Conjuntos de dados com distribuição não uniforme (desbalanceados)

Conjuntos de dados com distribuição não uniforme (desbalanceados)

Conjuntos de dados com distribuição não uniforme

Pergunta:

Taxa de acerto/erro é sempre uma boa métrica?

Exemplo:

- Você está construindo um algoritmo que reconhece uma determinada doença rara ($y = 1$ caso a doença esteja presente, e $y = 0$ caso não esteja)
- Seu conjunto de dados é formado por dados de 1000 pacientes, onde apenas 5 deles possui a doença
- Você verificou que a taxa de acerto do seu modelo para os **dados de teste** é de 99.5%.
- Tudo certo então?

Conjuntos de dados com distribuição não uniforme

Pergunta:

Taxa de acerto/erro é sempre uma boa métrica?

Exemplo:

- Você está construindo um algoritmo que reconhece uma determinada doença rara ($y = 1$ caso a doença esteja presente, e $y = 0$ caso não esteja)
- Seu conjunto de dados é formado por dados de 1000 pacientes, onde apenas 5 deles possui a doença
- Você verificou que a taxa de acerto do seu modelo para os **dados de teste** é de 99.5%.
- Tudo certo então?

Pergunta:

- Qual seria a taxa de acerto do modelo que considera $\hat{y} = 0$ para qualquer paciente?
- O que esse modelo aprendeu sobre a doença?

Conjuntos de dados com distribuição não uniforme

Pergunta:

Taxa de acerto/erro é sempre uma boa métrica?

Exemplo:

- Você está construindo um algoritmo que reconhece uma determinada doença rara ($y = 1$ caso a doença esteja presente, e $y = 0$ caso não esteja)
- Seu conjunto de dados é formado por dados de 1000 pacientes, onde apenas 5 deles possui a doença
- Você verificou que a taxa de acerto do seu modelo para os **dados de teste** é de 99.5%.
- Tudo certo então?

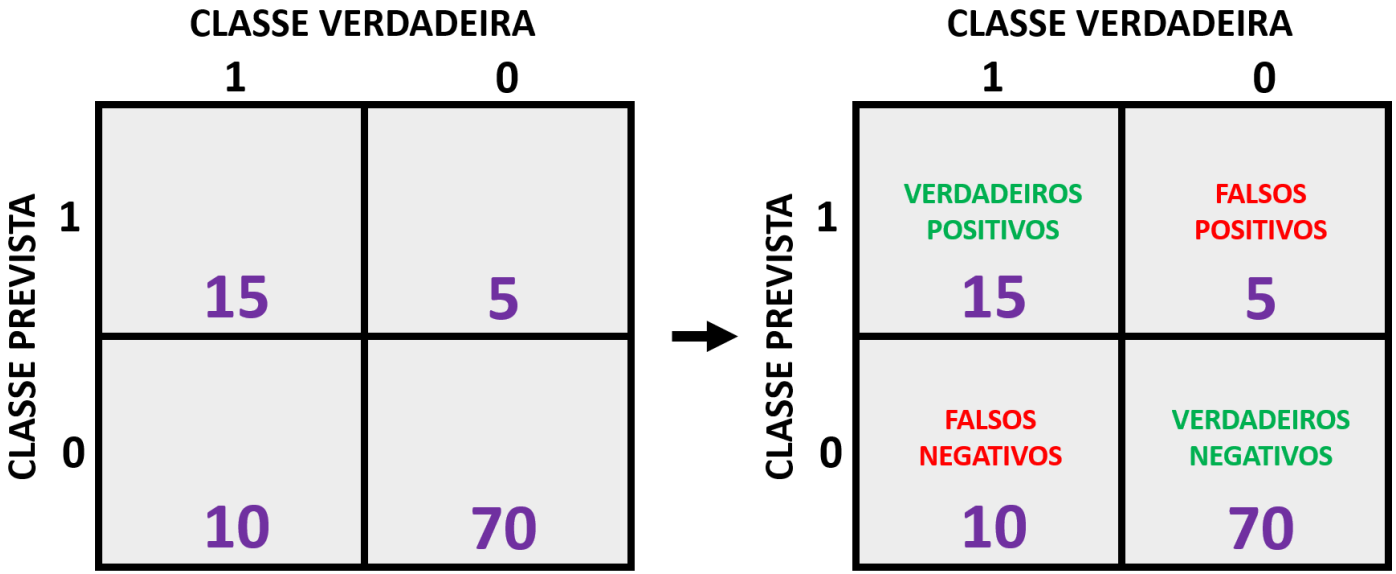
Pergunta:

- Qual seria a taxa de acerto do modelo que considera $\hat{y} = 0$ para qualquer paciente?
- O que esse modelo aprendeu sobre a doença?

