



## Trabalho 3

Júlia de Azevedo - 2312392

Robbie Carvalho - 2311833

Theo Canuto - 2311293

INF1771 - Grupo Tangerine

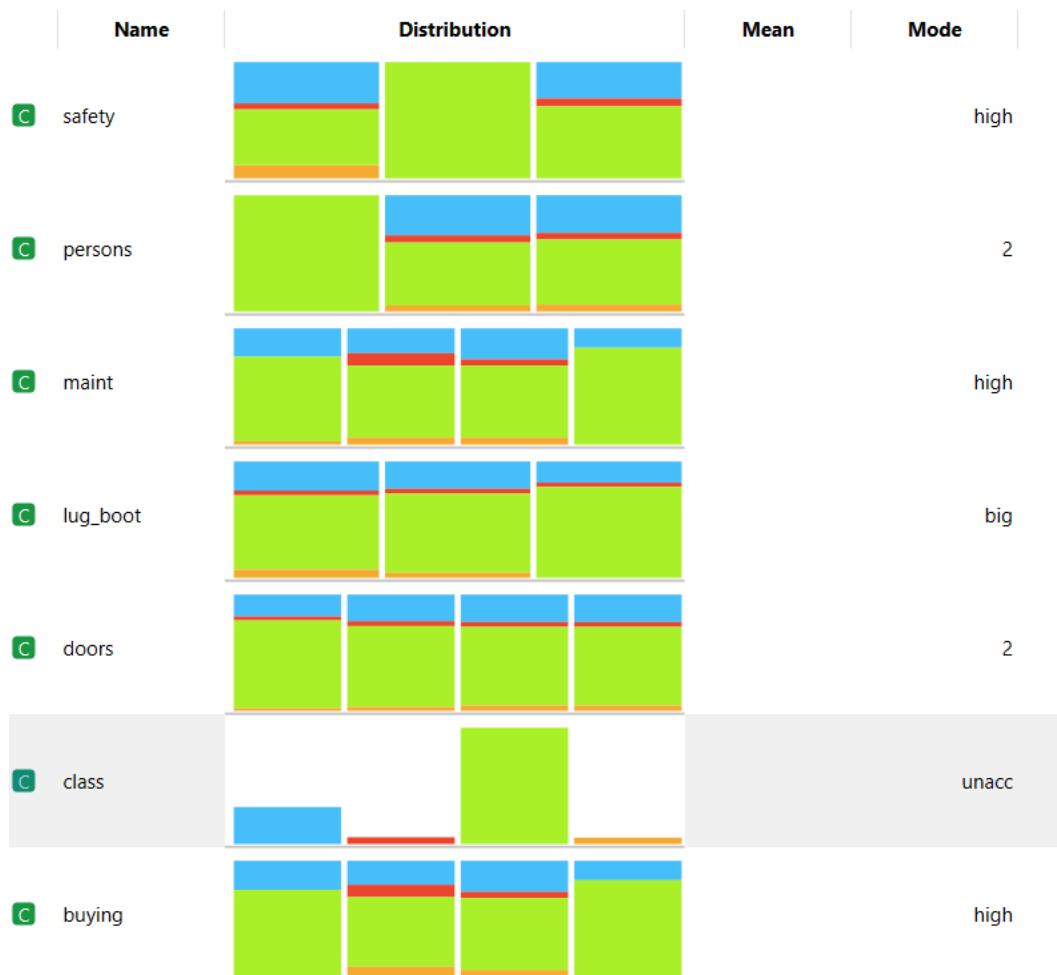
Prof. Augusto Baffa

*02 de Julho de 2025, Rio de Janeiro*

## 1. Dataset de avaliação de carros

### a. Análise Exploratória de Dados:

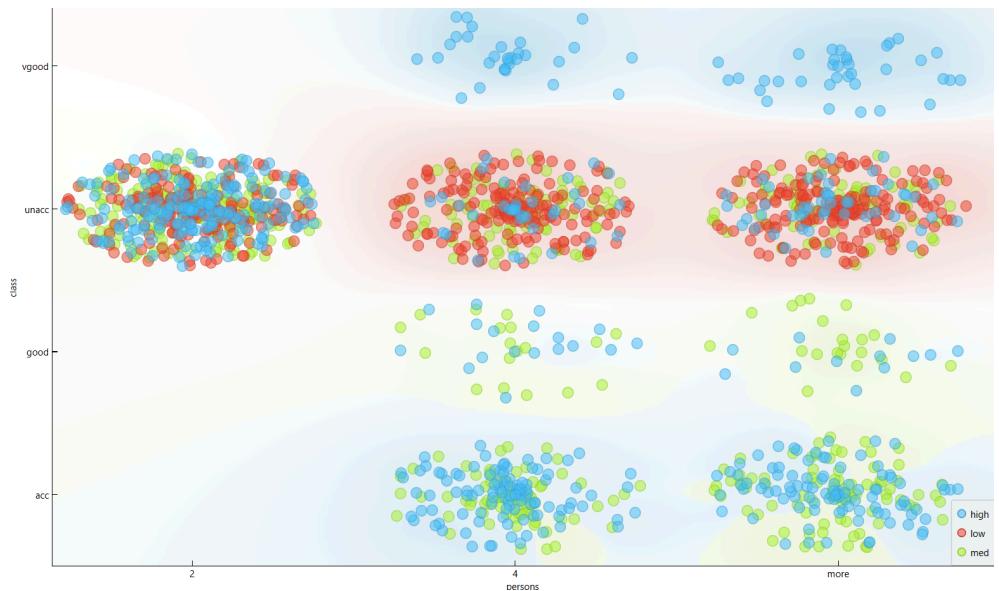
Para compreender melhor o dataset, iniciamos a visualização com Feature Statistics.



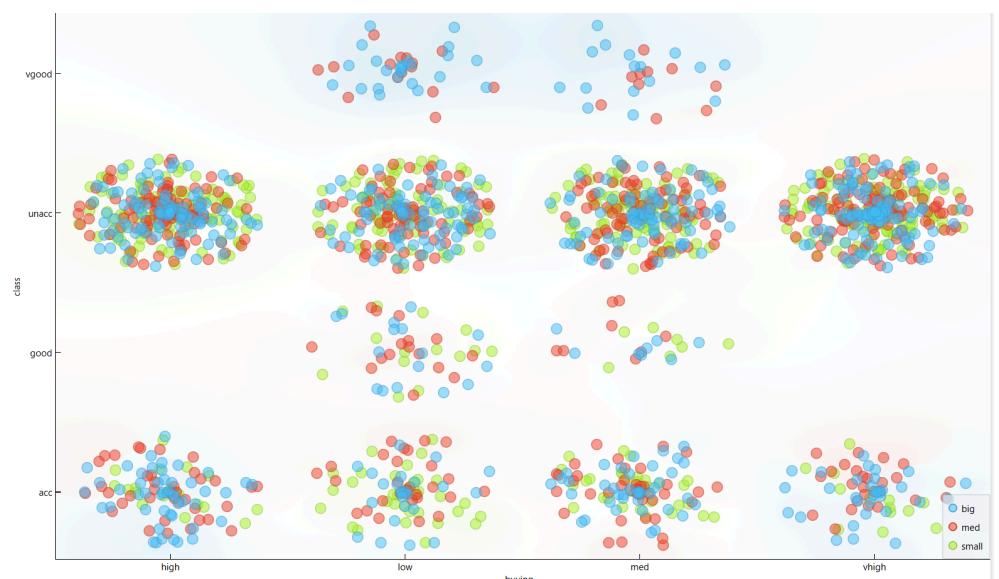
A partir disso, descobrimos muitas coisas. Dentre elas, as variáveis mais comuns no dataset através da moda fornecida no Feature Statistics. Uma das conclusões obtidas é que mais da metade dos carros do dataset está na classe "inaceitável".

