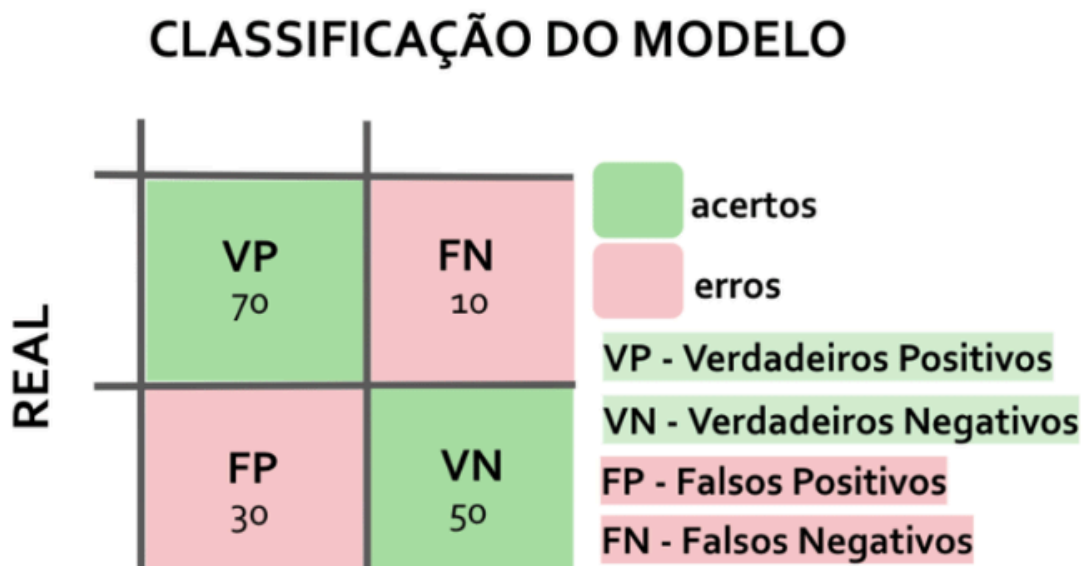


Relatório 25 - Prática: Métricas e Validação de Modelos de Aprendizado de Máquina (III)

Lucas Scheffer Hundsdorfer

Descrição da atividade

O primeiro vídeo começa explicando que existem dois tipos de modelos preditivos, o de regressão e o de classificação, o de regressão é onde o modelo deseja prever um valor específico como o preço de uma ação e essa parte de regressão usa métricas com o MSE (Mean Squared Error). Já o de classificação é onde você quer prever uma classe, como classificar se aquele email é spam ou não, esse tipo de modelo utiliza outro tipo de métricas como acurácia, precisão entre outros. Dentro dos modelos de classificação é bastante utilizado a matriz de confusão:



Cruzamento do que o modelo disse com a realidade. A acurácia parte desse conceito:

$$\frac{Acertos}{Total} = \frac{VN + VP}{VN + FN + VP + FP}$$

A partir da matriz conseguimos obter o cálculo da acurácia, a soma dos verdadeiros positivos e os verdadeiros falsos, dividido pelo total de observações. Outro conceito que parte da matriz é a precisão:

$$\frac{VP}{VP + FP}$$

O verdadeiro positivo dividido pelo verdadeiro positivo mais o falso positivo, isso resulta na precisão de seu modelo.

Outra métrica tirada a partir da matriz é o Recall ou Sensibilidade:

$$\frac{VP}{VP + FN}$$

Aqui é pego o verdadeiro positivo e é dividido pelo verdadeiro positivo mais o falso negativo. A precisão se mostra um contraponto em relação ao recall.