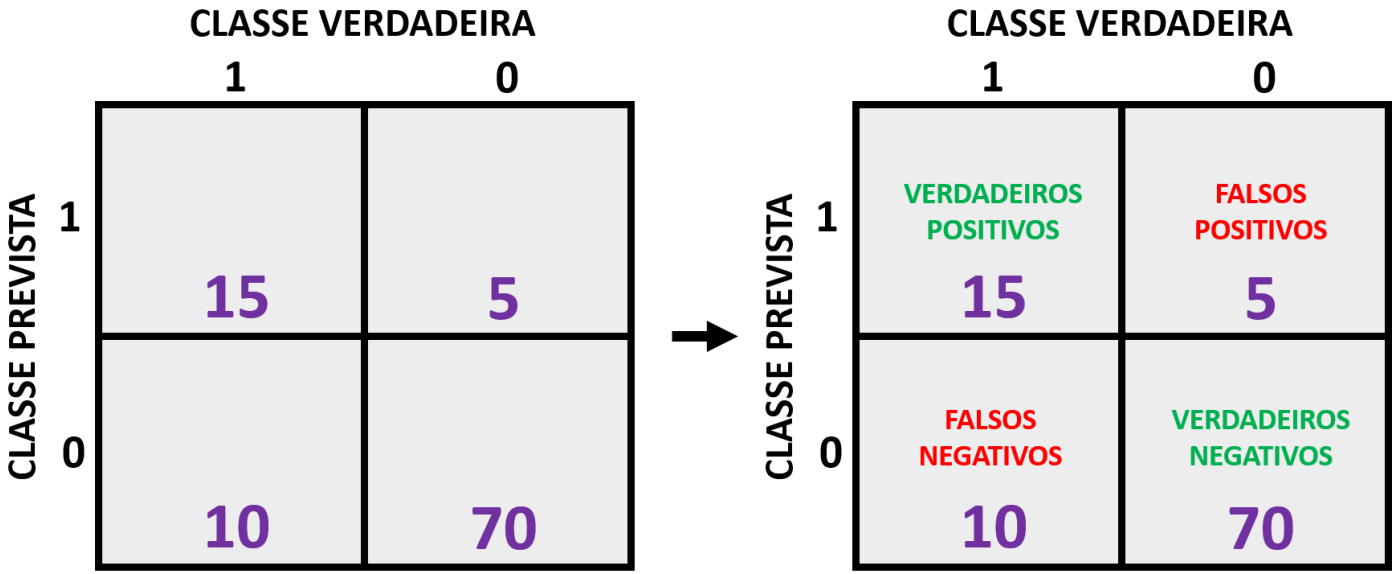
Resposta:

- Temos muitos $y = 0$ e poucos $y = 1$. Ou seja, os dados possuem uma distribuição não uniforme. Nesses casos, a taxa de acerto pode não ser uma boa métrica para avaliar a qualidade do seu modelo.

Matriz de confusão



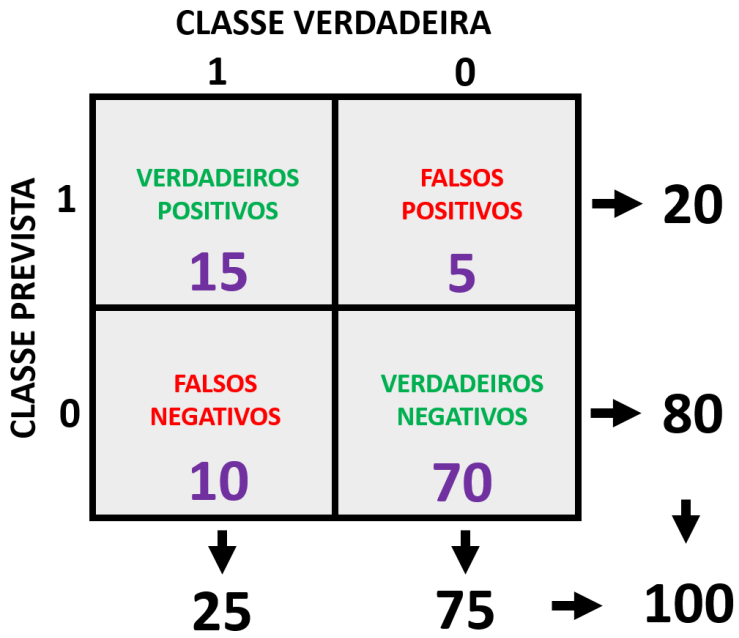
Matriz de confusão



Considere a **matriz de confusão** acima. Pergunta-se:

- Quantos pacientes que possuem a doença foram corretamente considerados doentes pelo modelo? (verdadeiros positivos) $\rightarrow$ $(y = 1, \hat{y} = 1)$
- Quantos pacientes que **não** possuem a doença foram corretamente considerados não doentes pelo modelo? (verdadeiros negativos) $\rightarrow$ $(y = 0, \hat{y} = 0)$
- Quantos pacientes que possuem a doença foram incorretamente considerados não doentes pelo modelo? (falsos negativos) $\rightarrow$ $(y = 1, \hat{y} = 0)$
- Quantos pacientes que **não** possuem a doença foram incorretamente considerados doentes pelo modelo? (falsos positivos) $\rightarrow$ $(y = 0, \hat{y} = 1)$