Após já ter algum entendimento do dataset, utilizamos o scatter plot para entender a relação entre a classe e mais de uma feature simultaneamente.

Por exemplo, ambos "safety" e "persons", tinham algum de seus labels integralmente em inaceitável:



E, também, "buying" e "lug\_boot" - ambos tinham uma distribuição bastante restrita para carros "vgood" e "good":



Ao final dessa etapa de análises, decidimos manter os dados como categóricos, principalmente para termos a acurácia e outras métricas que já conhecíamos melhor (e.g., F1-Score, Precision).

### b. Seleção de Atributos

Ao longo das aulas, aprendemos dois algoritmos, o PCA e o MDS, que têm como objetivo reduzir a dimensionalidade dos dados de forma distinta: o PCA (Análise de Componentes Principais) identifica combinações lineares ortogonais dos atributos originais que capturam progressivamente a maior parte da variância, rotacionando rigidamente os eixos de um espaço  $p$ -dimensional para um subespaço  $k$ -dimensional não correlacionado e mais compacto; já o MDS (Escalonamento Multidimensional) parte de uma matriz de dissimilaridades ou semelhanças entre pares de objetos e projeta esses pontos em um espaço de baixa dimensão de modo que as distâncias refletem com fidelidade as dissimilaridades

originais, oferecendo ainda melhor desempenho quando se trabalha com medidas não métricas.

Depois da análise geral do dataset, aplicamos o algoritmo MDS para entender mais a fundo as conexões que tentamos criar no scatter plot.

