MÉTRICAS UTILIZADAS EM ML

Nesta sequência de artigos, observamos como os empresários podem utilizar a Inteligência Artificial e, mais especificamente, Machine Learning. Descrevemos desde a percepção até a criação de modelos mas, como avaliar se estes modelos são adequados? A partir de agora iremos adentrar mais profundamente nos parâmetros de desempenho destes algoritmos.

Tipos de Métricas

A qualidade do aprendizado realizado pelos algoritmos é geralmente avaliada pelo desempenho que estes algoritmos demonstram no processo de treinamento junto aos dados, especificamente, no conjunto de testes que é uma amostra dos dados originais mas que não foram utilizados no processo de treinamento. Algumas métricas são calculadas a partir do desempenho do algoritmo. Dependendo do tipo de algoritmo e de como o cientista de dados quer avaliar este desempenho pode-se ter três possibilidades distintas:

- **Predição nominal da classe**: em que os rótulos de classe previstos são comparados aos valores de classe real;
- **Previsões com pontuação numérica**: considera algum escore numérico associado as previsões para classificar os exemplos de teste de acordo com a probabilidade de pertencer a uma classe;
- **Previsões probabilísticas**: onde as saídas numéricas associadas à previsão são interpretadas como probabilidades dos exemplos pertencentes à classe.

Aqui iremos tratar apenas da Predição nominal das classes tendo em vista que os outros dois métodos podem ser avaliados e calibrados de diversas formas.

		Predicted class		
		Positive	Negative	Total
Actual class	Positive	TP	FN	POS
	Negative	FP	TN	NEG
	Total	$PPOS$	$PNEG$	N

Figura 1 - Matriz de Cofusão para um problema Biclasse.

A melhor forma de avaliarmos um algoritmo de Machine Learning é através da construção da Matriz de Cofusão como mostrado na Figura 1 acima. Nesta matriz temos dois conjuntos de dados, os dados reais e os dados preditos, respectivamente ***Actual Class*** e ***Predicted Class***. Cada classe, por sua vez, possui exemplos **positivos** e **negativos**. A classe minoritária (que possui menor número de exemplos do total dos dados) é chamada de **negativa** e a classe dominante de **positiva**. Para elucidar, considere um exemplo de mamografias. As mamografias que estão normais pertencem à classe majoritária (maior número de exemplos) e são chamadas de **classe positiva**. Já as mamografias com tumores (que são a minoria) são denominadas da **classe negativa**. Na Figura 1, **TP** (Positivo-Verdadeiro) e **TN** (Negativo- Verdadeiro) indicam a classificação **correta** dos exemplos das classes Positivas e Negativas, respectivamente. **FN** (Negativo-Falso) e **FP** (Positivo-Falso) indicam exemplos positivos e negativos classificados de forma errada como negativo e positivo, respectivamente.

A partir desta matriz, diferentes métricas de desempenho podem ser extraídas. Estas medidas correspondem a diferentes formas de avaliar se o classificador é bom ou ruim. Um parâmetro de desempenho muito utilizado e comum para classificadores é a **acurácia** ou o seu complemento: **taxa-de-erro**. Acurácia é a porcentagem dos exemplos classificados de forma correta. Para dados balanceados, quanto mais próximo de 1 for a acurácia, melhor o classificador.

$$Acc = \frac{TP + TN}{N}$$

$$Error = 1 - Acc = \frac{FP + FN}{N}$$

Acc - Acurácia;
N - Número total de exemplos submetidos ao classificador;
Error - Taxa de Erro;
TP - Positivo-**Verdadeiro** (Classificado de forma correta)
TN - Negativo-**Verdadeiro** (Classificado de forma correta)

F-measure ou F1-score é uma métrica com foco na classe positiva e é muito utilizada na área de Recuperação da Informação (Information Retrieval). Quanto mais próximo de 1 for o F1-score, melhor é o classificador.

$$\text{Precision} = \frac{tp}{tp + fp} \qquad \text{True negative rate} = \frac{tn}{tn + fp}$$

$$\text{Recall} = \frac{tp}{tp + fn} \qquad \text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

(https://en.wikipedia.org/wiki/Precision_and_recall)

$$F_1 = 2\frac{precision \cdot recall}{precision + recall}$$

A biblioteca scikit-learn (https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html) já possui esta e outras métricas já programadas em Python.
Um pequeno exemplo de como calcular F1 em Python:

```
In[12]: from sklearn.metrics import f1_score
   ...: y_true = [0, 1, 2, 0, 1, 2]
   ...: y_pred = [0, 2, 1, 0, 0, 1]
   ...: F1 = f1_score(y_true, y_pred,average='macro')
   ...: print(F1)
0.26666666666666666
```

Neste caso temos que F1=26,6% que é um resultado ruim! Teríamos de refazer o modelo para chegarmos próximos aos 80% ou mais.

Abaixo temos um exemplo da acurácia.

```
In[13]: import numpy as np
   ...: from sklearn.metrics import accuracy_score
   ...: y_pred = [0, 2, 1, 3]
   ...: y_true = [0, 1, 2, 3]
   ...: ACC = accuracy_score(y_true, y_pred)
In[14]: print(ACC)
0.5
```

Aqui, o valor de Acc é de 50% o que também é muito ruim!

Agora que sabemos o que as principais métricas significam, ficou mais fácil avaliar-se os modelos de Machine Learning.

Bibliografia

Learning from Imbalanced Data Sets (ISBN 978-3-319-98073-7, ISBN 978-3-319-98074-4(eBook), https://doi.org/10.1007/978-3-319-98074-4)
Scikit-learn (https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html)
Wikipedia (https://en.wikipedia.org/wiki/Precision_and_recall)