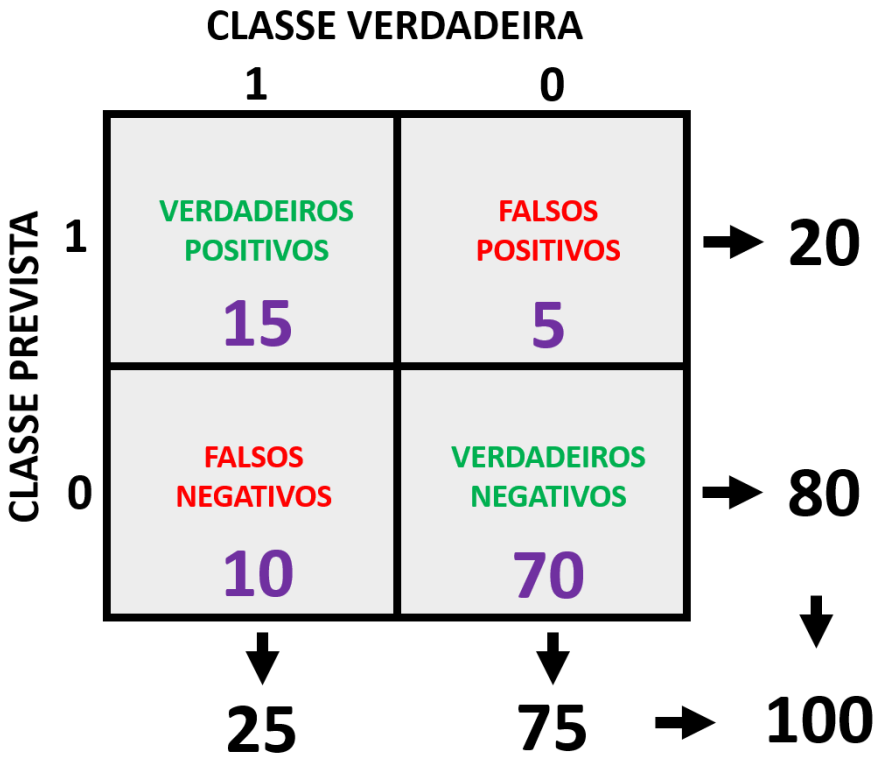
Matriz de confusão



Pergunta-se ainda:

- Quantos pacientes são de fato doentes? (amostras com $y = 1$)
- Quantos pacientes são não doentes? (amostras com $y = 0$)
- Qual é a proporção entre classes 1 e classes 0?
- Quantos pacientes foram classificados como doentes? (amostras com $\hat{y} = 1$)
- Quantos pacientes foram classificados como não doentes? (amostras com $\hat{y} = 0$)
- O número total de pacientes é dado por VP+VN+FP+FN?

Matriz de confusão



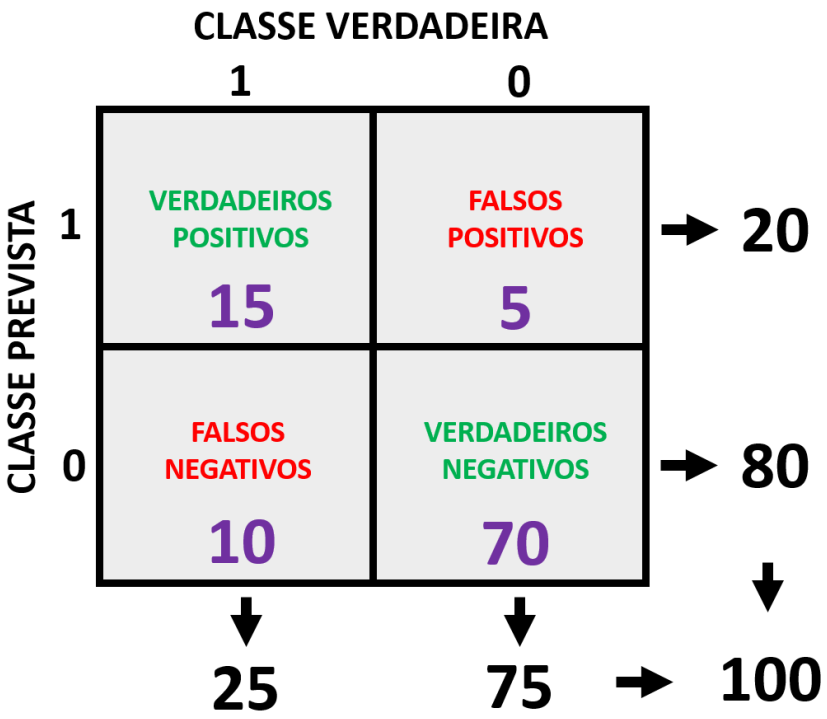
$$\text{Taxa de acerto} = \frac{VP+VN}{VP+VN+FP+FN} = \frac{15+70}{100} = 85\%$$

Pergunta-se ainda:

- Parece um bom modelo para você?
- Você concorda que a taxa de acerto desse modelo é dada conforme abaixo?

$$\text{taxa de acerto} = \frac{\text{verdadeiros positivos} + \text{verdadeiros negativos}}{\text{número total de amostras}}$$

Mais duas métricas importantes: Precisão e Recall (Revocação)