Gráfico Class

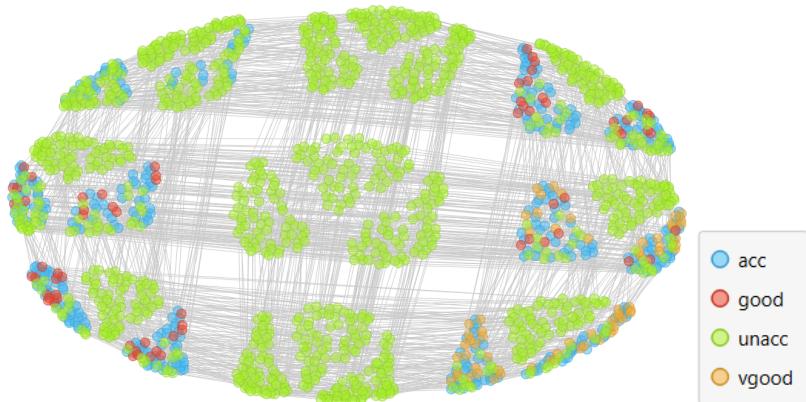


Gráfico Persons

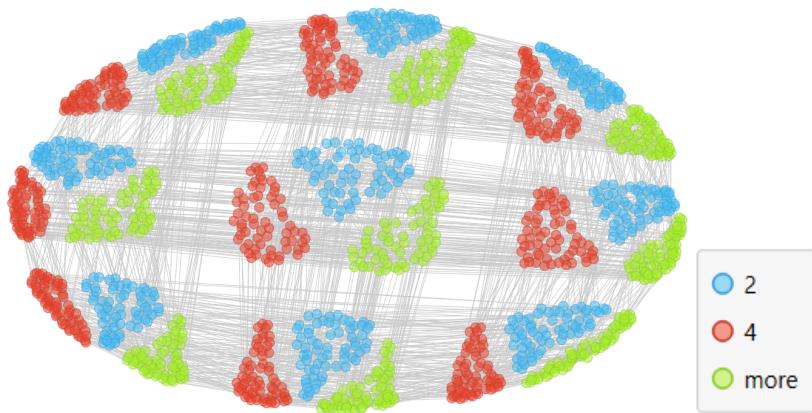


Gráfico Lug\_boost

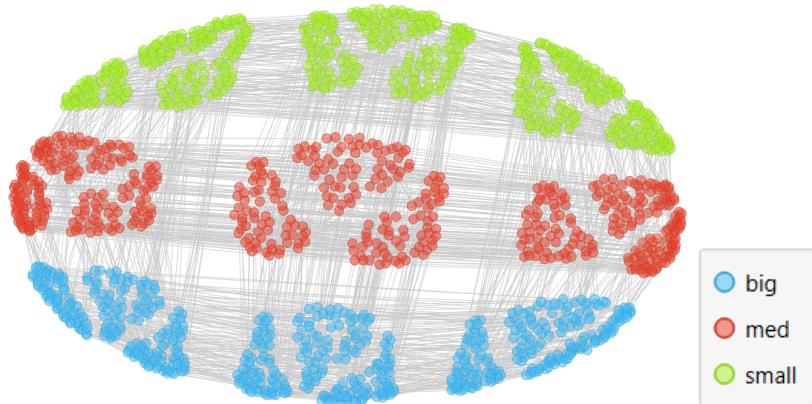


Gráfico Doors

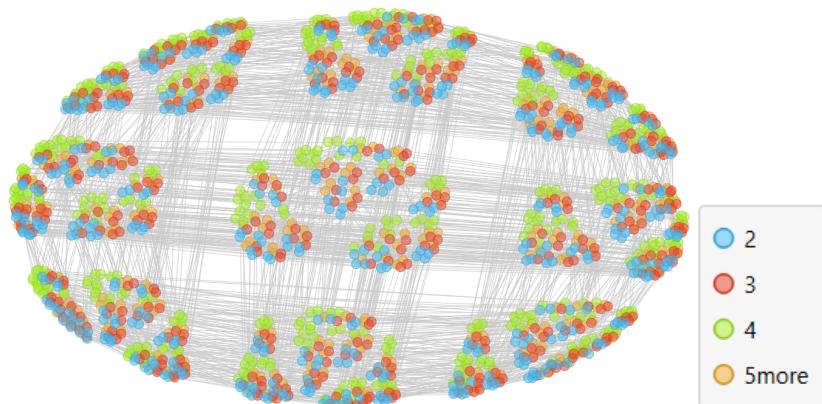


Gráfico Safety

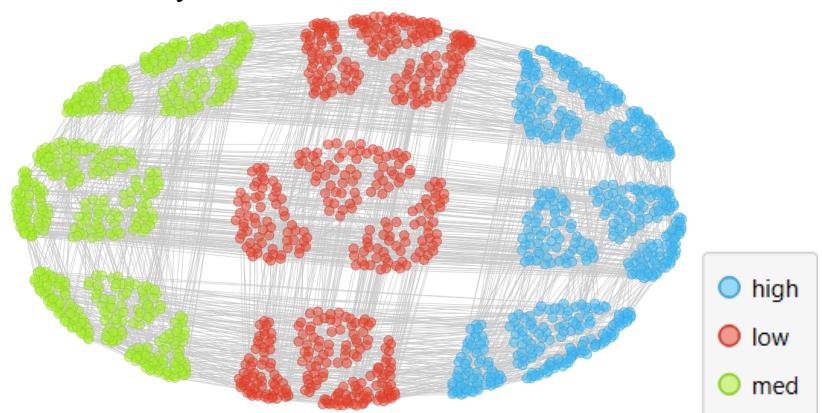


Gráfico Buying

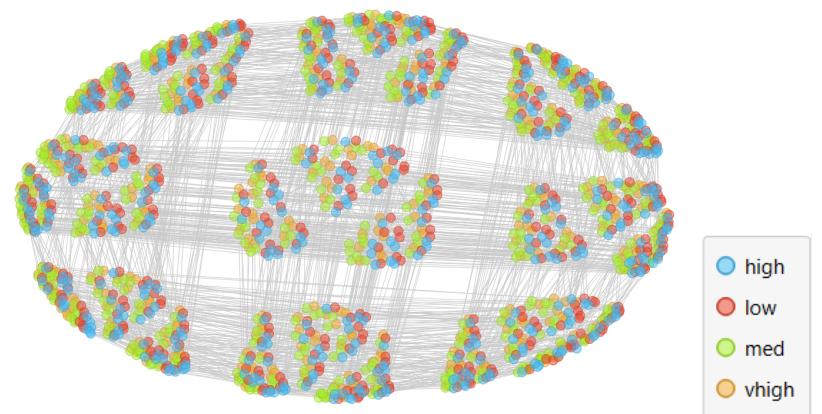
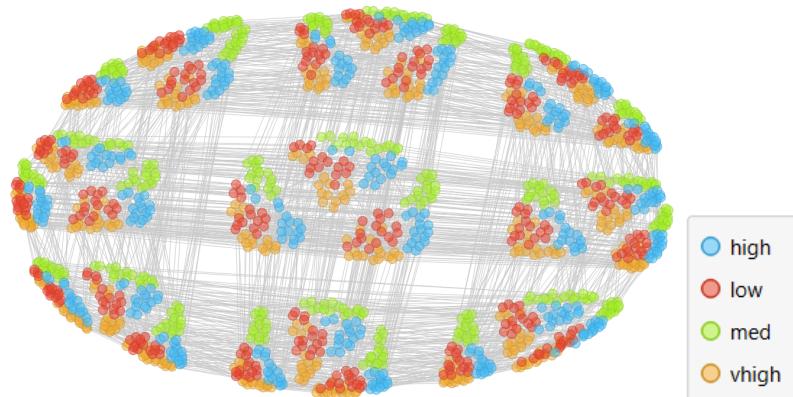
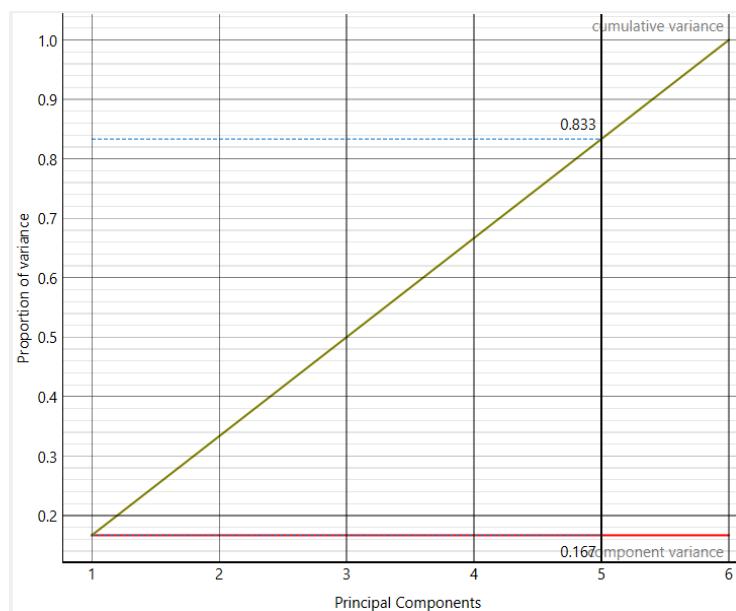


Gráfico Maint



Em seguida, aplicamos o algoritmo PCA para obter melhores maneiras de agrupar os componentes. Nesse algoritmo, percebemos que acima de 5 componentes teríamos pelo menos 80% de variância explicada.

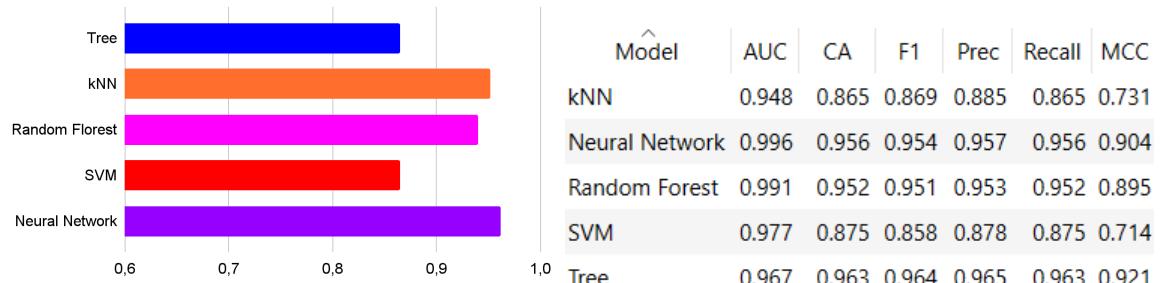
Nº de componentes	<5	5	6
Variância explicada	<80%	83%	100%



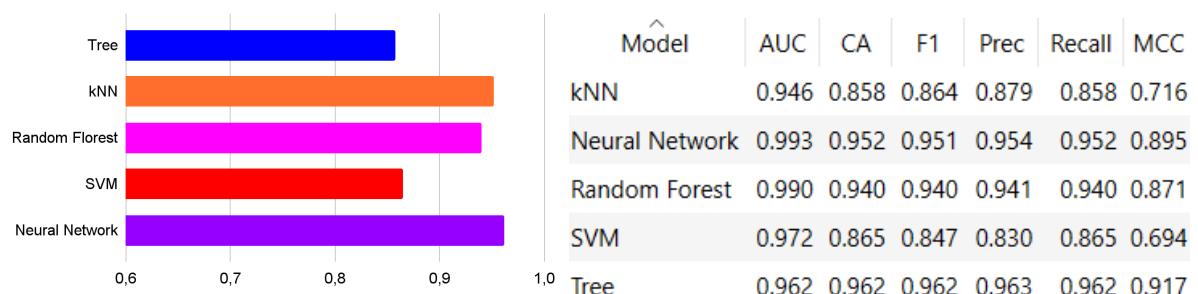
### c. Geração do Modelo

Para efeitos comparativos, primeiro testamos as previsões sem utilizar o PCA:

