessário utilizer o metódo *dataset*.replace(‘na’,np.nan’) para mais facilitar a manipulação do dataset.

Após a eliminação de colunas com grandes percentagem de dados vazios/perdidos será substituido os restantes dados vazios/perdidos pelo valor médio da coluna a que pertencem, para o cálculo da média será necessário a separação da coluna ‘class’ com as colunas com valores.

Depois do tratamento de dados vazios/perdidos é verificado o balanceamento dos dados a partir das classes.

Como a maior parte das entradas são da classe negativa (98,3%), se não balancearmos o dataset, o novo classificador vai ser biased em relação a essa mesma classe, o que vai levar a mais Falsos negativos, o que no problema em questão tem grande custo.

Devemos normalizar os se estivermos a comparar dois valores e estes se encontram bastante distantes.

Neste dataset, é necessário pois uma variação de 1 na temperatura não deve ser tratada exactamente como a variação de 1 na pressão dos pneus. É importante fazer para o algoritmo KNN, mas não para o Nayve Bayes ou para as decision-trees, pois estes são not distance based.

3. EXPLORAÇÃO

3.1 Digital Colposcopies

3.1.1 Métodos e Parametrização

A métrica usada em cada algoritmo desprovisionado seguinte será precisão (*Accuracy*) e medição F1 (*F1-measure*).

Em cada algoritmo foi usado técnica de Cross-Validation para obter amostras de datasets.

Foi utilizado como valores K: {5,10,15,20}

Naive Bayes

Naive Bayes foi usado como base para reparar todos os parâmetros do pre-processamento.

KNN

Para o modelo KNN usamos os parâmetros: número de vizinhos ‘k’ e métrica da distância ‘weight’.

Os valores k usados são impares porque e devido a complexidade do dataset foram utilizados valores como {1,3,5,7,9,11,13,15,17,19,21}.

Os valores da distância métrica (weight) foi uniforme onde o ‘weight’ é uma distância independente e onde o peso é proporcional em relação a distância.

Decision Trees

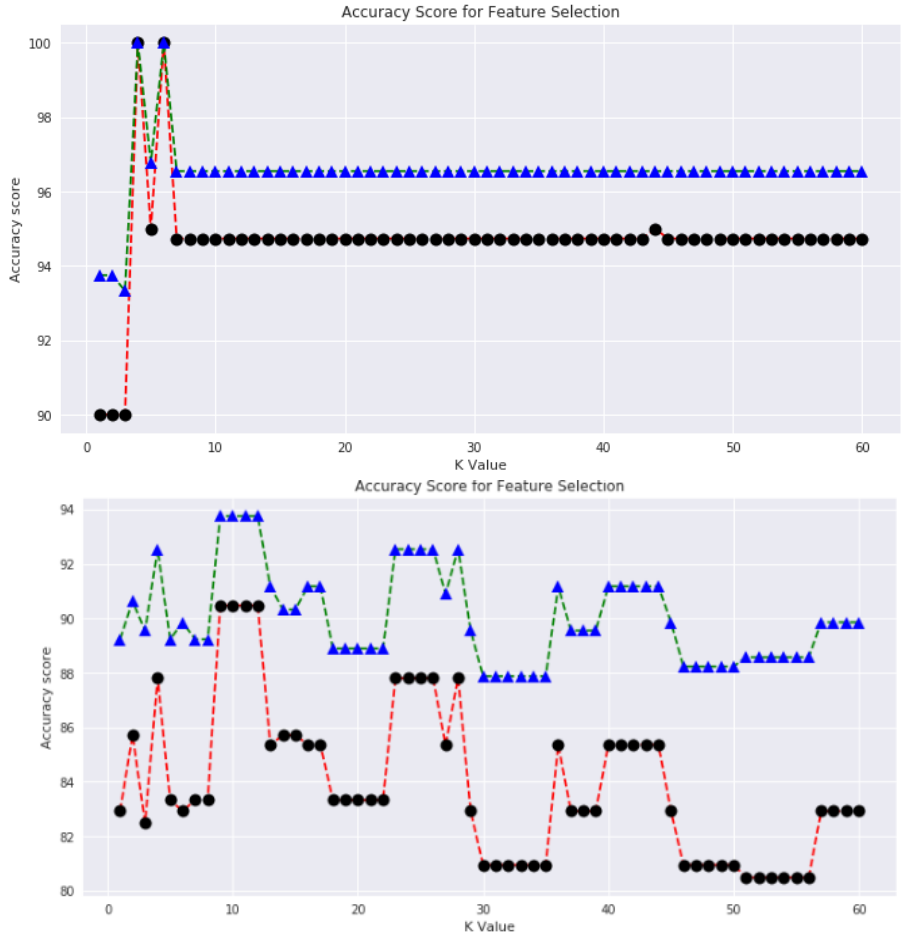
O algoritmo utilizado nas Decisions Trees foi CART, em que a base deste modelo é uma árvore não podada com profundidade de 20 nodes.
Para a iteração desde modelo usamos parâmetros como profundidade máxima da árvore e por node entre valores de 2 a 20.