$$\text{Taxa de acerto} = \frac{VP+VN}{VP+VN+FP+FN} = \frac{15+70}{100} = 85\%$$

$$\text{precisão} = \frac{VP}{VP+FP} = \frac{15}{15+5} = 75\%$$

Métrica que penaliza a ocorrência de **FP**

$$\text{recall} = \frac{VP}{VP+FN} = \frac{15}{15+10} = 60\%$$

Métrica que penaliza a ocorrência de **FN**

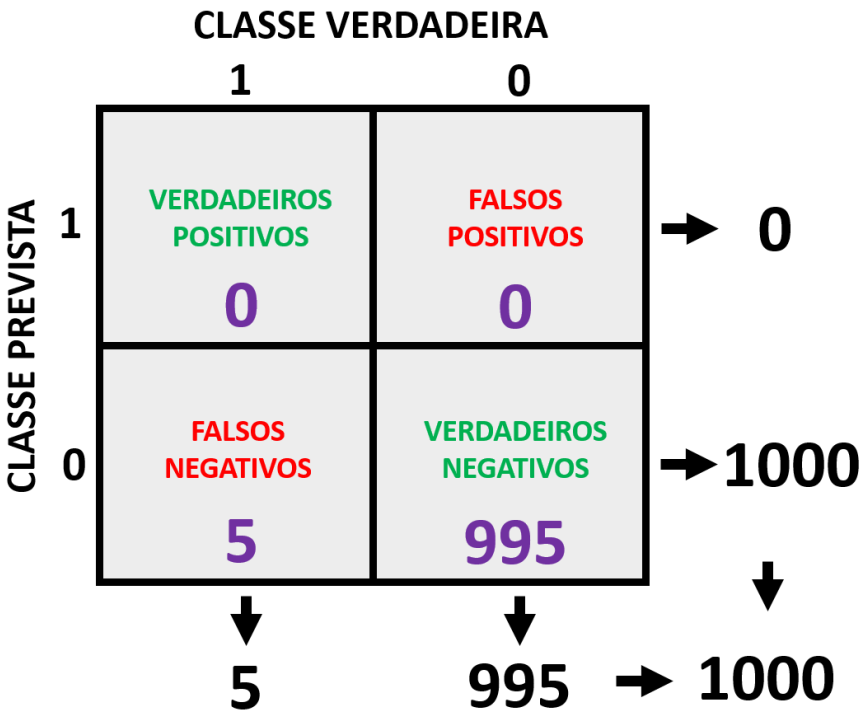
$$\text{F1 score} = \frac{2(\text{precisão})(\text{recall})}{(\text{precisão})+(\text{recall})} = \frac{2(0.75)(0.60)}{0.75+0.65} = 64.3\%$$

Métrica que engloba as duas anteriores

$$\text{precisão} = \frac{\text{pacientes considerados doentes e que eram de fato doentes}}{\text{pacientes considerados doentes}}$$

$$\text{recall} = \frac{\text{pacientes considerados doentes e que eram de fato doentes}}{\text{pacientes de fato doentes}}$$

Voltando para o exemplo anterior onde o modelo considera $\hat{y} = 0$ para qualquer paciente



Taxa de acerto $= \frac{VP+VN}{VP+VN+FP+FN} = \frac{0+995}{1000} = 99.5\%$

precisão $= \frac{VP}{VP+FP} = \frac{0}{0+0} = div/0$

Métrica que penaliza a ocorrência de **FP**

recall $= \frac{VP}{VP+FN} = \frac{0}{0+5} = 0\%$

Métrica que penaliza a ocorrência de **FN**

F1 score $= \frac{2(\text{precisão})(\text{recall})}{(\text{precisão})+(\text{recall})} = \frac{2(div/0)(0)}{div/0+0} = ?$

Métrica que engloba as duas anteriores

Não parece um bom modelo...

Algumas conclusões até aqui...

Algumas conclusões até aqui...

1. Quando temos um conjunto de dados desbalanceado (mais $y = 1$ do que $y = 0$), vimos que a taxa de acerto (acurácia) pode não ser uma boa métrica.

Algumas conclusões até aqui...

1. Quando temos um conjunto de dados desbalanceado (mais $y = 1$ do que $y = 0$), vimos que a taxa de acerto (acurácia) pode não ser uma boa métrica.
2. Uma solução para este problema consiste em utilizar outras métricas, tais como **precisão**, **recall** (revocação) e **F1 score**.

Algumas conclusões até aqui...

1. Quando temos um conjunto de dados desbalanceado (mais $y = 1$ do que $y = 0$), vimos que a taxa de acerto (acurácia) pode não ser uma boa métrica.
2. Uma solução para este problema consiste em utilizar outras métricas, tais como **precisão**, **recall** (revocação) e **F1 score**.
3. É desejável que um modelo apresente precisão e recall próximos de 100 %. Isso também resultará num F1 score próximo de 100 %.

Algumas conclusões até aqui...

1. Quando temos um conjunto de dados desbalanceado (mais $y = 1$ do que $y = 0$), vimos que a taxa de acerto (acurácia) pode não ser uma boa métrica.
2. Uma solução para este problema consiste em utilizar outras métricas, tais como **precisão**, **recall** (revocação) e **F1 score**.
3. É desejável que um modelo apresente precisão e recall próximos de 100 %. Isso também resultará num F1 score próximo de 100 %.
4. Entretanto, é muito importante sempre avaliar o caso concreto que está sob estudo.

Algumas conclusões até aqui...

1. Quando temos um conjunto de dados desbalanceado (mais $y = 1$ do que $y = 0$), vimos que a taxa de acerto (acurácia) pode não ser uma boa métrica.
2. Uma solução para este problema consiste em utilizar outras métricas, tais como **precisão**, **recall** (revocação) e **F1 score**.
3. É desejável que um modelo apresente precisão e recall próximos de 100 %. Isso também resultará num F1 score próximo de 100 %.
4. Entretanto, é muito importante sempre avaliar o caso concreto que está sob estudo.
5. Veremos um exemplo disso agora sobre transações financeiras fraudulentas.

Exemplo: transações financeiras fraudulentas

Transações financeiras fraudulentas

Suponha que você treinou uma rede neural que objetiva identificar transações financeiras fraudulentas envolvendo cartão de crédito.

Transações financeiras fraudulentas

Suponha que você treinou uma rede neural que objetiva identificar transações financeiras fraudulentas envolvendo cartão de crédito.

1. O conjunto de dados de validação possui registro de um total de 56961 transações financeiras

Transações financeiras fraudulentas

Suponha que você treinou uma rede neural que objetiva identificar transações financeiras fraudulentas envolvendo cartão de crédito.

1. O conjunto de dados de validação possui registro de um total de 56961 transações financeiras
2. Desse total, 56886 são transações legítimas ($y = 0$) e 75 são fraudulentas ($y = 1$)

Transações financeiras fraudulentas

Suponha que você treinou uma rede neural que objetiva identificar transações financeiras fraudulentas envolvendo cartão de crédito.