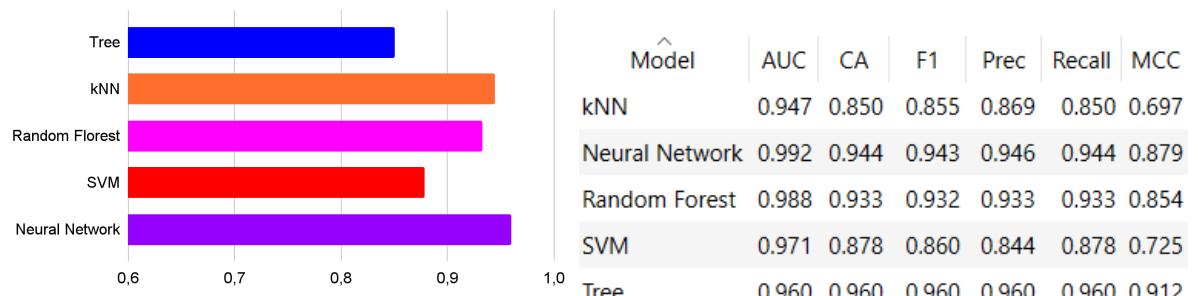
## Test on test data - Acurácia



## Test on cross validation (20 folds) - Acurácia

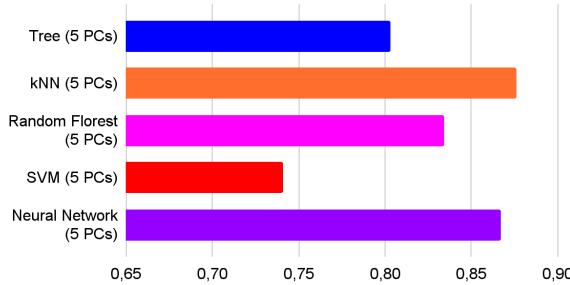


Test on cross validation (10 folds) (padrão) - Acurácia



E, depois, utilizamos o PCA para fazer o teste com 5 e 6 componentes (83% e 99% de variância explicada, respectivamente). Como exibiu a melhor acurácia no passo anterior, mantivemos a estratégia de teste em test data:

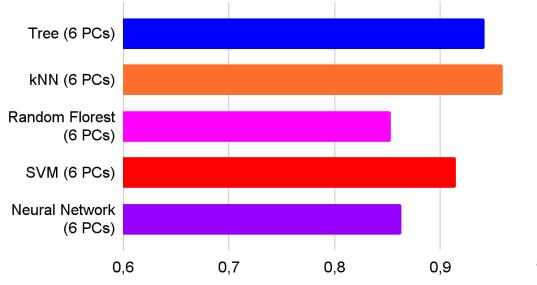
Acurácia



$\hat{M}$ odel

$\hat{M}$ odel	AUC	CA	F1	Prec	Recall	MCC
kNN (5 PCs)	0.902	0.803	0.805	0.809	0.803	0.576
Neural Network (5 PCs)	0.945	0.876	0.875	0.874	0.876	0.726
Random Forest (5 PCs)	0.938	0.834	0.825	0.825	0.834	0.626
SVM (5 PCs)	0.853	0.741	0.724	0.741	0.741	0.387
Tree (5 PCs)	0.809	0.867	0.866	0.865	0.867	0.707

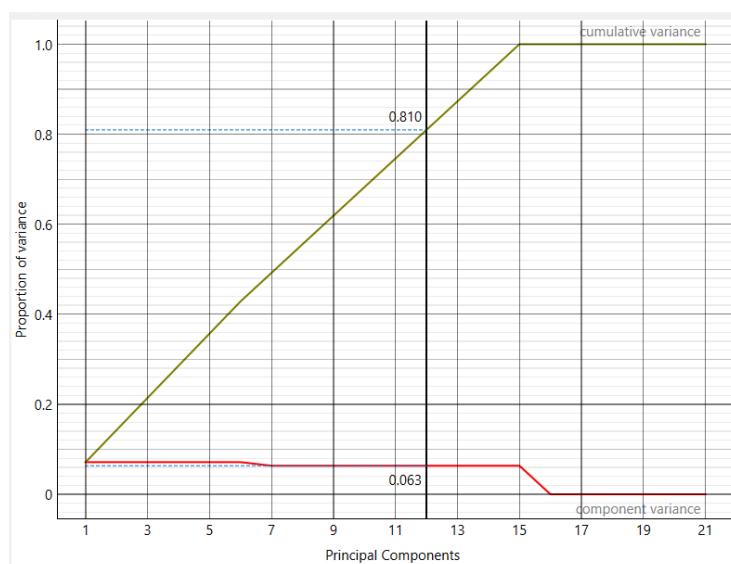
Acurácia



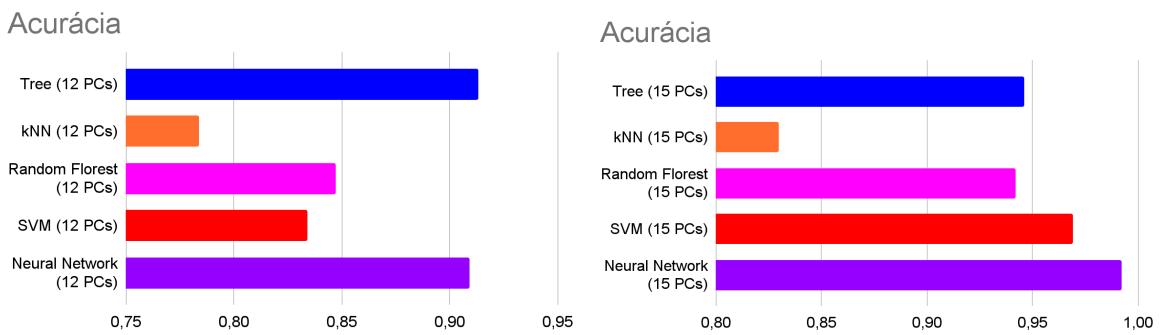
$\hat{M}$ odel

$\hat{M}$ odel	AUC	CA	F1	Prec	Recall	MCC
kNN (6 PCs)	0.976	0.942	0.941	0.942	0.942	0.872
Neural Network (6 PCs)	0.997	0.959	0.958	0.961	0.959	0.913
Random Forest (6 PCs)	0.949	0.853	0.849	0.849	0.853	0.677
SVM (6 PCs)	0.987	0.915	0.898	0.882	0.915	0.812
Tree (6 PCs)	0.805	0.863	0.864	0.868	0.863	0.701

OBS: O que acontece quando fazemos sem continuize?



Nº de componentes	<12	12	13	14	15	16-21
Variância explicada	<80%	80%	87%	93%	99%	100%



O melhor de todos os modelos em todos os testes foi com PCA, Rede Neural com 6 PCs: 99.7%.

### **Resultados e aprendizados:**

A análise qualitativa via MDS mostrou que os pontos coloridos segundo o atributo safety formam agrupamentos muito bem definidos para as categorias high, med e low, evidenciando que esse é o fator mais discriminante na distinção entre “vgood” e “good” versus “acc” e “unacc”. Já buying e maintenance também apresentam clusters coerentes, indicando que custo de aquisição e de manutenção exercem forte influência na avaliação dos carros, ao passo que variáveis como persons, lug\_boot e doors exibem maior dispersão e sobreposição, sinalizando impacto relativamente menor na classificação.

Complementando essa visão, a aplicação de PCA ao mesmo conjunto revelou que 5 componentes principais explicam 83% da variância total. Os dois primeiros eixos concentram sobretudo as flutuações relativas a safety, buying e maint, permitindo reduzir substancialmente o número de dimensões sem perda significativa de informação, o que não só acelera o treinamento de modelos de classificação como também facilita visualizações bidimensionais que espelham a estrutura de dispersão observada nos gráficos de MDS.

Nos testes iniciais sem redução de dimensionalidade, a árvore de decisão se destacou, alcançando cerca de 96.3% de acurácia no conjunto de teste, seguida por Rede Neural e Random Forest (ambos com cerca de 95% de acurácia) e pelo k-NN e o SVM (acurácia de 86.5% e 87.5% respectivamente). Esse comportamento se manteve - variação na performance de no máximo 2% - na validação cruzada (tanto para 10 quanto para 20 folds), indicando alta estabilidade dos classificadores quando todos os atributos originais estão disponíveis.

