

**Data Science Project 2018**

**TEAM 27**

**APS Failure at Scania Trucks**

&

**Quality Assessment of Digital Colposcopies**

Francisco Neves, ist424770

Francisco Catarrinho, ist424771

João Ramos, ist180915

**TABLE OF CONTENTS**

[1. Introduction 3](#_Toc529983058)

[2. Pre-processing 3](#_Toc529983059)

[**2.1 APS Failure at Scania Trucks 3**](#_APS_Failure_at)

[2.1.1 Unsupervised Learning 3](#_Toc529983061)

[2.1.2 Supervised Learning 3](#_Toc529983062)

[**2.2 Quality Assessment of Digital Colposcopies 3**](#_Toc529983063)

[2.2.1 Unsupervised Learning 3](#_Toc529983064)

[2.2.2 Supervised Learning 3](#_Toc529983065)

[3. Exploration 4](#_Toc529983066)

[**3.1 APS Failure at Scania Trucks 4**](#_Toc529983067)

[3.1.1 Unsupervised Learning 4](#_Toc529983068)

[3.1.1.1 Methods and Parametrization 4](#_Toc529983069)

[3.1.1.2 Results 4](#_Toc529983070)

[3.1.2 Supervised Learning 4](#_Toc529983071)

[3.1.2.1 Methods and Parametrization 4](#_Toc529983072)

[3.1.2.2 Results 4](#_Toc529983073)

[**3.2 Quality Assessment of Digital Colposcopies 4**](#_Toc529983074)

[3.2.1 Unsupervised Learning 4](#_Toc529983075)

[3.2.1.1 Methods and Parametrization 4](#_Toc529983076)

[3.2.1.2 Results 4](#_Toc529983077)

[3.2.2 Supervised Learning 5](#_Toc529983078)

[3.2.2.1 Methods and Parametrization 5](#_Toc529983079)

[3.2.2.2 Results 5](#_Toc529983080)

[4. Critical Analysis 5](#_Toc529983081)

[5. Conclusions 5](#_Toc529983082)

# Introduction

O presente relatório no âmbito da U.C. de Ciência de Dados tem como objetivo a análise de dois problemas distintos.

O primeiro problema, **APS Failure at Scania Trucks** fornece-nos um *dataset* com informação recolhida acerca do sistema APS de camiões da *Scania.* Este *dataset* apresenta 170 atributos contínuos e 1 atributo *target* binário, estando dividido num conjunto de treino com 60.000 observações e num conjunto de teste com 16.000 observações. O atributo *target* pode tomar os valores *pos* e *neg,* que indicam se ocorreu uma avaria ou não respetivamente. Para a avaliação dos resultados obtidos é-nos fornecida uma métrica dada por , onde é o número de falsos positivos e o número de falsos negativos na classificação, sendo o objetivo do problema minimizar a mesma.

No segundo problema, **Quality Assessment of Digital Colposcopies**, é-nos apresentada informação relativa a uma série de imagens capturadas em colposcopias usando 3 diferentes técnicas de visualização usadas neste tipo de exame. O *dataset* está dividido em 3 ficheiros, cada um correspondendo a uma das técnicas (*green, schiller* e *hinselmann*). Existem 62 atributos contínuos e 7 atributos *target*, sendo que 6 destes representam a avaliação dada por diferentes peritos do domínio acerca da qualidade das imagens. Já o último atributo representa o consenso entre os vários peritos. No total, o *dataset* tem 287 observações (98 no *green*, 92 no *schiller* e 97 no *hinselmann*).

# Pre-processing

## APS Failure at Scania Trucks

## Unsupervised Learning

* **Association Rules**

Devido ao elevado número de *missing values* presentes no *dataset* começou-se por analisar os resultados obtidos usando diferentes técnicas de imputação. No entanto, as *association rules* resultantes eram sempre distorcidas por estas substituições, pelo que se optou eliminar todas as linhas que contivessem *missing values* o que resultou em *association rules* de maior qualidade e veracidade. Foi transformado o *dataset* num *dataset* transaccional usando discretização *equal-width* e *equal-frequency* com diferentes *bins.* Para reduzir o elevado tempo de computação originado pelo elevado número de colunas do *dataset*, e de maneira a descobrir regras mais relevantes para o problema, foi feito um *feature selection* usando o método ANOVA onde se seleccionou as 10 melhores *features* para explicar o atributo *target*.

* **Clustering**

Utilizou-se o PCA para diminuir a dimensionalidade dos atributos assim como se fez normalização dos dados para não haver influência das diferentes escalas entre atributos na aplicação do algoritmo K-Means. Testaram-se várias técnicas para a remoção de *missing values* tendo a substituição pela mediana a técnica que melhores resultados apresentou. Testou-se ainda a remoção de *outliers*, no entanto decartámos esta transformação pois esta resultou em *clusters* piores.