Random Forests

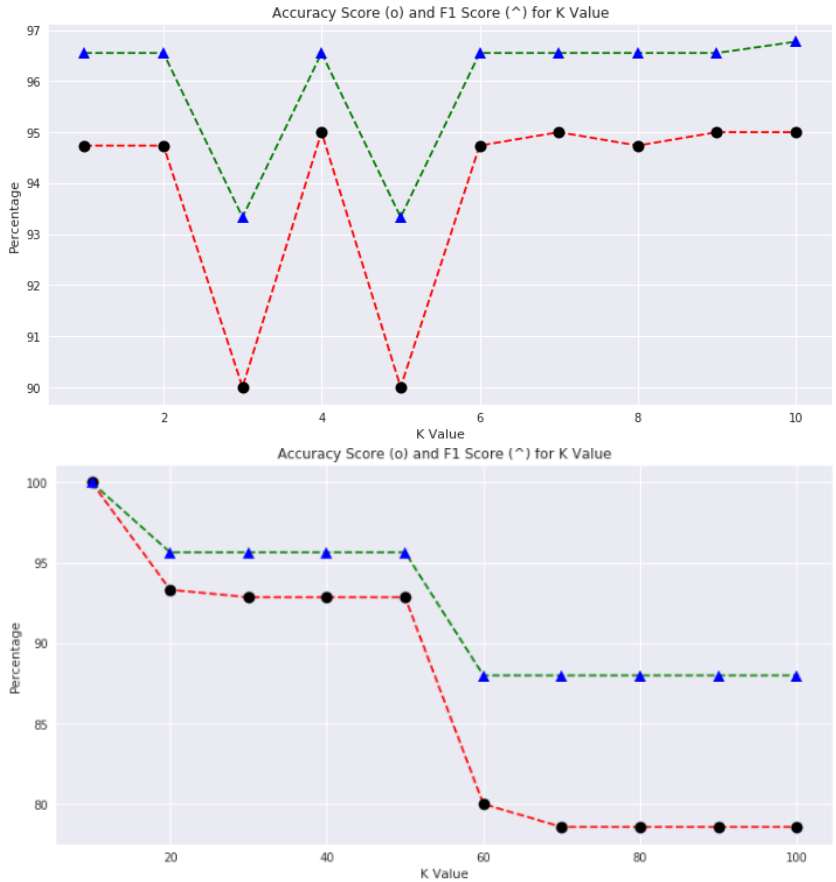
Para Random Forests usamos como parâmetro valores de 5 a 20 para número de árvores e valores de 2 a 20 para profundidade máxima em cada árvore.