Ao incorporar o PCA, vemos que com apenas 5 componentes (83% da variância explicada) houve uma perda moderada de desempenho: a Rede Neural passou a ter uma acurácia de 87.6%, o k-NN caiu para 80.3% e Random Forest para 83.4%, evidenciando que parte da informação discriminativa ficou retida nos componentes excluídos. Já com 6 PCs, a Rede Neural voltou a sua acurácia de cerca de 95%, o k-NN a 94.2% e a Random Forest a 85.3%. Esses achados sugerem que reduzir o espaço de atributos de 6 para 5 componentes impactou negativamente a predição.

A aplicabilidade dos dados obtidos no modelo de avaliação de carros pode ser resumida da seguinte forma:

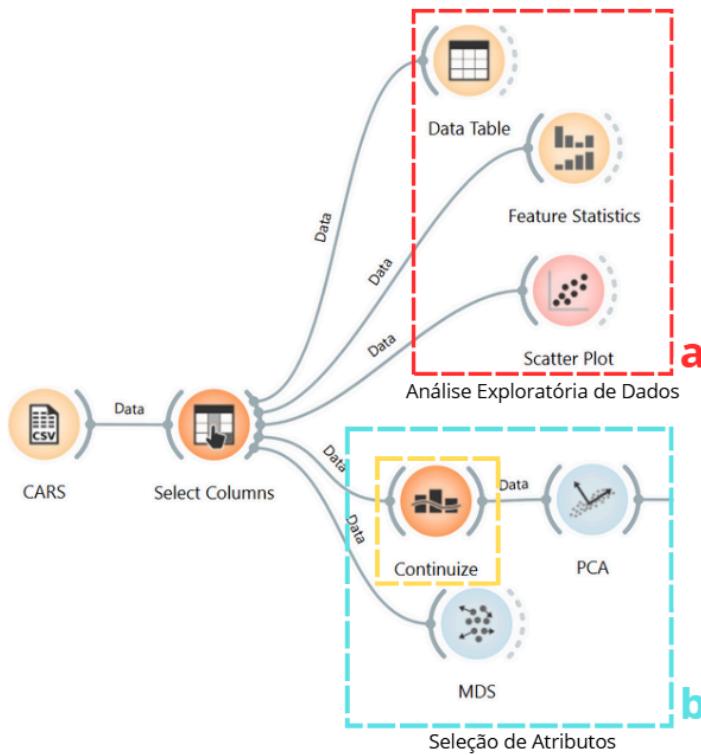
- Classificação Automática de Carros: O modelo pode ser usado para classificar carros em diferentes condições, agilizando processos de vendas e estimativas de preço em plataformas de compra e venda.
- Segmentação de Mercado e Marketing: Os dados ajudam a identificar preferências dos consumidores, permitindo a personalização de ofertas e campanhas direcionadas, além de informar o desenvolvimento de novos produtos.
- Gestão de Frota: Empresas de aluguel ou frotas podem usar o modelo para avaliar rapidamente o estado dos veículos, otimizando custos de avaliação para necessidade de manutenção e substituição.
- Otimização de Vendas: O modelo pode alimentar sistemas de recomendação em plataformas de venda, aprimorando a experiência de compra ao sugerir veículos com base nas preferências do consumidor.
- Análise de Tendências de Mercado: Os dados ajudam a identificar tendências no mercado automotivo, auxiliando empresas a ajustar suas estratégias de estoque e vendas.

A partir da análise dos dados, técnicas como PCA e MDS foram aplicadas para identificar os atributos mais influentes na avaliação dos carros. O PCA, por exemplo, ajuda a reduzir a dimensionalidade dos dados e a concentrar a análise nas variáveis que mais contribuem para a variância, permitindo uma avaliação mais eficiente dos carros e facilitando a previsão de sua condição futura.

Essas informações podem ser utilizadas para treinar modelos de aprendizado de máquina, como redes neurais, SVM, e árvores de decisão, para prever com precisão a avaliação de carros com base em suas características. Esses modelos podem, por exemplo, classificar a condição de um carro, prever o preço futuro ou até sugerir a manutenção adequada. Além disso, essas informações podem ser aplicadas para ajustar estratégias de preço e estoque em empresas de venda de carros, otimizando a oferta de veículos conforme as preferências do mercado. Em plataformas de venda, também é possível usar essas informações para criar sistemas de recomendação personalizados, sugerindo carros com base nas características mais valorizadas pelos consumidores.

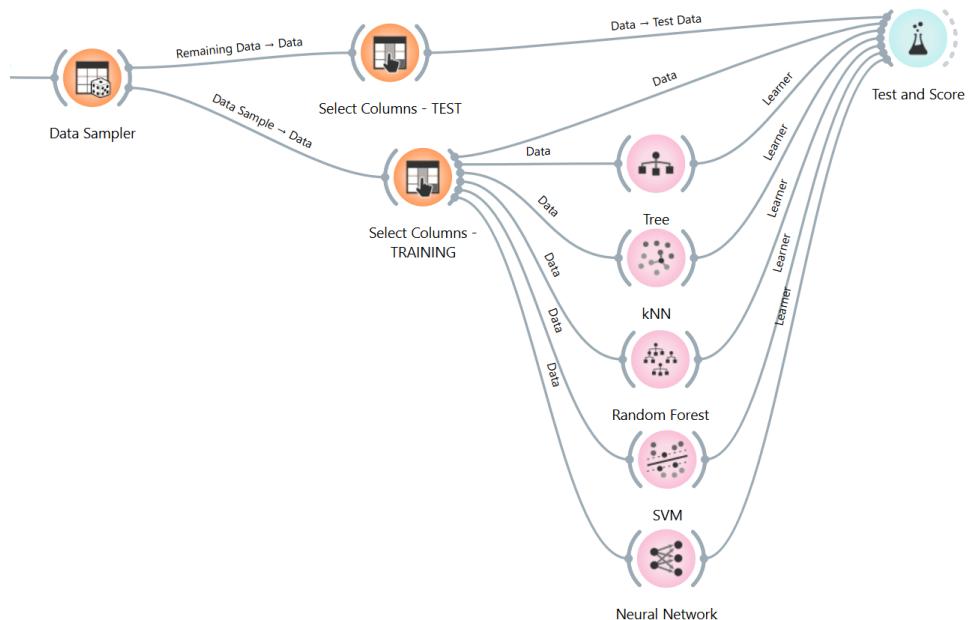
Em resumo, as informações extraídas dos dados podem ser usadas para otimizar decisões de vendas, gestão de frota e marketing, além de melhorar a experiência do consumidor e aumentar a precisão das previsões de avaliação de carros no futuro.

#### d. Carga de dados e separação de conjuntos



\*OBS.: para o teste com e sem continuize, apenas removemos a bolinha e conectamos select columns direto no PCA.

#### c) Geração do Modelo



Os fluxos vêm de uma única ramificação, depois de PCA. No entanto, o fluxo sem PCA passa pelo select columns e filtra para ter apenas os atributos do próprio dataset (persons, lug\_boot, doors, safety, buying, maint), enquanto os de PCA são ajustados no gráfico de PCA e têm seu próprio select columns para o conjunto PCx.

## 2. Dataset de Aprovação de Crédito

### a. Análise Exploratória de Dados:

Seguindo a estratégia do primeiro dataset analisado, partimos para uma análise de credit approval utilizando Feature Statistics.