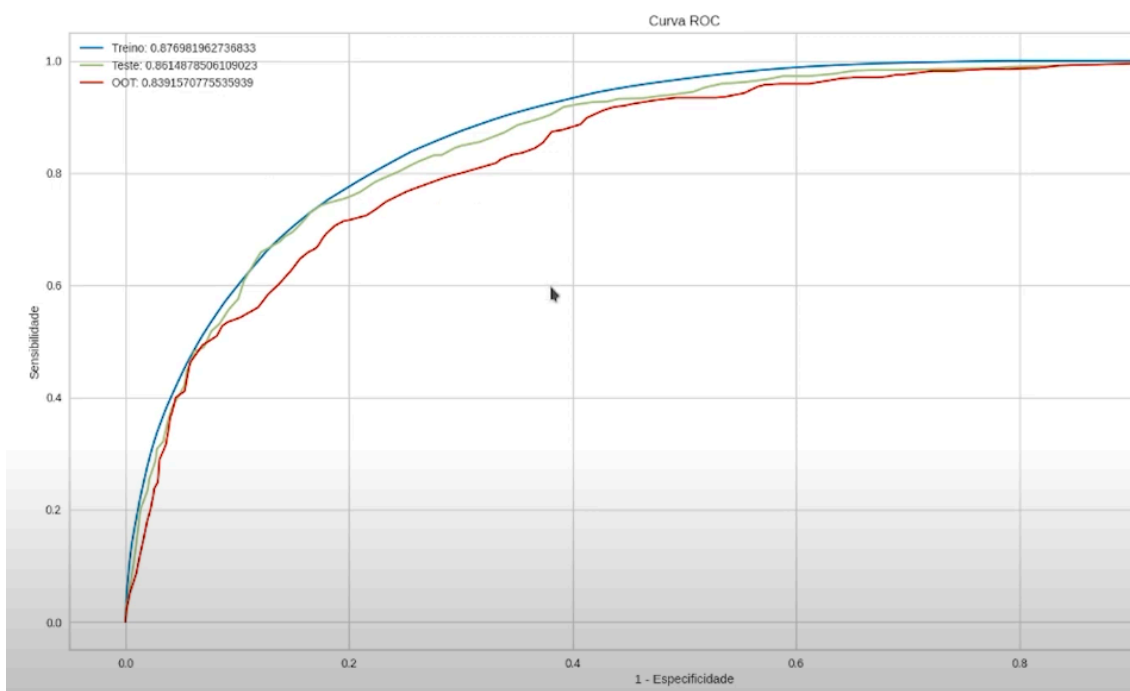
$$\frac{VN}{VN + FP}$$

Esse cálculo é a o cálculo da métrica especificidade, usado para calcular a curva roc

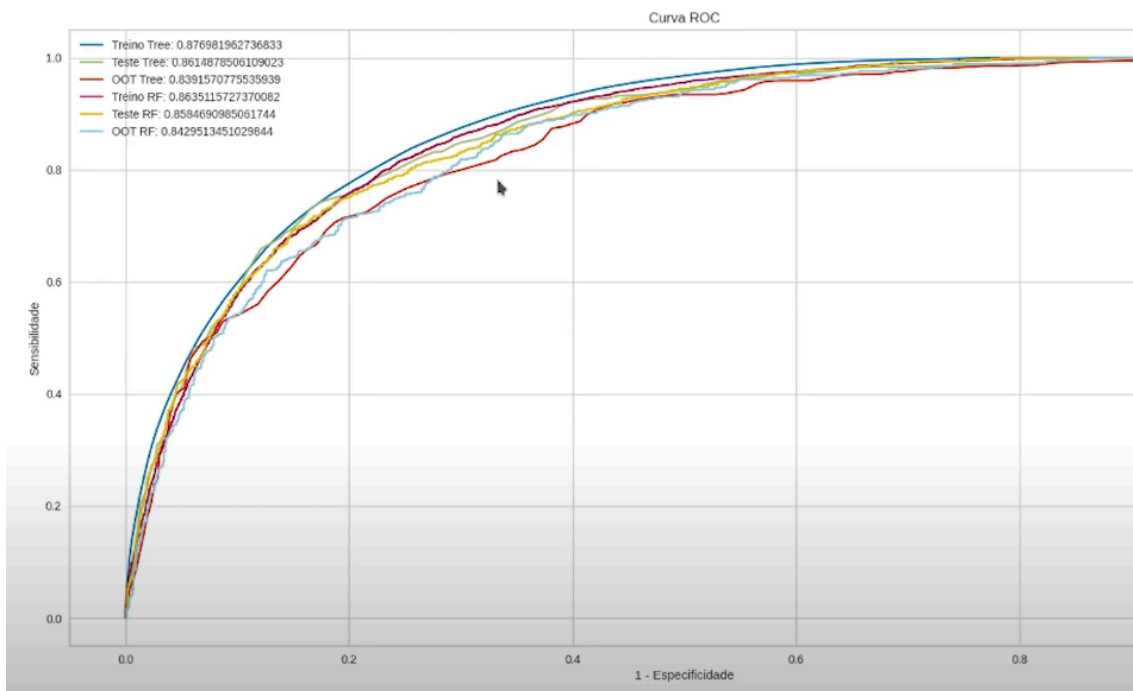
Uma das métricas mais famosas é o (AUC) ou Área sob a curva Roc que é determinada através da sensibilidade e a especificidade, é bem simples, aplicando a sensibilidade no eixo y e a especificidade no eixo x. Dentro do vídeo é utilizado o excel para apresentar a parte teórica do vídeo.