3.1.2 Resultados

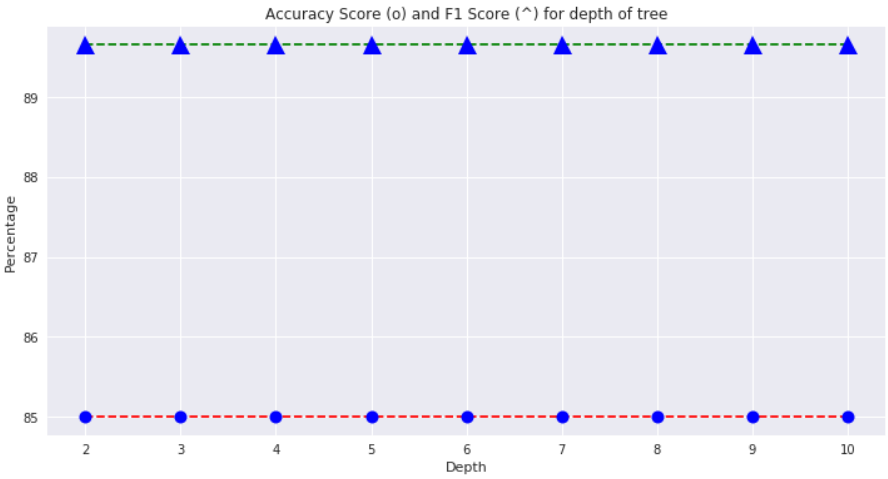
Resultados Naive Bayes:

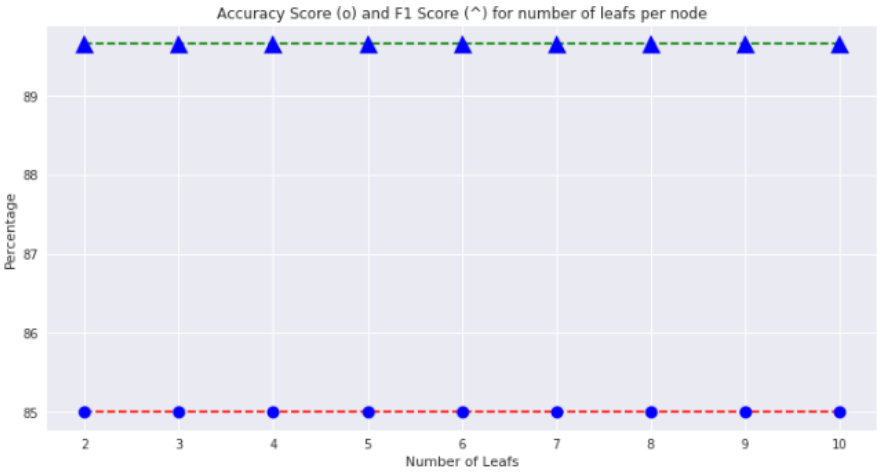


Resultados KNN:

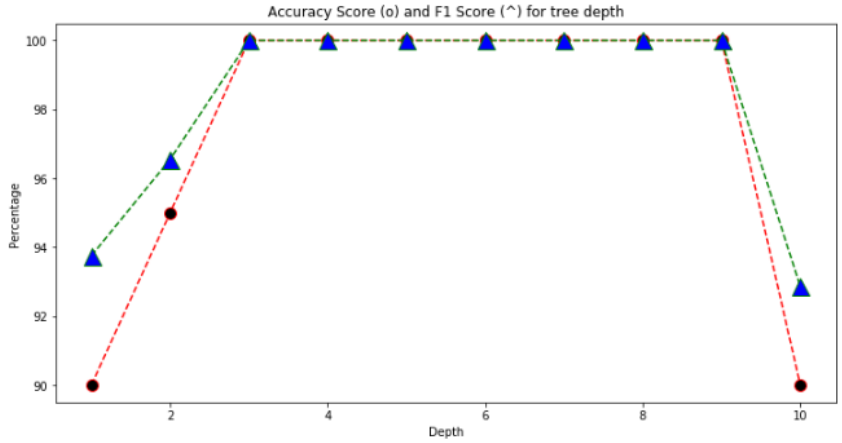


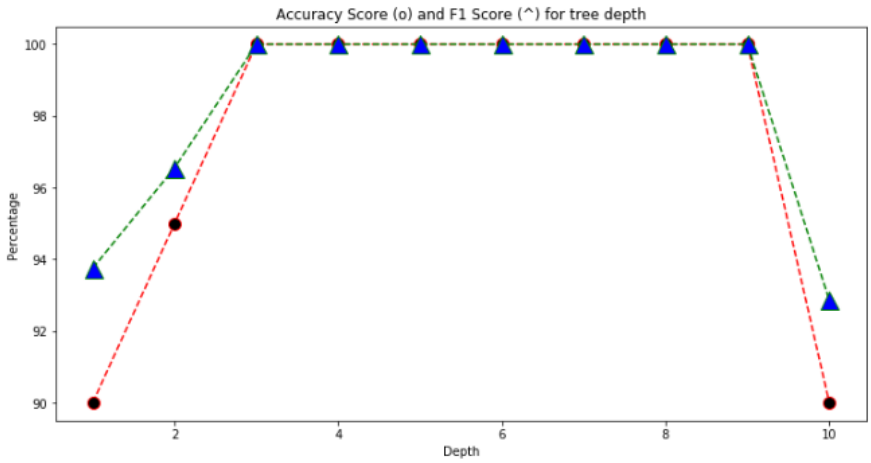
Resultados Decision Trees:





Resultados Random Forest:





3.2 APS failure at Scania trucks

3.2.1 Métodos e Parametrização

As métricas usadas em cada algoritmo supervisionado seguinte serão a pontuação do APS(APS score) e a precisão(Accuracy).

APS score = custo_falsos_negativos * total_positivos + custo_falsos_positivos * total_negativos

Supervisionado:

Naive Bayes

Naive Bayes foi usado como base para reparar todos os parâmetros do pre-processamento.

KNN

Para o modelo KNN usamos os parâmetros: número de vizinhos 'k' e métrica da distância 'weight'.

Os valores k usados são impares porque e devido a complexidade do dataset foram utilizados valores como {1,3,5,7,9,11}.

Os valores da distância métrica (weight) foi uniforme onde o 'weight' é uma distância independente e onde o peso é proporcional em relação a distância.

Decision Trees

O algoritmo utilizado nas Decisions Trees foi CART, em que a base deste modelo é uma árvore não podada com profundidade de 20 nodes.
Para a iteração desde modelo usamos parâmetros como profundidade máxima da árvore e por node entre valores de 2 a 20.

Random Forests
Para Random Forests usamos como parâmetro valores de 5 a 20 para número de árvores e valores de 2 a 20 para profundidade máxima em cada árvore.

Não Supervisionado:

Sub-Sampling

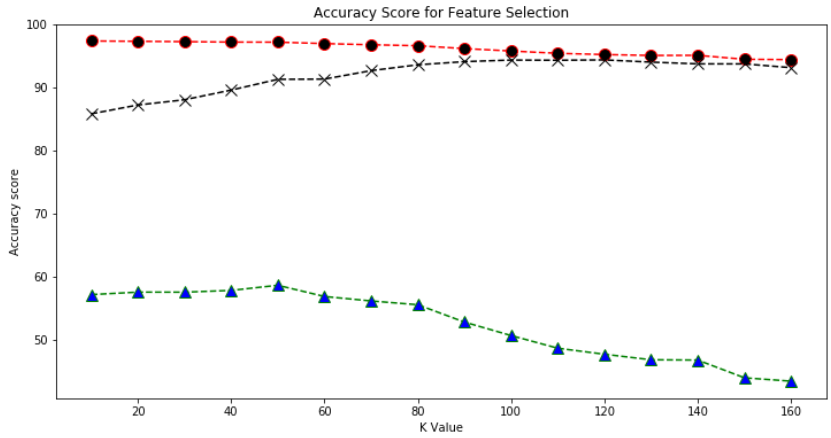
Como na aprendizagem não supervisionada não temos informação da classe o que implica a não existência de fase de treino e teste, decidimos concatenar os dois sets, do qual resultou 76000 instâncias. Utilizamos *stratified sub sampling* para gerar as instâncias para executar as computações em menor tempo.