A partir disso, algumas observações foram feitas. Ao contrário do dataset carros, temos dados faltando. Podemos perceber também que as classes estão muito mais balanceadas, com leve prevalência da classe “menos”.

### b. Seleção de atributos

Pela quantidade de atributos, decidimos por utilizar Rank ao invés de MDS neste dataset. Escolhemos analisar os atributos de acordo com a tabela. Utilizamos os métodos Gain ratio (recebe atributos categóricos, penaliza alta cardinalidade),  $\chi^2$  (recebe atributos categóricos, teste de independência), ReliefF (recebe atributos numéricos/categóricos, baseado em vizinhança) e FCBF (recebe atributos numéricos/categóricos, filtra redundâncias e relevância).

Essa combinação garante que nosso processo de seleção seja agnóstico ao formato dos dados, mantendo somente aqueles atributos que efetivamente carregam sinal preditivo, independentemente de serem numéricos ou categóricos.

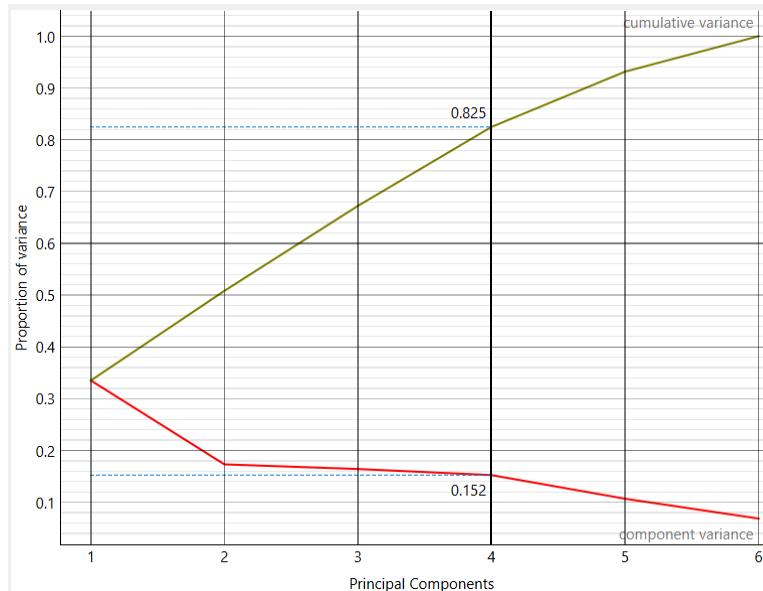
	#	Gain ratio	$\chi^2$	ReliefF	FCBF
1	C A9	2	0.426	170.746	0.356
2	C A10	2	0.159	82.966	0.080
3	N A11		0.132	278.523	0.004
4	N A15		0.064	70.772	0.004
5	N A8		0.057	85.136	0.018
6	C A4	3	0.036	3.810	0.026
7	C A5	3	0.036	34.763	0.026
8	C A6	14	0.031	37.367	0.110
9	C A7	9	0.028	0.001	0.123
10	N A14		0.022	12.028	0.012
11	C A13	3	0.020	12.339	0.008
12	N A3		0.020	22.213	0.010
13	N A2		0.011	10.615	0.022
14	C A12	2	0.001	0.374	0.022
15	C A1	2	0.000	0.125	0.016

Desta maneira, selecionamos 6 dos atributos que demonstraram ter maior relevância para o dataset e testamos as diferenças entre o teste com todos os atributos vs. com o nosso filtro observado pela tabela anterior. Esses atributos foram: A10, A9, A5, A6, A7 de categóricos e A11 de numérico.

Em seguida, aplicamos o algoritmo PCA para obter melhores maneiras de agrupar os componentes. A tabela abaixo ilustra as possibilidades testadas e os resultados obtidos.

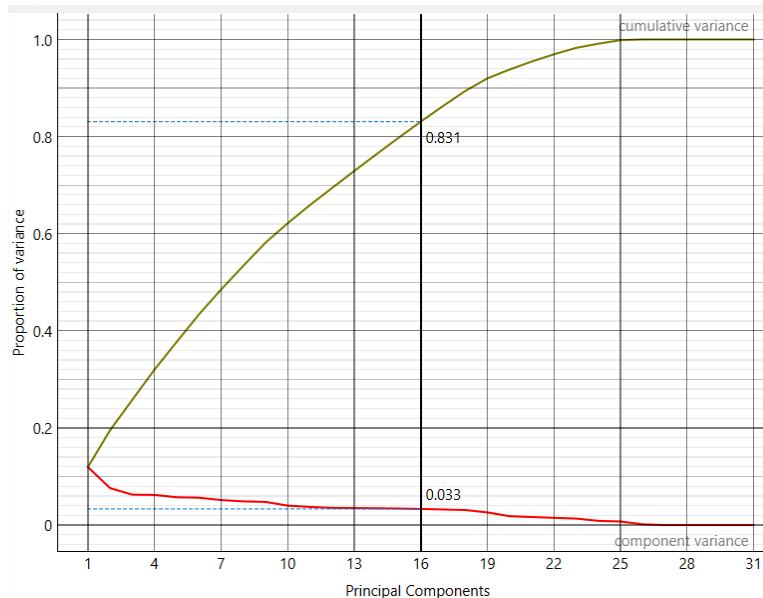
- PCA + RANK + CONTINUIZE

Nº de componentes	<4	4	5	6
Variância explicada	<80%	82%	93%	100%



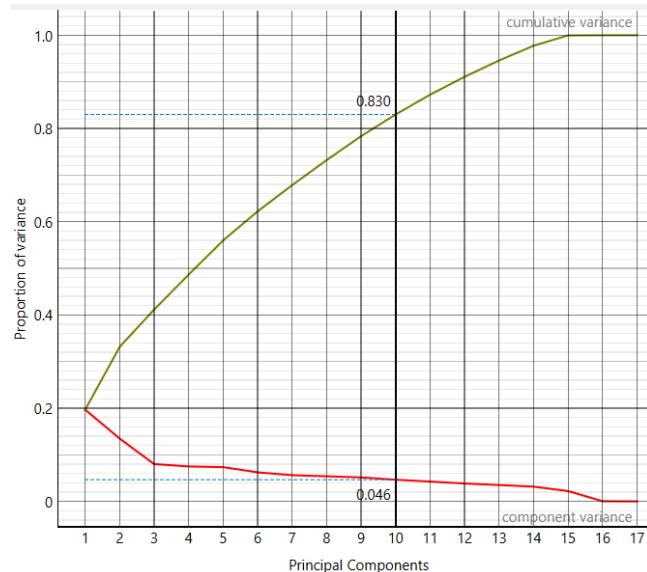
- PCA + RANK + SEM CONTINUIZE

Nº de componentes	<16	16	17	...	20	21	22	23	24-31
Variância explicada	<80%	83%	86%	...	93%	95%	96%	98%	100%