## Supervised Learning

Uma vez que o *dataset* apresenta cerca de 99% de linhas com *missing values* não foi considerada a remoção de todas estas linhas. Optou-se então por testar a sua substituição utilizando diferentes métodos (substituição por zero, média e mediana) e para cada classificador foi escolhido o método que obteve melhores resultados. Decidiu-se, no entanto, remover as linhas e colunas que apresentassem mais de 90% de *missing values,* já que estas não apresentam informação suficiente para serem relevantes e acabam por distorcer os dados.

Após a análise do desvio padrão de cada um dos atributos eliminou-se a coluna *cd\_000*, pois apresentava um desvio padrão de 0 (tomava sempre o mesmo valor), não acrescentado informação pertinente para o contexto do problema. Com o objetivo de reduzir a dimensionalidade dos atributos foi feita ainda uma análise utilizando o *Coeficiente de Correlação de Pearson* e descobriram-se 4 grupos de atributos que apresentavam uma correlação linear entre si. Foi então utilizado o *PCA* para reduzir cada um destes grupos em apenas um.

Para a execução do classificador *KNN* procedeu-se à normalização dos atributos com vista a reduzir o impacto das diferentes escalas de atributos sobre a distância calculada para os *vizinhos*.

Foi testada a remoção de *outliers* utilizando o método de *IQR* e verificou-se que os resultados da classificação pioravam. Após uma reflexão acerca dos resultados obtidos aquando da remoção dos *outliers,* chegou-se à conclusão que os mesmos são bastante importantes devido à natureza do problema. Uma vez que se pretende analisar a presença de avarias, os atributos que apresentem valores “fora do normal” (*outliers)* são essenciais para realizar tal previsão. Também não foram removidos valores potencialmente incorretos, pois todos os atributos estão anonimizados e é impossível saber quais os valores que cada atributo poderia tomar.

Por fim, devido ao elevado desequilíbrio entre as instâncias da classe positiva (1.000) e da classe negativa (59.000), procedeu-se ao balanceamento de ambas as classes utilizando diferentes métodos, de modo a evitar que os classificadores ficassem *biased* para a classe negativa, devido à elevada discrepância do número de instâncias de cada classe.

## Quality Assessment of Digital Colposcopies

## Unsupervised Learning

## Association Rules

À semelhança do primeiro problema, transformou-se o *dataset* num *dataset* transaccional usando a discretização por intervalos de *equal-width* e *equal-frequency*. Fez-se também *feature selection* para colmatar problemas de computação ao aplicar o algoritmo APRIORI com um número elevado de atributos.

## Clustering

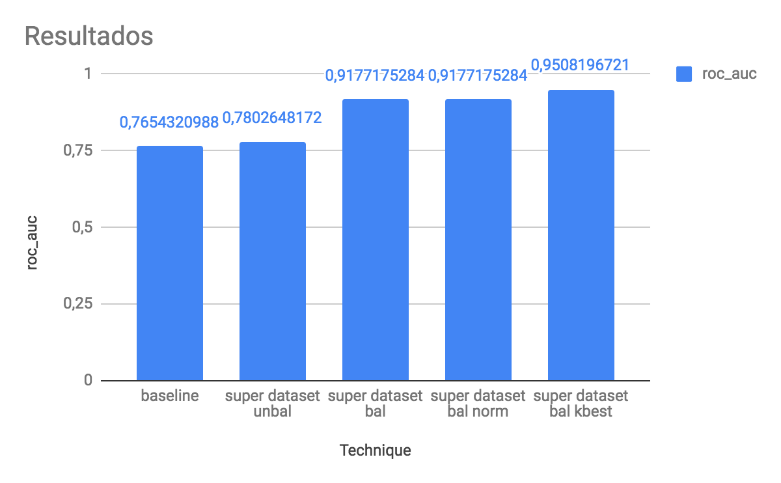
## Tal como no primeiro problema, fez-se a normalização dos dados para evitar influências das diferentes escalas entre atributos nos resultados obtidos pelo algoritmo K-Means.

## Supervised Learning

O *dataset* deste problema não apresenta *missing values,* pelo que não foi necessário efetuar nenhuma transformação nesse sentido.

Ao analisar os *datasets*, verificou-se que em situação de empate entre os vários peritos, o consenso a que se chegaria era sempre de que a imagem tinha boa qualidade. Para combater um consenso possivelmente errado, analisou-se para cada um dos peritos a percentagem de concordância com o consenso geral (excluindo os casos de empate) e verificou-se que um dos peritos apresenta uma concordância menor relativamente aos restantes. Decidiu-se então excluir a votação efetuado por este perito e recalculou-se o consenso utilizando apenas os restantes 5 peritos, evitando assim qualquer caso de empate.