1. O conjunto de dados de validação possui registro de um total de 56961 transações financeiras
2. Desse total, 56886 são transações legítimas ($y = 0$) e 75 são fraudulentas ($y = 1$)
3. Note o evidente desbalanceamento presente nos dados.

Transações financeiras fraudulentas

Suponha que você treinou uma rede neural que objetiva identificar transações financeiras fraudulentas envolvendo cartão de crédito.

1. O conjunto de dados de validação possui registro de um total de 56961 transações financeiras
2. Desse total, 56886 são transações legítimas ($y = 0$) e 75 são fraudulentas ($y = 1$)
3. Note o evidente desbalanceamento presente nos dados.
4. Note que um modelo hipotético do tipo $\hat{y} = 0$ teria uma taxa de acerto muito próxima de 100 %. Entretanto, nenhuma operação fraudulenta seria detectada por este modelo.

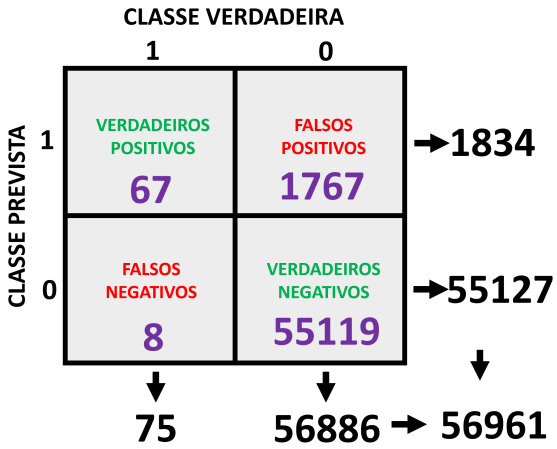
Transações financeiras fraudulentas

Suponha que você treinou uma rede neural que objetiva identificar transações financeiras fraudulentas envolvendo cartão de crédito.

1. O conjunto de dados de validação possui registro de um total de 56961 transações financeiras
2. Desse total, 56886 são transações legítimas ($y = 0$) e 75 são fraudulentas ($y = 1$)
3. Note o evidente desbalanceamento presente nos dados.
4. Note que um modelo hipotético do tipo $\hat{y} = 0$ teria uma taxa de acerto muito próxima de 100 %. Entretanto, nenhuma operação fraudulenta seria detectada por este modelo.

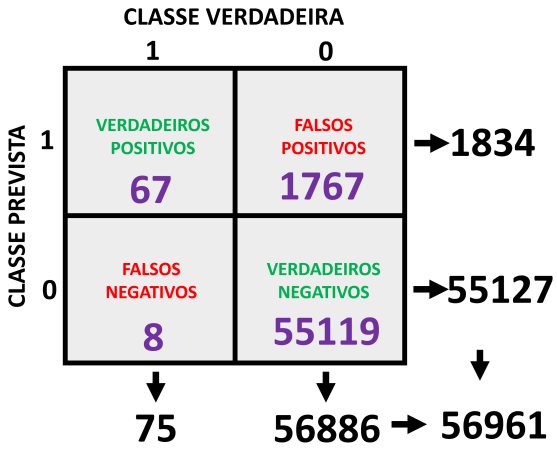
A rede neural foi submetida aos dados acima descritos, resultando na **matriz de confusão** do próximo slide.

Transações financeiras fraudulentas



Pergunta-se:

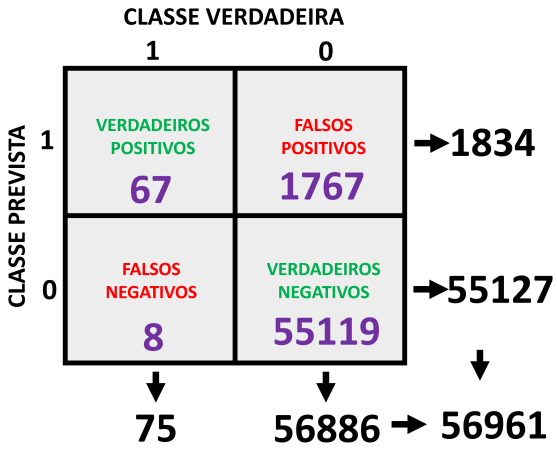
Transações financeiras fraudulentas



Pergunta-se:

- Quantas transações fraudulentas foram corretamente consideradas fraudulentas pelo modelo? (verdadeiros positivos) $\rightarrow$ $(y = 1, \hat{y} = 1)$

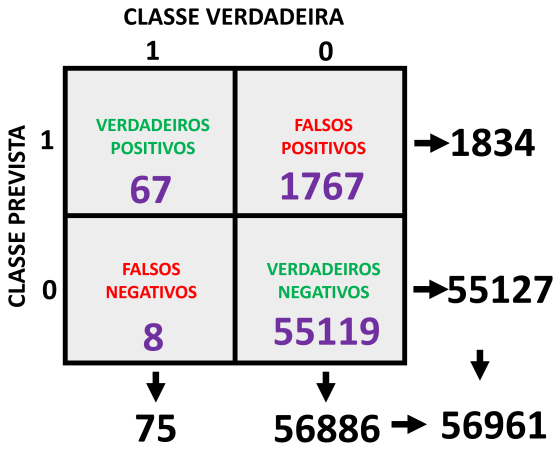
Transações financeiras fraudulentas



Pergunta-se:

- Quantas transações fraudulentas foram corretamente consideradas fraudulentas pelo modelo? (verdadeiros positivos) $\rightarrow$ $(y = 1, \hat{y} = 1)$
- Quantas transações **não** fraudulentas foram corretamente consideradas não fraudulentas pelo modelo? (verdadeiros negativos) $\rightarrow$ $(y = 0, \hat{y} = 0)$

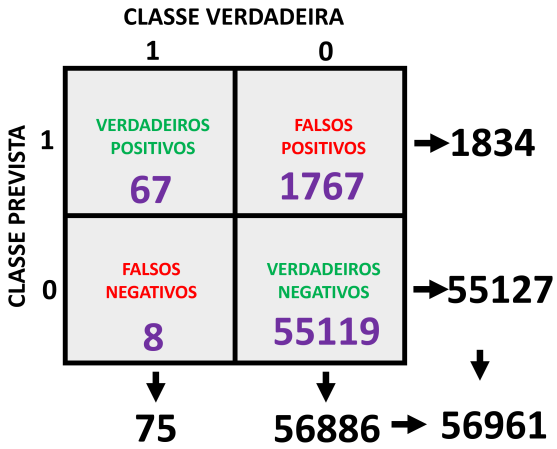
Transações financeiras fraudulentas



Pergunta-se:

- Quantas transações fraudulentas foram corretamente consideradas fraudulentas pelo modelo? (verdadeiros positivos) $\rightarrow$ $(y = 1, \hat{y} = 1)$
- Quantas transações **não** fraudulentas foram corretamente consideradas não fraudulentas pelo modelo? (verdadeiros negativos) $\rightarrow$ $(y = 0, \hat{y} = 0)$
- Quantas transações fraudulentas foram incorretamente consideradas não fraudulentas pelo modelo? (falsos negativos) $\rightarrow$ $(y = 1, \hat{y} = 0)$

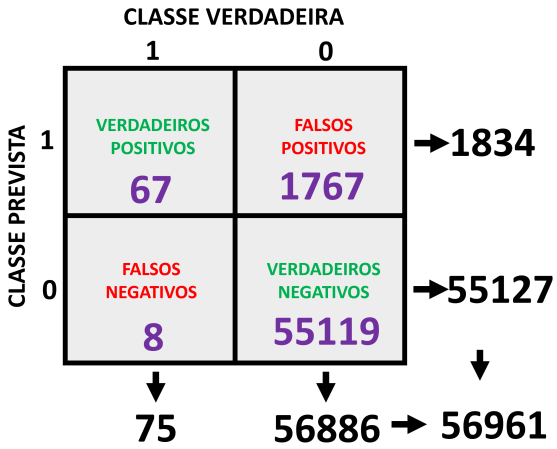
Transações financeiras fraudulentas



Pergunta-se:

- Quantas transações fraudulentas foram corretamente consideradas fraudulentas pelo modelo? (verdadeiros positivos) $\rightarrow$ $(y = 1, \hat{y} = 1)$
- Quantas transações **não** fraudulentas foram corretamente consideradas não fraudulentas pelo modelo? (verdadeiros negativos) $\rightarrow$ $(y = 0, \hat{y} = 0)$
- Quantas transações fraudulentas foram incorretamente consideradas não fraudulentas pelo modelo? (falsos negativos) $\rightarrow$ $(y = 1, \hat{y} = 0)$
- Quantas transações **não** fraudulentas foram incorretamente consideradas fraudulentas pelo modelo? (falsos positivos) $\rightarrow$ $(y = 0, \hat{y} = 1)$

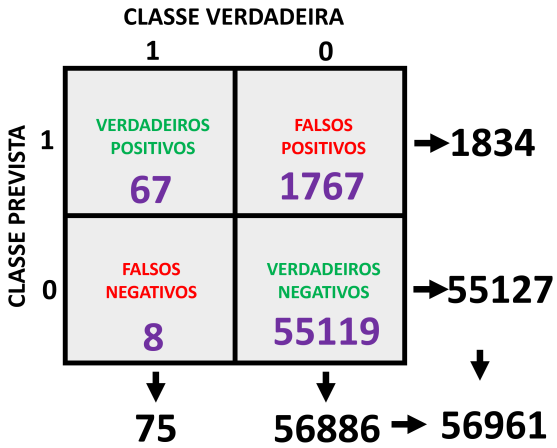
Transações financeiras fraudulentas



Pergunta-se:

- Quantas transações fraudulentas foram corretamente consideradas fraudulentas pelo modelo? (verdadeiros positivos) $\rightarrow$ $(y = 1, \hat{y} = 1)$
- Quantas transações **não** fraudulentas foram corretamente consideradas não fraudulentas pelo modelo? (verdadeiros negativos) $\rightarrow$ $(y = 0, \hat{y} = 0)$
- Quantas transações fraudulentas foram incorretamente consideradas não fraudulentas pelo modelo? (falsos negativos) $\rightarrow$ $(y = 1, \hat{y} = 0)$
- Quantas transações **não** fraudulentas foram incorretamente consideradas fraudulentas pelo modelo? (falsos positivos) $\rightarrow$ $(y = 0, \hat{y} = 1)$
- E aí, trata-se de um bom modelo?