id	Verdadeiro	Probabilidade	Modelo	Verdadeiro Negativo	Falso Negativo	Falso Positivo	Verdadeiro Positivo
1	Churn	0,61	Churn	0	0	0	1
2	Churn	0,51	Não Churn	0	1	0	0
3	Não Churn	0,501	Não Churn	1	0	0	0
4	Churn	0,75	Churn	0	0	0	1
5	Churn	0,49	Não Churn	0	1	0	0
6	Não Churn	0,1	Não Churn	1	0	0	0
7	Churn	0,55	Não Churn	0	1	0	0
8	Churn	0,62	Churn	0	0	0	1
	Cutoff	0,6					
						Acurácia	62,5%
						Precisão	100,0%
						Recall	50,0%
						Especificidade	60,0%
			</				

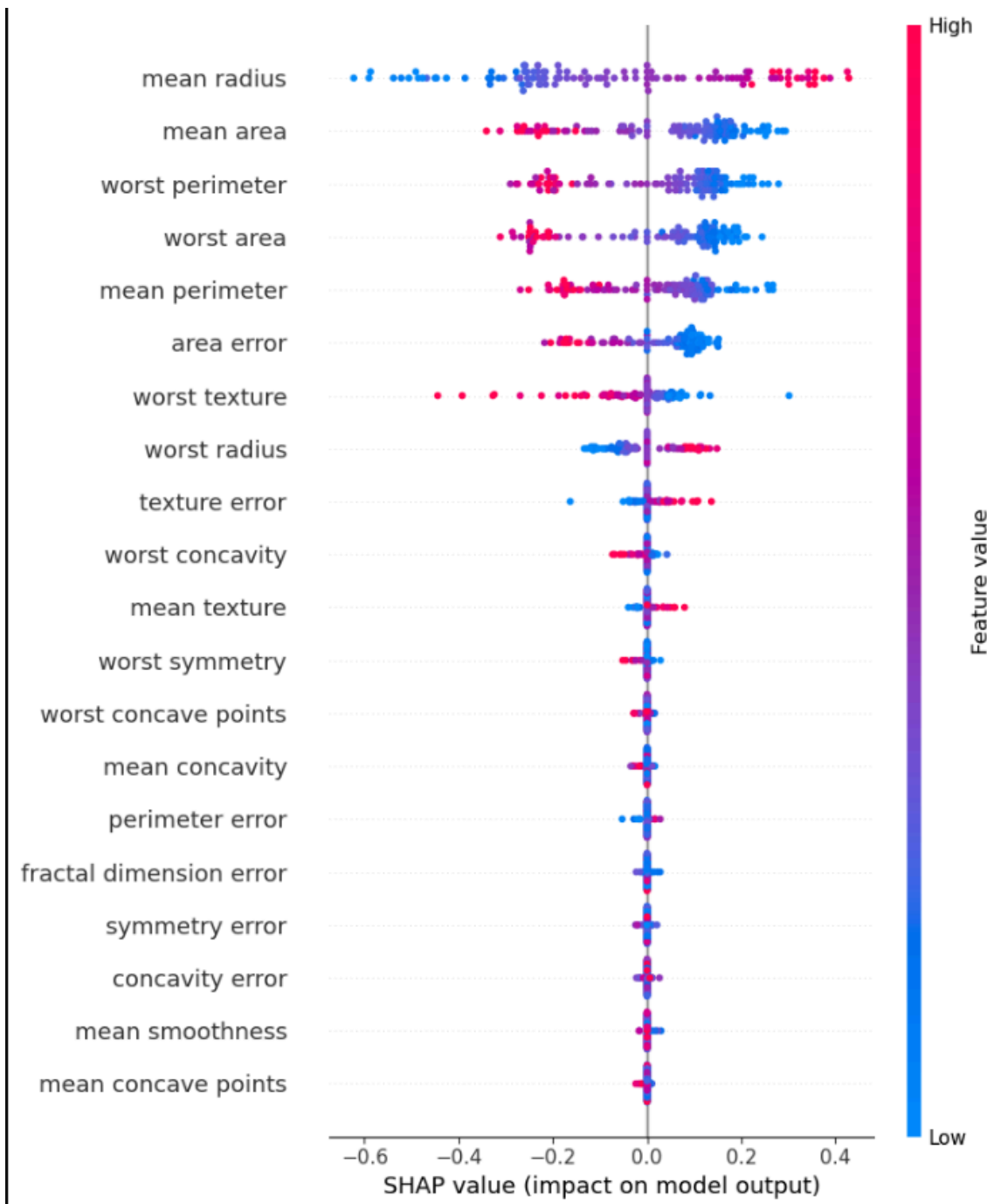
O exemplo usado utiliza o conceito de churn sendo churn nada mais do que a perda do consumidor em um período específico. E o que é demonstrado é que o modelo retorna uma probabilidade ser churn e existe esse cutoff que é o que determina a partir dele qual é churn. E assim é construída toda a matriz de confusão. A curva roc é feita com a variação do cutoff onde a cada mudança o recall e a especificidade vão alterando:



Essa é a curva roc que ele apresenta durante o vídeo, ali é mostrado a curva durante o treino, durante os testes e o OOT (out of time). A curva quanto mais 'barriguda' demonstra que o modelo está melhor. Já que a área sob a curva é maior quer dizer que ela acerta mais vezes. Também é mostrado comparando mais modelos, ele pega e treina a random forest de cada exemplo e plota utilizando o matplotlib para ver qual curva ficou melhor e coloca de legenda qual foi o ápice de cada:

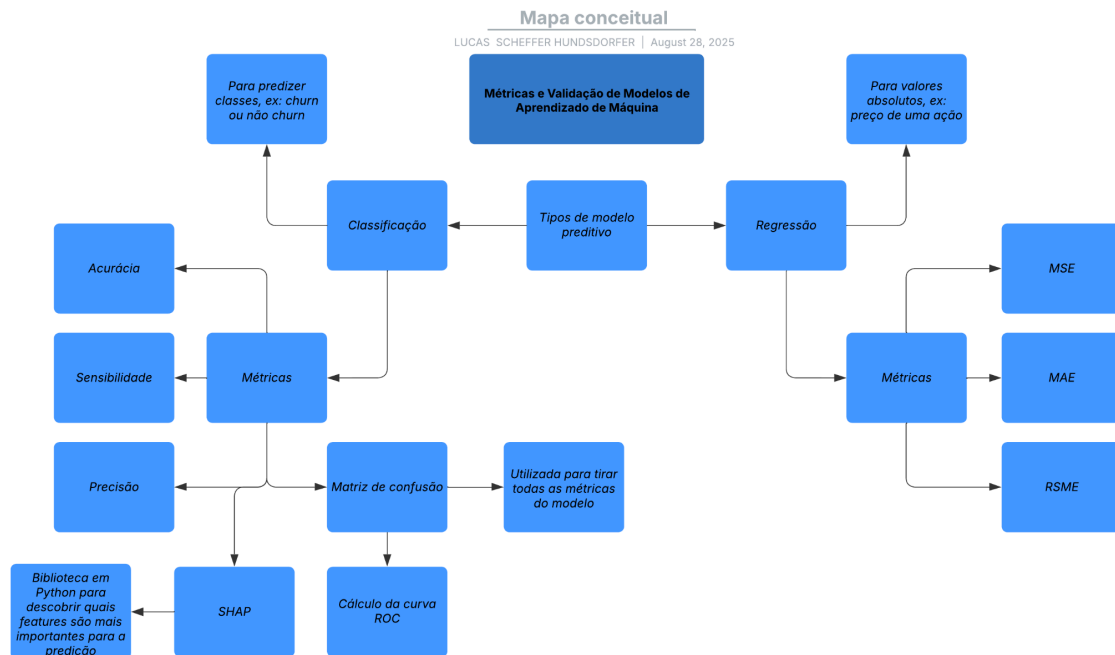


Já o segundo vídeo é bem mais prático e não envolve o cálculo das métricas para validar o modelo, porém apresenta uma biblioteca muito útil para isso que é a SHAP (SHapley Additive exPlanations), o principal objetivo dessa biblioteca é explicar quanto de cada feature para uma previsão do modelo, conseguindo dizer quanto cada variável influenciou no resultado:



No exemplo da aula foi demonstrado isso aqui, a importância de cada variável para a decisão do modelo, e aqui mostra que a 'mean concave points' é a menos importante para a predição e 'mean radius' é a mais.

Insight visual original:



Conclusões

Minha conclusão que fica através desses dois vídeos, é que as métricas não são cálculos super complicados e que também vão além do óbvio que seria a precisão e a acurácia que existe cada métrica para cada caso e que saber interpretar elas pode ajudar a entender seu modelo pode melhorar mais rápido e mais eficientemente. Fora o entendimento da biblioteca SHAP que ajuda a compreender como o seu modelo está levando em consideração cada feature.

Referências

▶ [COMPARANDO MODELOS DE MACHINE LEARNING!!](#)

▶ [Explain Machine Learning Models with SHAP in Python](#)