Decidiu-se juntar os diferentes *datasets* num único e verificou-se que se obtinham melhores resultados relativamente aos obtidos nos *datasets* individuais. Tendo em conta o contexto do problema, cada *dataset* é relativo a uma técnica de análise de imagem diferente, pelo que foi necessário adicionar dois novos atributos binários (denominados *schiller* e *hinselmann*) que permitem descriminar a que técnica cada instância pertence.

Para remoção dos *outliers* testaram-se os métodos *IQR* e *z-score,* mas verificou-se que o *z-score* remove 25% dos dados para *z<3* e o *IQR* remove 80%. Tendo em conta o tamanho do *dataset* (287 instâncias) e a elevada percentagem de dados removidos, optou-se por descartar esta transformação.

Tal como no primeiro problema, é-nos apresentado um *dataset* *unbalanced* pelo que se recorreu ao seu balanceamento utilizando *SMOTE 50-50*.

Para diminuir a dimensionalidade dos atributos com vista a obter melhores resultados, efetuou-se *feature selection* utilizando o método *ANOVA* onde se escolheu as 10 melhores *features* para explicar o atributo *target.*

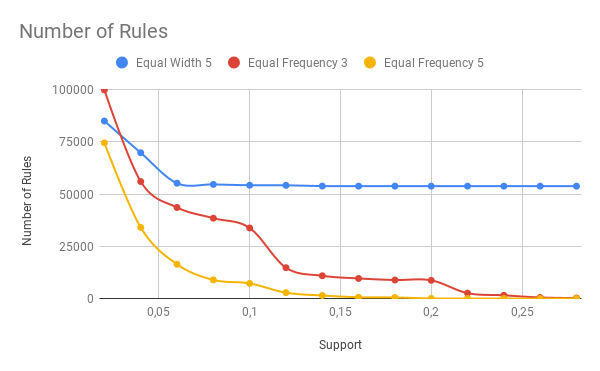
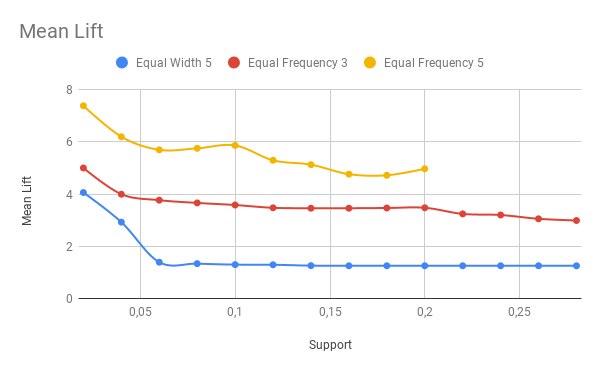
# Exploration

## APS Failure at Scania Trucks

## Unsupervised Learning

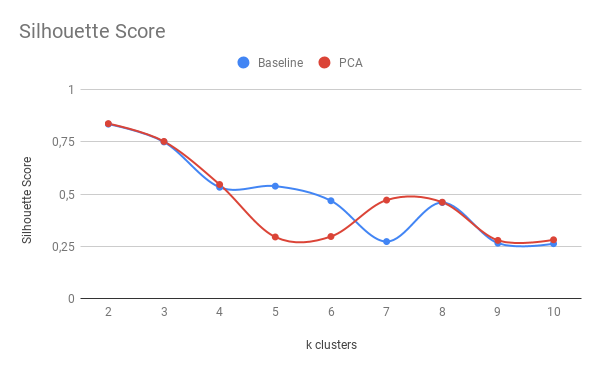
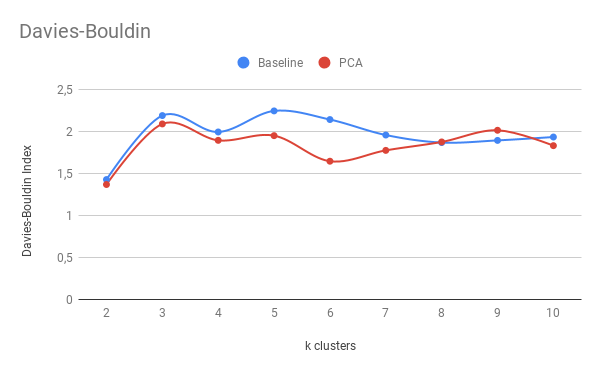
* **Association Rules**