Transações financeiras fraudulentas

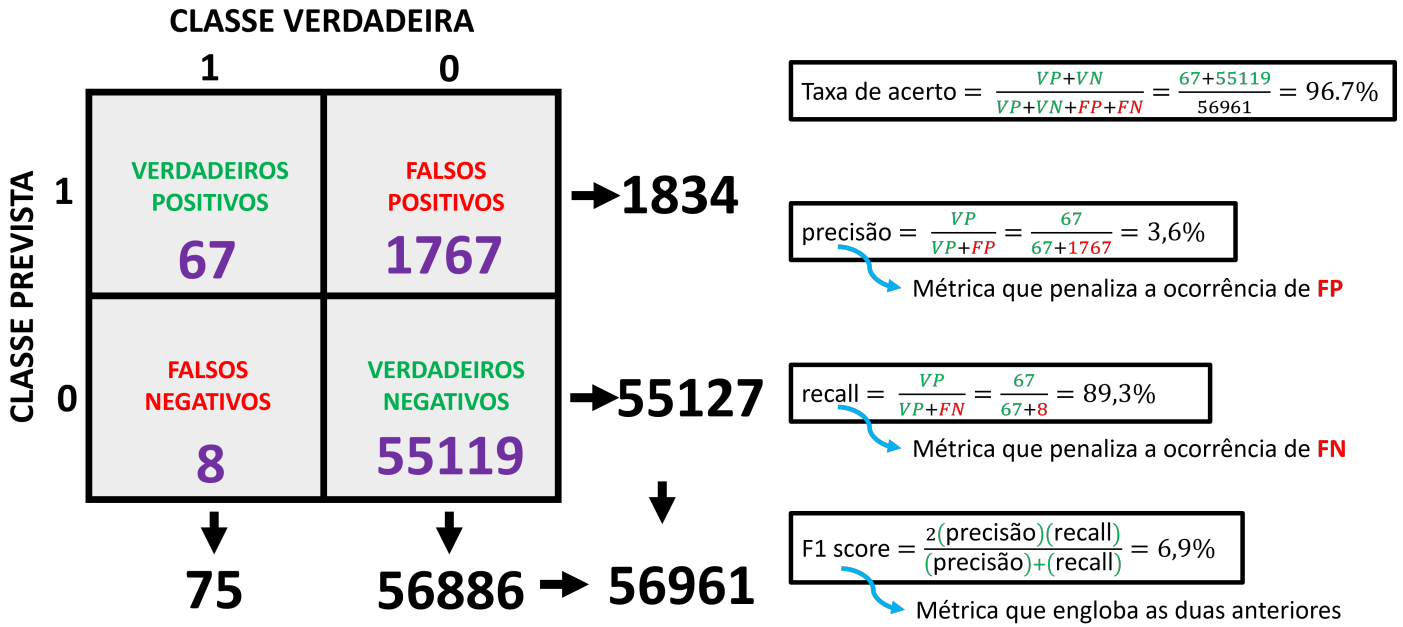


Pergunta-se:

- Quantas transações fraudulentas foram corretamente consideradas fraudulentas pelo modelo? (verdadeiros positivos) $\rightarrow$ $(y = 1, \hat{y} = 1)$
- Quantas transações **não** fraudulentas foram corretamente consideradas não fraudulentas pelo modelo? (verdadeiros negativos) $\rightarrow$ $(y = 0, \hat{y} = 0)$
- Quantas transações fraudulentas foram incorretamente consideradas não fraudulentas pelo modelo? (falsos negativos) $\rightarrow$ $(y = 1, \hat{y} = 0)$
- Quantas transações **não** fraudulentas foram incorretamente consideradas fraudulentas pelo modelo? (falsos positivos) $\rightarrow$ $(y = 0, \hat{y} = 1)$
- E aí, trata-se de um bom modelo?
- Note que o modelo é bastante razoável, pois ele foi capaz de identificar 67 das 75 transações fraudulentas presentes. Entretanto, 1767 transações não fraudulentas (de um total de 56886) foram classificadas como fraudulentas. "Na próxima vez que sua compra usando cartão de crédito não for autorizada, você já tem uma ideia do que pode ter acontecido".

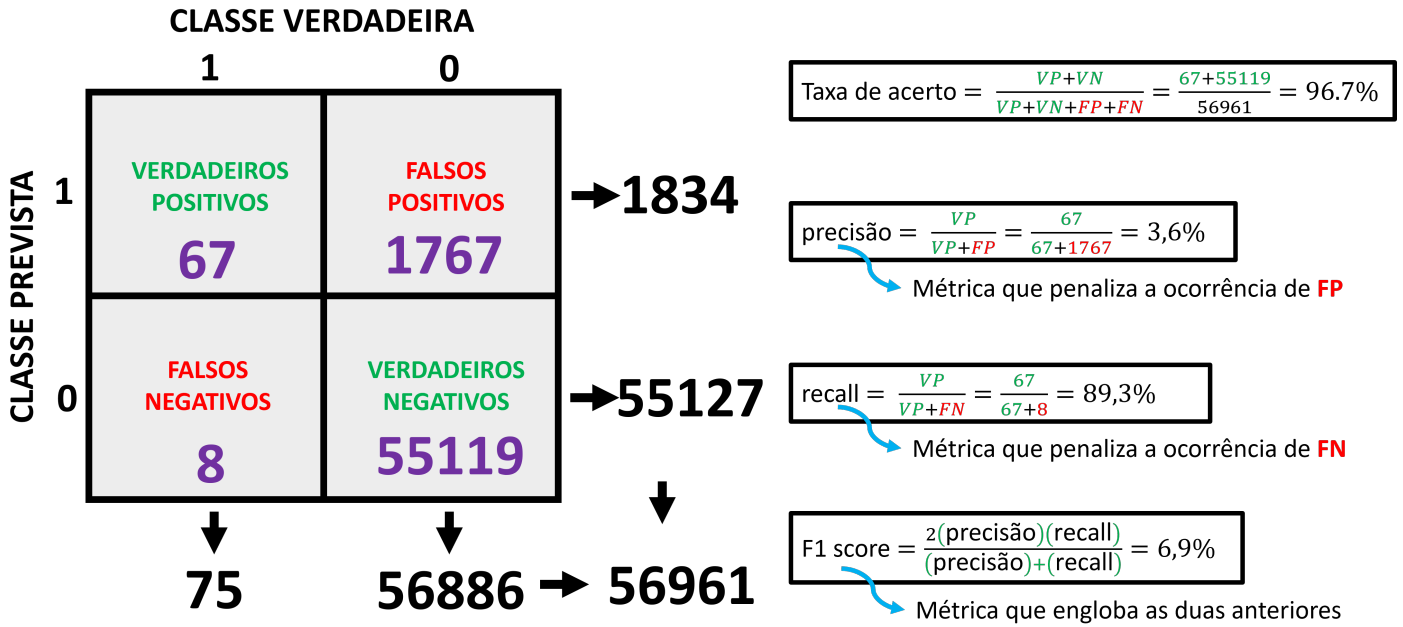
Ou seja, acabamos de perceber que o modelo treinado é adequado à aplicação em tela (identificação de transações possivelmente fraudulentas). Entretanto, ao calcularmos as métricas precisão, recall e F1 score, chegamos nos seguintes resultados numéricos:

Ou seja, acabamos de perceber que o modelo treinado é adequado à aplicação em tela (identificação de transações possivelmente fraudulentas). Entretanto, ao calcularmos as métricas precisão, recall e F1 score, chegamos nos seguintes resultados numéricos:



Conclusões:

Ou seja, acabamos de perceber que o modelo treinado é adequado à aplicação em tela (identificação de transações possivelmente fraudulentas). Entretanto, ao calcularmos as métricas precisão, recall e F1 score, chegamos nos seguintes resultados numéricos:



Conclusões:

- Busque por boas métricas, mas use-as com cautela.
- Nunca se desconecte do problema concreto (real) que está sendo investigado.

Importante!

- Quando temos mais rótulos do tipo $y = 1$, podemos, no momento do treinamento do modelo, dar um peso maior aos rótulos do tipo $y = 1$ em comparação com os rótulos do tipo $y = 0$.
- Isso fará com que a função custo penalize mais intensamente um erro de previsão relacionado ao rótulo $y = 1$ em comparação com um erro de previsão relacionado ao rótulo $y = 0$.
- Veremos como fazer isso nos próximos slides.

Dados desbalanceados: atribuindo pesos diferentes para cada classe

Dados desbalanceados: atribuindo pesos diferentes aos rótulos

Opção 1:

$$\text{peso classe } 0 = \frac{1}{\text{quantidade de amostras com } y = 0}$$

$$\text{peso classe } 1 = \frac{1}{\text{quantidade de amostras com } y = 1}$$

Por exemplo, se temos 100 amostras com rótulo $y = 0$ e 50 com rótulo $y = 1$, então

$$\text{peso classe } 0 = \frac{1}{100} = 0.01$$

$$\text{peso classe } 1 = \frac{1}{50} = 0.02$$

Dados desbalanceados: atribuindo pesos diferentes aos rótulos

Opção 1:

$$\text{peso classe } 0 = \frac{1}{\text{quantidade de amostras com } y = 0}$$

$$\text{peso classe } 1 = \frac{1}{\text{quantidade de amostras com } y = 1}$$

Por exemplo, se temos 100 amostras com rótulo $y = 0$ e 50 com rótulo $y = 1$, então

$$\text{peso classe } 0 = \frac{1}{100} = 0.01$$

$$\text{peso classe } 1 = \frac{1}{50} = 0.02$$

Opção 2:

$$\text{peso classe } 0 = \frac{1}{\text{quantidade de amostras com } y = 0} \times \frac{\text{total de amostras}}{2}$$

$$\text{peso classe } 1 = \frac{1}{\text{quantidade de amostras com } y = 1} \times \frac{\text{total de amostras}}{2}$$

Para o mesmo exemplo, teríamos

$$\text{peso classe } 0 = \frac{1}{100} \times \frac{150}{2} = 0.75$$

$$\text{peso classe } 1 = \frac{1}{50} \times \frac{150}{2} = 1.5$$

De olho no código!

De olho no código!

Vamos agora ver um caso com dados reais, de transações fraudulentas envolvendo cartões de crédito. Os dados encontram-se fortemente desbalanceados e, por este motivo, faremos uma atribuição de pesos diferentes para cada classe.

Acesse o Python Notebook usando o QR code ou o link abaixo:

`https://colab.research.google.com/github/xaximpvp2/master/blob/main/codigo_aula22_classificacao_desbalanceada.ipynb`

- Link para acesso à base de dados: `https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud`
- Acessando essa base de dados, note que não é possível saber o significado real de cada característica do problema.
- Isso porquê uma técnica do tipo PCA (Principal Component Analysis) foi utilizada para reduzir a dimensionalidade dos dados e abstrair seu sentido original (questões de sigilo).
- PCA é uma técnica de aprendizado não supervisionado usada para reduzir a dimensionalidade de um conjunto de dados, buscando manter as características (informações) principais presentes nele. A ideia central do PCA é transformar um grande número de variáveis correlacionadas em um número menor de variáveis não correlacionadas chamadas de componentes principais.

Atividade de aula

Parte 1

Rode todo o código. Responda às questões nele contidas e complete-o, se necessário.

Parte 2

1) Interprete adequadamente os resultados obtidos pela matriz de confusão.
2) Descreva qual procedimento você poderia adotar visando reduzir a quantidade de falsos positivos gerados pelo modelo.