K-Means

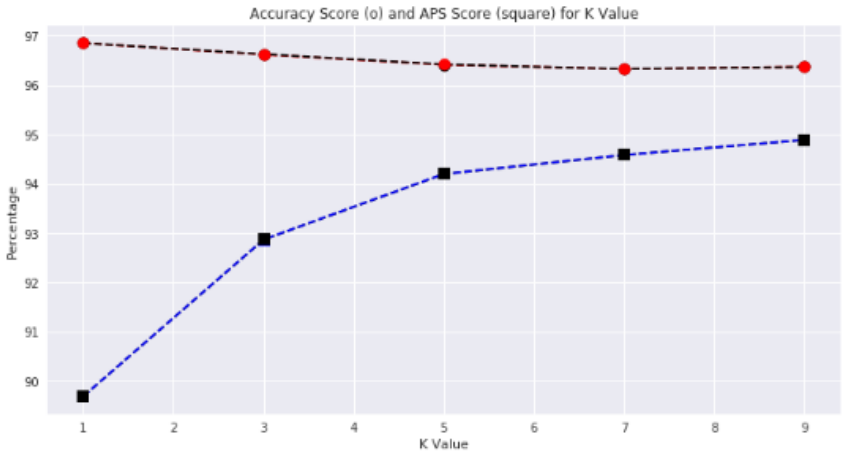
Em ordem para ser possível visualizar clusters, foi necessário reduzir o dataset. Utilizamos o algoritmo PCA que varia o número de *features* entre 2 a 150 e valores de k entre 2 a 9. Os gráficos indicam que o k e número de feature ótimo é 2.

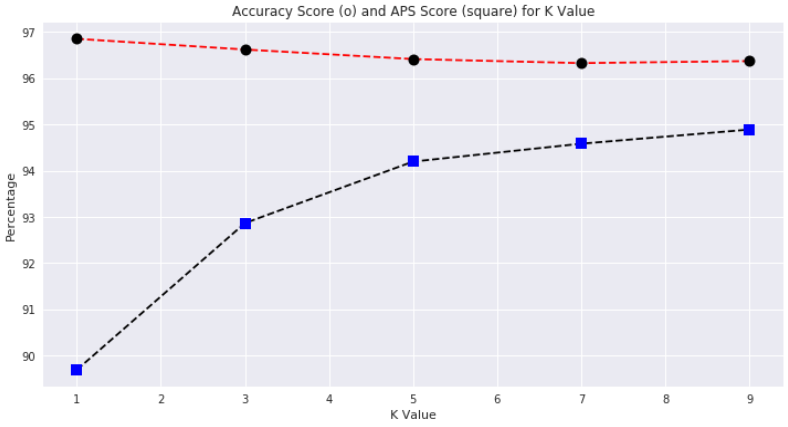
3.2.2 Resultados

Resultados Naive Bayes:

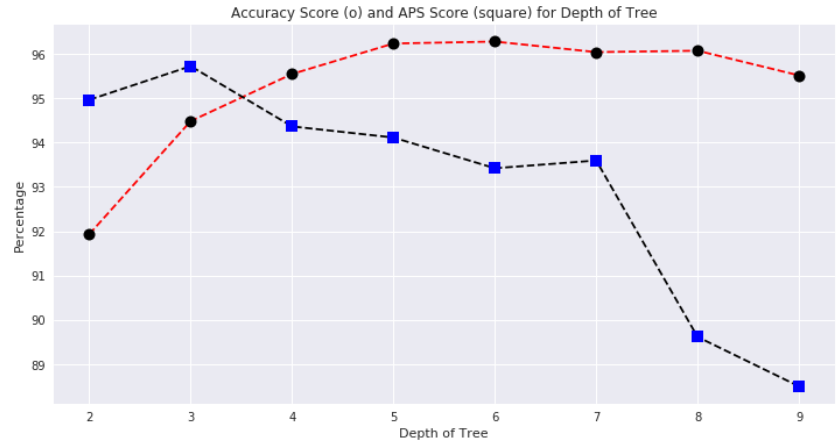


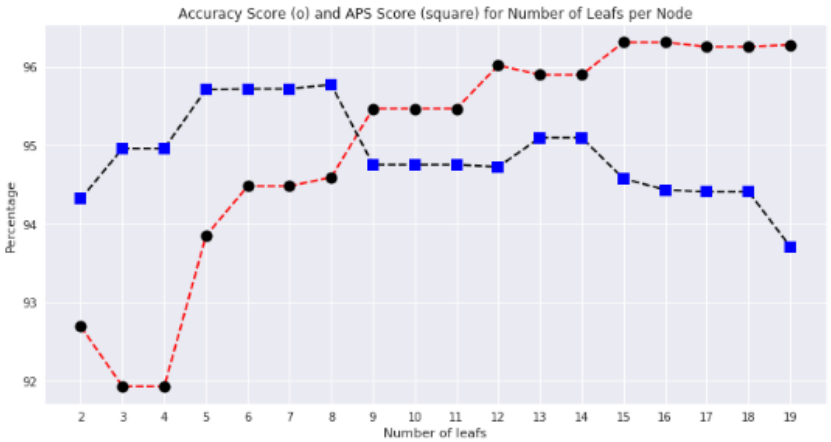
Resultados KNN:



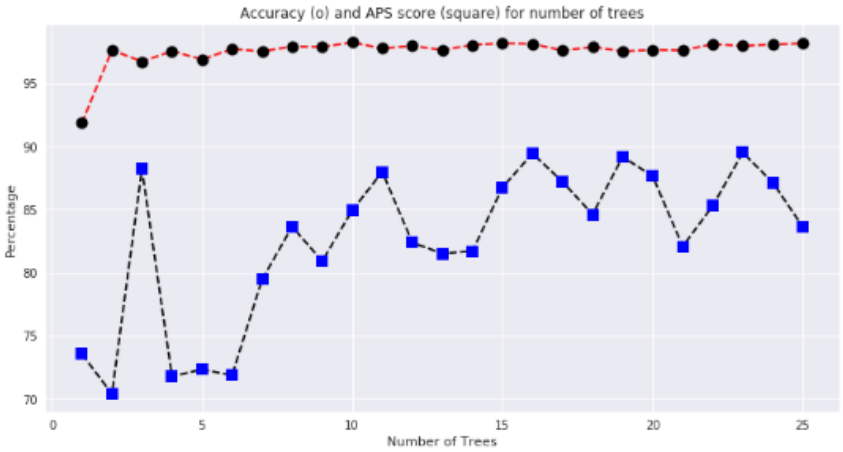


Resultados Decision Trees:





Resultados Random Forest:



4. ANÁLISE CRÍTICA

Na APS é verificado uma redução na APS score em relação com o aumento dos dados, excluíndo o algoritmo KNN em que existe um crescimento exponencial. No que se refere a Accuracy da APS é verificado um redução bastante pequena nos algoritmos Naive-Bayes e KNN enquanto nos algoritmos Decision Tree e Random Forest é verificado um crescimento bastante acentuado apenas no inicio com tendência a estabilizar.

Na Digital Colposcopies é verificado uma estabilização na F1 Score e Accuracy no algoritmos Decision Tree e no algoritmo Random Forest entre os valores de depth 3 e 9, existindo um crescimento acentuado anteriormente e uma redução acentuada posteriormente apenas no algoritmo Random Forest.

Ainda na Digital Colposcopies podemos ver que existe overfitting quando o valor ‘k’ é baixo devido a variação existente no gráfico.

5. CONCLUSÕES

A conclusão que conseguimos obter é que quanto maior a existência de dados em um dataset, será obtida melhor qualidade e certidão dos resultados gerados.

Podemos concluir que o dataset APS tem menos variação com KNN e Naive-Bayes enquanto o dataset Digital Colposcopies tem menos variação com Random Forest e Decision Tree, este facto é devido que o dataset APS encontra-se categorizado por classes.