Utilizou-se o algoritmo APRIORI para encontrar *frequent item sets*. Manteve-se uma confiança de 90% e variou-se o suporte mínimo entre 2 e 30%. Para discretizar os dados foram considerados intervalos de *equal-width* de 5 *bins* e intervalos de *equal-frequency* de 3 e 5 *bins.*

Como se pode observar no gráfico, o aumento do suporte mínimo leva a uma diminuição do número de regras geradas e do lift médio das regras obtidas. Podemos também constatar que, as regras com maior qualidade são apresentadas para a discretização por intervalos de frequência, tendo sido até um suporte de 20% a com 5 *bins* a que obteve um lift médio maior mantendo ainda um número de regras não muito elevado. A partir dos 20% a discretização por intervalos de frequencia de 5 *bins* deixou de criar regras tendo a de 3 *bins* continuado a criar regras.

* **Clustering**

Utilizou-se o algoritmo K-Means e variou-se o número de *clusters* (k) entre 2 e 10. De seguida, avaliou-se os *clusters* criados usando o índice de *Davies-Bouldin* e o *Silhouette Coefficient*.

Como se pode ver a partir dos gráficos, para um valor de k=2 o *Silhouette Score* é máximo e o índice de *Davies-Bouldin* mínimo, o que indica que o melhor *clustering* é obtido para este número de clusters. É de notar que este valor corresponde ao número de classes (‘pos’ e ‘neg’) existentes no problema. Assim, usou-se o *Adjusted Rand Score* como alternativa supervisionada (usando a *ground truth)* às métricas apresentadas anteriormente para avaliar os *clusters* criados. No entanto, o valor obtido foi de apenas ~0.51, pelo que os *clusters* obtidos, apesar de serem bons segundo as primeiras métricas, não correspondem à *ground truth*.

## Supervised Learning

## Methods and Parametrization

Para avaliar a *performance* de cada um dos classificadores utilizou-se uma divisão treino/teste de 70-30%, que corresponde aproximadamente à divisão original dos ficheiros do *dataset*. Considerou-se esta divisão a mais adequada devido ao número de instâncias do *dataset*, que se encontra na ordem das dezenas de milhar. Uma vez que o *dataset* não apresenta as classes uniformemente distribuídas (*dataset unbalanced*), a *accuracy* não é a melhor métrica para avaliar os resultados obtidos pelos modelos. Assim, decidiu-se usar a *ROC AUC* por ser uma métrica que melhor avalia a performance em *datasets unbalanced*, juntamente com a métrica original, que nos permitirá comparar os modelos especificamente para o problema em questão.

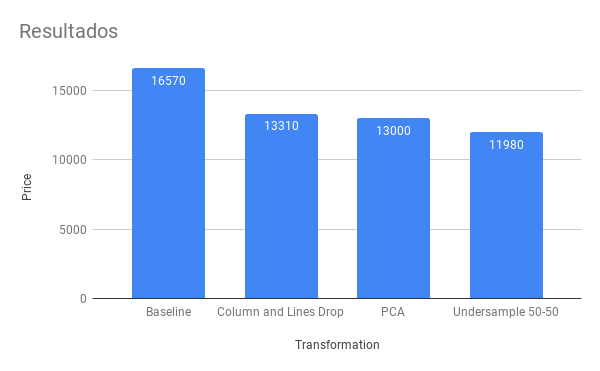
Naive Bayes: Foi escolhida a versão *Gaussiana* do *Naive Bayes*.