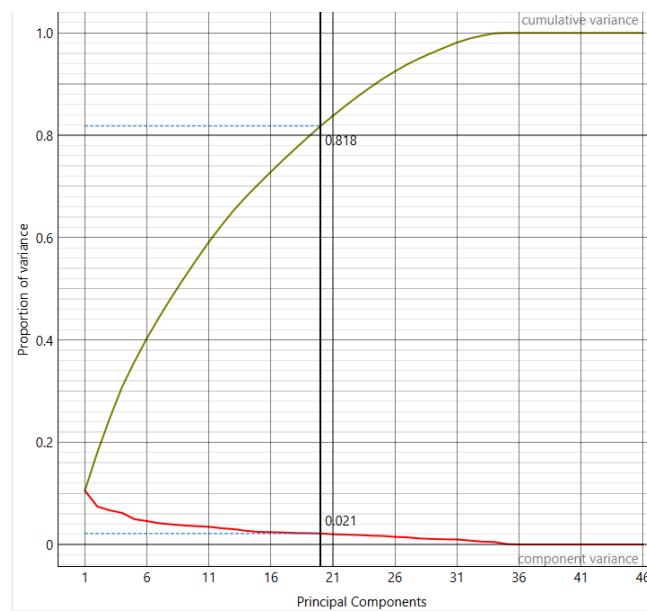
- PCA + SEM RANK + CONTINUIZE

Nº de componentes	<10	10	11	12	13	14	15-17
Variância explicada	<80%	83%	87%	91%	94%	97%	100%



- PCA + SEM RANK + SEM CONTINUIZE

Nº de componentes	<20	20	21		31/32	33-35	36-46
Variância explicada	<80%	81%	83%	...	98%	99%	100%



### c. Geração do Modelo

Após obtermos melhores resultados para test em test data no dataset anterior, decidimos manter para esse também, mas haveria a possibilidade de triplicar o número de testes, por exemplo, e incluir a cross validation com 20 e 10 folds. Para efeitos comparativos, primeiro testamos as previsões sem utilizar o PCA:

- SEM O FILTRO DE RANK E COM CONTINUIZE

Model	AUC	CA	F1	Prec	Recall	MCC
kNN	0.748	0.734	0.733	0.734	0.734	0.460
Neural Network	0.900	0.860	0.860	0.860	0.860	0.717
Random Forest	0.922	0.836	0.836	0.836	0.836	0.668
SVM	0.919	0.850	0.851	0.854	0.850	0.703
Tree	0.786	0.841	0.841	0.841	0.841	0.679

- SEM O FILTRO DE RANK E SEM CONTINUIZE

Model	AUC	CA	F1	Prec	Recall	MCC
kNN	0.759	0.744	0.743	0.743	0.744	0.481
Neural Network	0.903	0.845	0.846	0.847	0.845	0.689
Random Forest	0.926	0.845	0.845	0.846	0.845	0.687
SVM	0.914	0.860	0.860	0.864	0.860	0.722
Tree	0.811	0.797	0.798	0.803	0.797	0.599

- COM O FILTRO DE RANK E SEM CONTINUIZE

Model	AUC	CA	F1	Prec	Recall	MCC
kNN	0.899	0.841	0.841	0.843	0.841	0.682
Neural Network	0.884	0.826	0.827	0.829	0.826	0.653
Random Forest	0.895	0.845	0.846	0.847	0.845	0.689
SVM	0.873	0.841	0.841	0.845	0.841	0.683
Tree	0.856	0.836	0.836	0.839	0.836	0.673

- COM O FILTRO DE RANK E COM CONTINUIZE

Model	AUC	CA	F1	Prec	Recall	MCC
kNN	0.858	0.802	0.802	0.803	0.802	0.601
Neural Network	0.906	0.845	0.846	0.850	0.845	0.694
Random Forest	0.882	0.841	0.841	0.841	0.841	0.679
SVM	0.879	0.845	0.846	0.852	0.845	0.696
Tree	0.830	0.821	0.822	0.824	0.821	0.643

Como podemos observar, em geral as variações entre o uso ou não do filtro de RANK geralmente não surtem grandes aumentos/quedas na performance dos modelos testados. Alguns exemplos destacáveis são: k-NN indo de 73.4% a 84.1% de acurácia e Árvore de decisão indo de 79.7% a 83.6% de acurácia. Também houve um caso notável de queda de performance, na Rede Neural, onde a pior acurácia sem o filtro de RANK (86% ou 84.5%) ainda sim foi igual ou superior a melhor com filtro de RANK (84.5%). No entanto, nota-se que obtivemos um filtro sólido em nossas escolhas através do Feature Rank, haja vista que se conseguiu reduzir os 15 atributos para apenas os 6 que julgamos mais importantes, uma redução de 60% que não impactou drástica e negativamente os resultados de nenhum modelo.

Depois, utilizamos o PCA para fazer o teste com as quantidades de componentes mais extremas (mais próximas a 80% e mais próximas a 100%) das possibilidades mostradas no item b):

- PCA COM O FILTRO DE RANK COM CONTINUIZE

Model	CA	Model	CA	
kNN (4 PCs)	0.802	kNN (5 PCs)	0.816	
Neural Network (4 PCs)	0.845	Neural Network (5 PCs)	0.845	
Random Forest (4 PCs)	0.812	Random Forest (5 PCs)	0.807	
SVM (4 PCs)	0.845	SVM (5 PCs)	0.845	
Tree (4 PCs)	0.821	x	Tree (5 PCs)	0.812

Aqui a variação máxima e mínima de componentes foi pouca (4 PCs para 5 PCs), então vemos poucas mudanças de performance. Tanto o SVM quanto a Rede Neural tiveram o mesmo resultado para ambos os testes, com uma acurácia líder de 84.5%. O k-NN para 4 PCs demonstrou o pior resultado, com uma acurácia cerca de 4% menor que os melhores (80.2%).

- PCA COM O FILTRO DE RANK SEM CONTINUIZE

$\hat{M}$ Model	CA	$\hat{M}$ Model	CA
kNN (16 PCs)	0.841	kNN (23 PCs)	0.841
Neural Network (16 PCs)	0.826	Neural Network (23 PCs)	0.826
Random Forest (16 PCs)	0.831	Random Forest (23 PCs)	0.841
SVM (16 PCs)	0.841	SVM (23 PCs)	0.841
Tree (16 PCs)	0.836	x	Tree (23 PCs) 0.836

Aqui a variação máxima e mínima de componentes foi de 7 PCs de diferença, com resultados iguais para quase todos os algoritmos, exceto Random Forest. Nele, 23 PCs performou ligeiramente melhor (de 83.1% para 84.1% de acurácia).

- PCA SEM O FILTRO DE RANK COM CONTINUIZE

$\hat{M}$ Model	CA	$\hat{M}$ Model	CA
kNN (10 PCs)	0.744	kNN (14 PCs)	0.744
Neural Network (10 PCs)	0.836	Neural Network (14 PCs)	0.836
Random Forest (10 PCs)	0.860	Random Forest (14 PCs)	0.860
SVM (10 PCs)	0.855	SVM (14 PCs)	0.855
Tree (10 PCs)	0.812	x	Tree (14 PCs) 0.812

Aqui a variação máxima e mínima de componentes foi de 4 PCs de diferença, com resultados iguais para todos os algoritmos em ambos os testes.

- PCA SEM O FILTRO DE RANK SEM CONTINUIZE

$\hat{M}$ Model	CA	$\hat{M}$ Model	CA
kNN (20 PCs)	0.744	kNN (33 PCs)	0.744
Neural Network (20 PCs)	0.845	Neural Network (33 PCs)	0.845
Random Forest (20 PCs)	0.855	Random Forest (33 PCs)	0.884
SVM (20 PCs)	0.860	SVM (33 PCs)	0.860
Tree (20 PCs)	0.797	x	Tree (33 PCs) 0.797

Aqui a variação máxima e mínima de componentes foi maior que as anteriores - de 13 PCs de diferença - com resultados iguais para todos os algoritmos em ambos os testes, exceto o Random Forest, que teve um crescimento de cerca de 3% de performance para o teste com mais PCs.