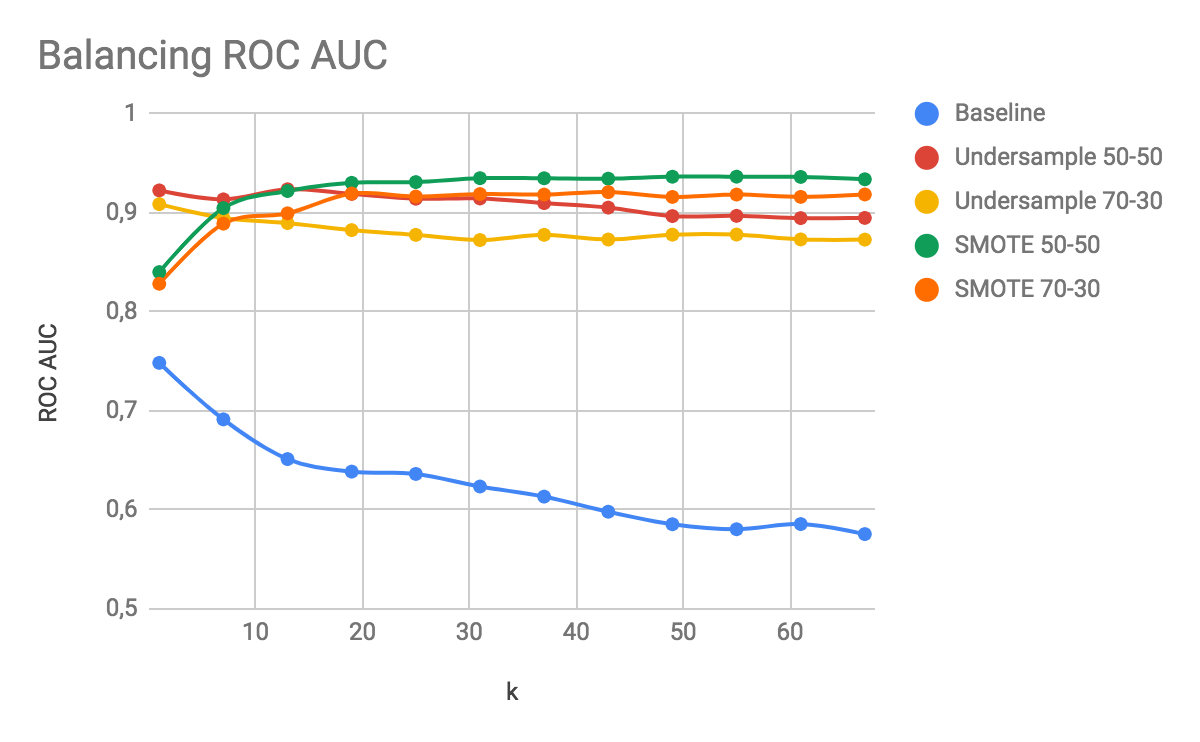
KNN: Variou-se o valor de *k-vizinhos* entre 1 e 67, com incrementos de 6 para evitar números pares que poderiam levar a empates durante a classificação. Utilizou-se a distância euclidiana para fazer o cálculo de distâncias entre vizinhos.

Decision Trees: Utilizou-se o algoritmo *CART* e variou-se o parâmetro *min\_samples\_split* entre 5 e 1000*,* que corresponde ao número mínimo de instâncias para dividir um nó da árvore.

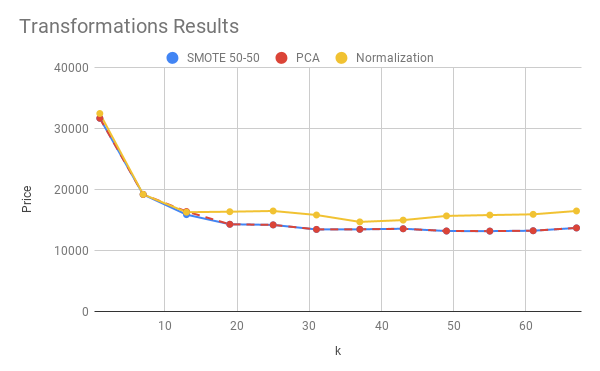
Random Forests: Tal como na *Decision Tree*, variou-se o *min\_samples\_split* entre os mesmos valores e ainda o parâmetro *n\_estimators* entre 10 e 1000, que corresponde ao número de árvores a considerar pelo algoritmo.

## Results

Naïve Bayes: Comparou-se os resultados obtidos usando várias técnicas de pré-processamento. Definiu-se uma *baseline* usando a técnica de tratamento a *missing values* que melhores resultados obteve para este classificador. A *baseline* obtida foi das melhores entre os classificadores, o que demonstra que o Naïve Bayes não é tão *biased* quanto outros classificadores quando apresentado com dados *unbalanced.* Também apresentou melhorias expectáveis ao remover *features* correlacionadas, já que este assume a independência das suas variáveis. Houve uma pequena melhoria ao fazer *balancing* dos dados tendo sido a técnica com melhores resultados o *undersampling* aleatório com um rácio 50-50% entre classes.

KNN: Devido ao acentuado *unbalancement* do *dataset*, obtiveram-se resultados bastante negativos usando este algoritmo antes de aplicar *balancing*. Isto deve-se à grande densidade de instâncias da classe negativa, pelo que existem por norma mais vizinhos dessa classe, fazendo com que o classificador se torne muito *biased* em relação a esta.

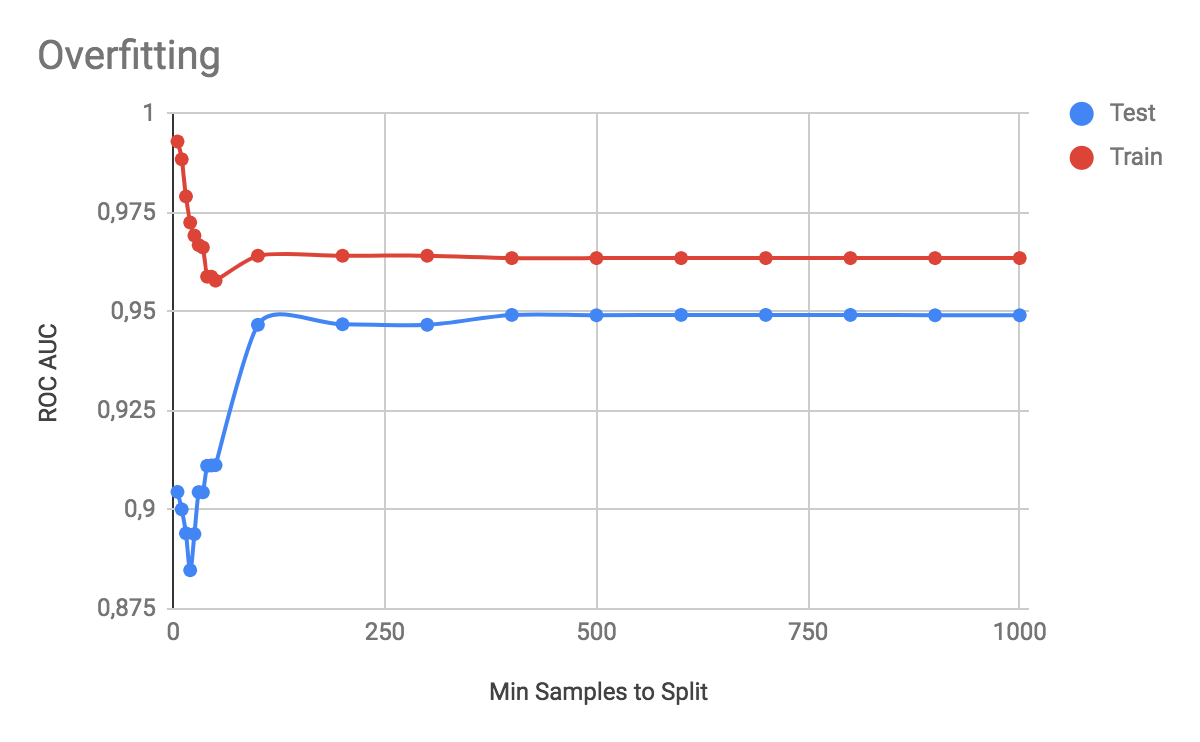
Aplicaram-se várias técnicas para fazer o balanceamento dos dados e como se pode concluir pelo gráfico, os melhores resultados foram obtidos utilizando a técnica de balanceamento *SMOTE 50-50*. É de salientar que para os dados não balanceados, o melhor resultado é obtido para *k=*1, piorando sucessivamente com o aumento do valor de *k,* corroborando a hipótese acima referida.

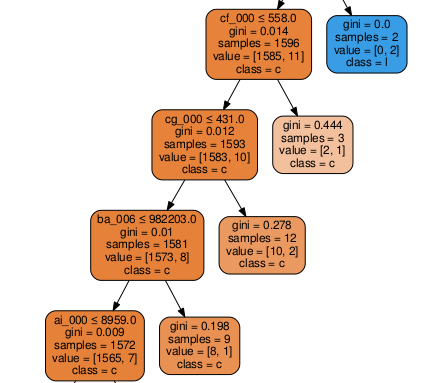


Após o balanceamento dos dados testou-se o uso do PCA, para reduzir o espaço dimensional das *features*, e a normalização dos dados, para reduzir o efeito das diferentes escalas entre atributos no cálculo das distâncias entre vizinhos, visto a melhorar os resultados obtidos por este classificador.

Como podemos observar no gráfico, e ao contrário do esperado, estas transformações não obtiveram melhores resultados para o contexto do problema, tendo o melhor preço sido atingido usando apenas o balanceamento de dados. Em particular, a normalização piorou os resultados, o que pode querer indicar a existência de atributos mais importantes que outros no *dataset* pelo que o efeito da sua escala acaba por ser positiva na determinação das classes.