O melhor de todos os modelos em todos os testes com RANK obteve uma acurácia de 84.5%, sendo um empate entre: SVM, Random Forest, e Rede Neural. O melhor de todos os modelos em todos os testes sem RANK obteve uma acurácia de 88.4%, do Random Forest.

### **Resultados e aprendizados:**

Num geral, podemos observar que respeitando o limiar de 80% de variância explicada, a performance dos modelos não tende a se alterar significativamente. Isso nos permitiu concluir que é possível enxugar a dimensionalidade deste dataset em todas as suas variações sem muito prejuízo nos resultados.

A seleção de atributos mostrou que os fatores que mais influenciaram a aprovação de créditos foram A9, A10 e A11 (conforme visto na tabela de RANK).

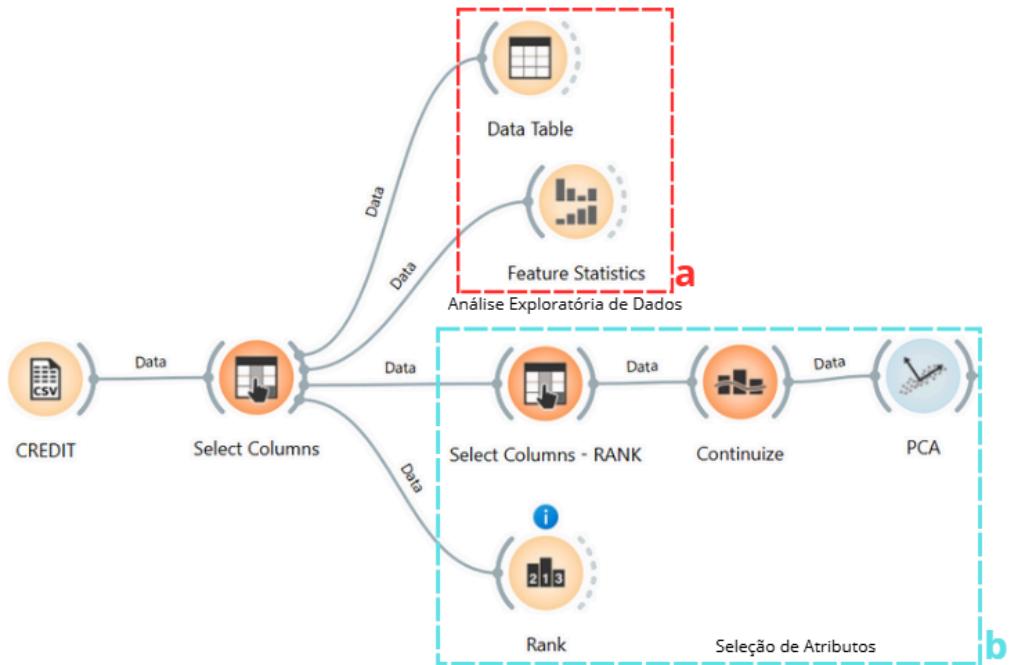
A aplicabilidade dos dados obtidos no modelo de avaliação de crédito pode ser resumida da seguinte forma:

- Classificação Automática de Risco de Crédito: o modelo passa a ter a capacidade de categorizar solicitantes em perfis de baixo, médio ou alto risco, agilizando processos de análise e decisão de aprovação de crédito.
- Segmentação de Clientes e Marketing: os atributos extraídos permitem identificar perfis de clientes e padrões de comportamento de pagamento, viabilizando campanhas direcionadas e ofertas personalizadas de produtos financeiros.
- Gestão de Carteira de Crédito: instituições financeiras podem usar o score gerado para monitorar a saúde da carteira, antecipar potenciais inadimplências e otimizar estratégias de cobrança e renegociação.
- Otimização de Políticas de Crédito: o modelo alimenta sistemas de recomendação que sugerem taxas de juros, prazos e limites ideais para cada perfil de cliente, melhorando a experiência do usuário e a rentabilidade da carteira.
- Análise de Tendências e Inteligência de Mercado: a agregação histórica dos scores possibilita identificar mudanças no comportamento de crédito e antecipar ajustes em políticas de concessão diante de tendências macroeconômicas.

Os dados extraídos podem ser usados para otimizar decisões de concessão de crédito, gestão de risco e campanhas de marketing, melhorando tanto a eficiência operacional quanto a experiência do cliente e aumentando a precisão das previsões de comportamento de pagamento no futuro.

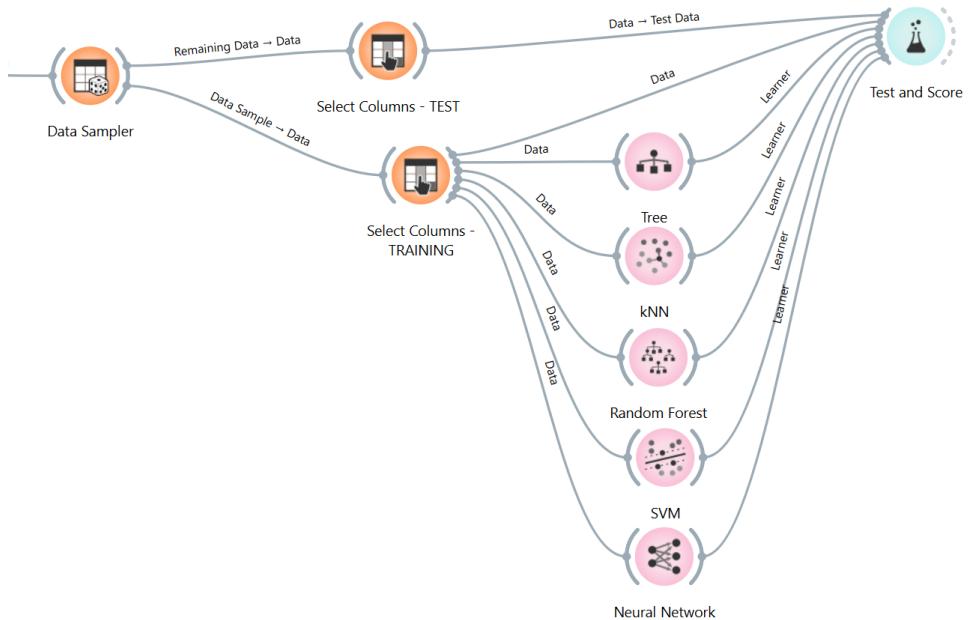
**OBS:** Apesar de destacado e utilizado em ambos os datasets, k-NN não é utilizado para regressões por que ele não realiza aprendizado de parâmetros nem ajusta um modelo; suas previsões em tarefas de regressão são obtidas diretamente pela agregação (por exemplo, média ou mediana) dos valores dos k vizinhos mais próximos.

#### d. Carga de dados e separação de conjuntos



\*OBS.: para o teste com/sem continuize e com/sem o filtro de rank, apenas ajustamos as bolinhas para obter a combinação desejada.

#### c) Geração do Modelo



Os fluxos vêm de uma única ramificação, depois de PCA. No entanto, o fluxo sem PCA passa pelo select columns e filtra para ter apenas os atributos do próprio dataset - com rank ou sem rank, enquanto os de PCA são ajustados no gráfico de PCA e têm seu próprio select columns para o conjunto PCx.