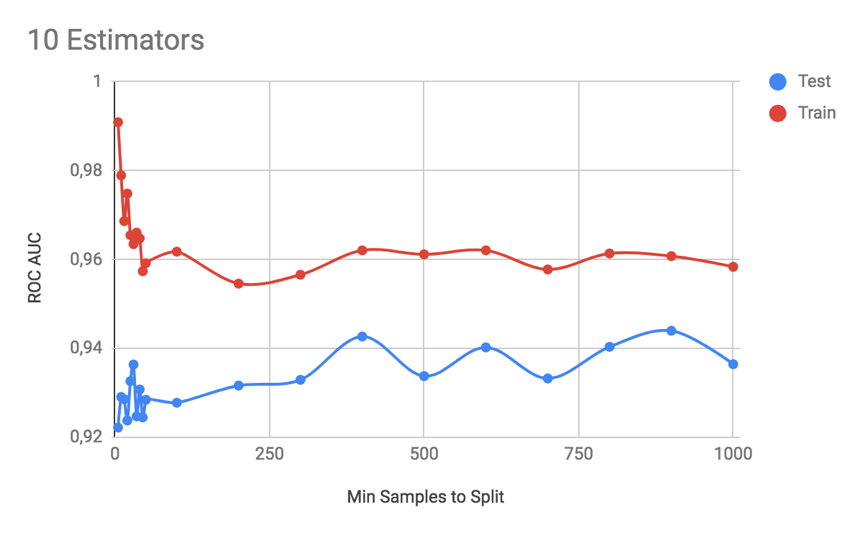
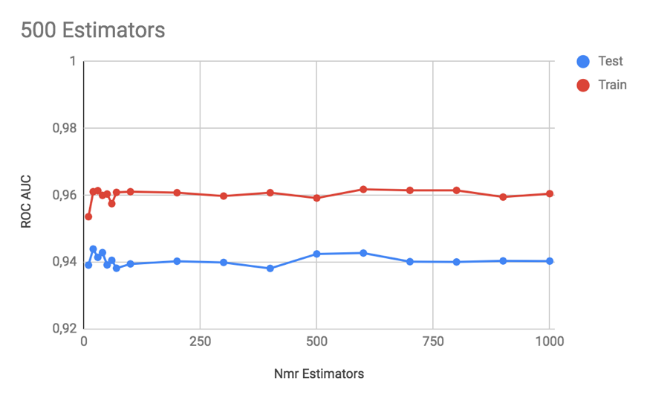
Decision Tree: Em semelhança aos classificadores anteriores, foi feita uma análise aos resultados obtidos utilizando os diferentes métodos para substituir *missing values* e para balancear os dados. Não se realizou *feature selection* já que as *Decision Trees* particionam por si só os dados utilizando os atributos que determinam como sendo mais relevantes. Verificou-se que a substituição dos *missing values* pela mediana, em conjunto com a heurística de remoção de linhas e colunas tendo em conta o número de *missing* values, e o uso de *undersampling* aleatório com um rácio 70-30% obtiveram os melhores resultados.

Como se pode verificar pelo gráfico, o modelo inicialmente está em bastante *overfitting*. Ao aumentar o número de instâncias para subdividir cada nó da árvore, verifica-se que os valores da ROC AUC para o conjunto de treino pioram enquanto que os do conjunto de teste melhoram. Tal demostra que o classificador generaliza melhor o problema ao aumentar este parâmetro, já que não são criados nós apenas para casos específicos, não tornando assim o modelo tão próximo do *dataset* de treino, não ocorrendo tanto *overfitting*. A partir de um valor de min\_samples\_split 400 e até 1000, os valores da ROC AUC não se alteram pelo que não existe uma perda de informação ao aumentar este valor.



Analisou-se a árvore gerada para *min\_samples\_split=400* e verificou-se que a partir de uma certa profundidade a classe que o modelo vai atribuir é sempre a mesma, pelo que se poderia aplicar *pruning* à arvore, limitando a sua profundidade e resultando num modelo mais eficiente computacionalmente sem alteração aos resultados que produz.

Random Forest: Após uma breve análise verificou-se que o *n\_estimators* tinha uma grande influência na variação dos resultados. Para comprovar a nossa teoria realizou-se uma análise detalhada, em que, como se pode concluir na primeira imagem, com um número pequeno de *estimators* (10) existe uma volatilidade nos valores obtidos para a ROC AUC*.* Por oposição, ao aumentar o número de *estimators* para 500, como se verifica no segundo gráfico, os resultados obtidos são menos voláteis. Concluímos assim, que com um número maior de estimators conseguimos construir uma Random Forest menos *biased* reduzindo o *overfitting,* e melhorando assim os resultados.



## Quality Assessment of Digital Colposcopies

## Unsupervised Learning

## Methods and Parametrization

Vunc sed eede. Praesent vitae lectus. Praesent neque justo, vehicula eget, interdum id, facilisis et, nibh adsumes.  
Phasellus at purus et libero lacinia dictum. Fusce aliquet. Nulla eu ante placerat leo semper dictum.

## Results

Vunc sed eede. Praesent vitae lectus. Praesent neque justo, vehicula eget, interdum id, facilisis et, nibh adsumes.  
Phasellus at purus et libero lacinia dictum. Fusce aliquet. Nulla eu ante placerat leo semper dictum.

## Supervised Learning

## Methods and Parametrization

Para o problema em questão, que apresenta um número de instâncias na ordem das centenas, utilizou-se *cross validation* em 10 *folds*, visto que é a melhor estratégia de treino para *datasets* desta grandeza. À semelhança do primeiro problema, utilizou-se a *ROC AUC* para avaliar a *performance* dos classificadores uma vez que o *dataset* não é balanceado.

Naive Bayes: Escolheu-se o Bernoulli Naive Bayes.

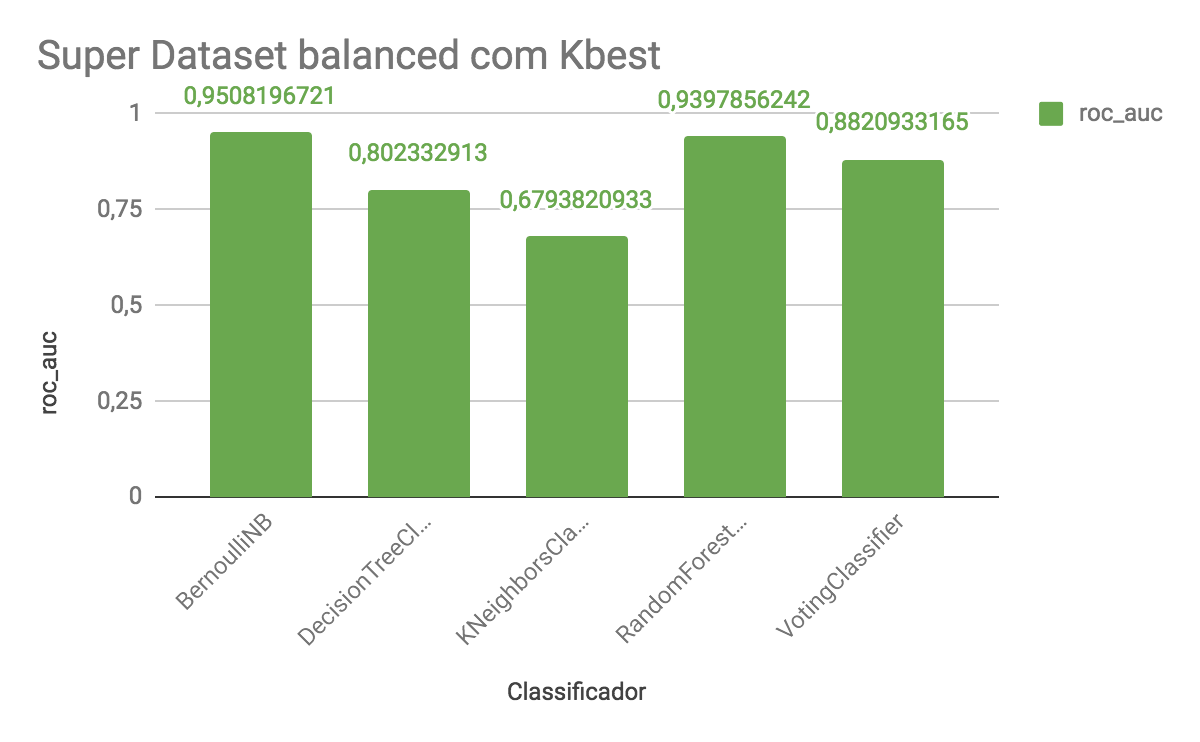
KNN: Variou-se os *k-vizinhos* entre 1 e 99, com incrementos de 2 para evitar empates.

Decision Trees: Variou-se *min\_samples\_split* entre 2 e 50.

Random Forests: Variou-se *min\_samples\_split* entre 2 a 100 e *n\_estimators* entre 2 e 50.

## Results

Vunc sed eede. Praesent vitae lectus. Praesent neque justo, vehicula eget, interdum id, facilisis et, nibh adsumes.  
Phasellus at purus et libero lacinia dictum. Fusce aliquet. Nulla eu ante placerat leo semper dictum.



# Critical Analysis

Vunc sed eede. Praesent vitae lectus. Praesent neque justo, vehicula eget, interdum id, facilisis et, nibh adsumes.  
Phasellus at purus et libero lacinia dictum. Fusce aliquet. Nulla eu ante placerat leo semper dictum.

# Conclusions

Vunc sed eede. Praesent vitae lectus. Praesent neque justo, vehicula eget, interdum id, facilisis et, nibh adsumes.  
Phasellus at purus et libero lacinia dictum. Fusce aliquet. Nulla eu ante placerat leo semper dictum.