ETAPAS DE PRÉ-PROCESSAMENTO

BALANCEAMENTO DA BASE DE DADOS

Cristiane Neri Nobre

Dados desbalanceados

- O problema de dados desbalanceados é tópico da área de **classificação de dados.**
- Em vários conjuntos de dados reais, o número de objetos **varia para diferentes classes**.
- Isso é comum em aplicações em que dados de um subconjunto das classes aparecem com uma **frequência maior que os dados das demais classes**.

Dados desbalanceados

Imagine a seguinte situação:

- Suponha que 80% das pessoas que estão indo ao hospital hoje para verificar se estão com o COVID-19 (do inglês Coronavirus Disease 2019) dê NEGATIVO.

- Neste contexto, 20% dos usuários testados estariam com o COVID-19.

- Assim, a classe com pacientes saudáveis é a classe **majoritária** e a classe de pacientes não saudáveis (com COVID-19) é a classe **minoritária**.

Dados desbalanceados

- Vários algoritmos de **Aprendizado de Máquina** têm seu desempenho prejudicado na presença de **dados desbalanceados**.

- Quando alimentados com **dados desbalanceados**, esses algoritmos tendem a favorecer a classificação de novos dados na **classe majoritária**.

Dados desbalanceados
O que fazer?

As principais técnicas para resolver este problema são:

- Redefinir o tamanho do conjunto de dados
- Utilizar diferentes custos de classificação para as diferentes classes
- Induzir um modelo para uma classe

Dados desbalanceados
O que fazer?

- **Redefinir o tamanho do conjunto de dados**

  - Neste caso, podemos tanto adicionar instância à **classe minoritária** (métodos **oversampling**) quanto remover instâncias da **classe majoritária** (métodos **undersampling**)

Problema com os métodos Oversampling:

- Existe o risco de as instâncias acrescentadas representarem situações em nunca ocorrerão, induzindo um modelo inadequado para os dados.
- Pode ocorrer também o problema de ***overfitting***, em que o modelo é superajustado aos dados de treinamento

Problema com os métodos Undersampling:

- Quando os dados são eliminados da classe majoritária, é possível que dados de grande importância para a indução do modelo correto sejam perdidos.

- Isso pode levar ao problema de ***underfitting***, em que o modelo induzido não se ajusta aos dados de treinamento.

Dados desbalanceados
O que fazer?

- **Utilizar diferentes custos de classificação para as diferentes classes**

  - A utilização de custos de classificação diferentes para as classes majoritária e minoritária tem como **dificuldade a definição desses custos**.

  - Por exemplo, se o número de exemplos da classe majoritária for o dobro do número da exemplos da classe minoritária, um erro de classificação para um exemplo da classe minoritária pode equivaler à ocorrência de dois erros de classificação para um exemplo da classe majoritária.

  - Entretanto, a definição dos diferentes custos geralmente não é tão direta.

Dados desbalanceados
O que fazer?

- **Induzir um modelo para uma classe**
  - A classe minoritária ou majoritária são aprendidas separadamente.
  - Neste caso, pode ser utilizado algoritmo de classificação para uma classe apenas.

Dados desbalanceados

Como realizar o balanceamento das classes no Python?

1. Com o método **SMOTE** (oversamplig)

```
pip install imbalanced-learn
from imblearn.over_sampling import SMOTE

base = pd.read_csv('/content/sample_data/cancer.csv', sep=';')
#processa base
Incluir aqui todas as etapas de pré-processamento

#Dividir em conjunto de treino e teste
X_treino, X_teste, y_treino, y_teste = train_test_split(X_prev, y_classe, test_size = 0.20, random_state = 42)

#Balanceamento com qualquer método oversampling
smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(X_treino, y_treino)

#agora é só treinar o modelo com estes arquivos gerados com o Smote e depois testar com o arquivo de teste (que está desbalanceado)
```

Dados desbalanceados

Como realizar o balanceamento das classes no Python?

2. Com o método **TomekLinks** (Undersampling)

```
pip install imbalanced-learn
from imblearn.under_sampling import TomekLinks

base = pd.read_csv('/content/sample_data/cancer.csv', sep=';')
#processa base
Incluir aqui todas as etapas de pré-processamento

#Dividir em conjunto de treino e teste
X_treino, X_teste, y_treino, y_teste = train_test_split(X_prev, y_classe, test_size = 0.20, random_state = 42)

#Balanceamento com qualquer método oversampling
balanceamento_under = TomekLinks(sampling_strategy='auto')
X_under, y_under = balanceamento_under.fit_resample(X_treino, y_treino)

#agora é só treinar o modelo com estes arquivos gerados com o TomeLinks e depois testar com o arquivo de teste (que está desbalanceado)
```

Dados desbalanceados

Como realizar o balanceamento das classes no Python?

2. Com o método **RandomUnderSampler** (Undersampling)

```
pip install imbalanced-learn
from imblearn.under_sampling import RandomUnderSampler

base = pd.read_csv('/content/sample_data/cancer.csv', sep=';')
#processa base
Incluir aqui todas as etapas de pré-processamento

#Dividir em conjunto de treino e teste
X_treino, X_teste, y_treino, y_teste = train_test_split(X_prev, y_classe, test_size = 0.20, random_state = 42)

#Balanceamento com qualquer método oversampling
undersample = RandomUnderSampler(random_state=42)
X_resampled, y_resampled = undersample.fit_resample(X_treino, y_treino)

#agora é só treinar o modelo com estes arquivos gerados com o RandomUnderSampler e depois testar com o arquivo de teste (que está desbalanceado)
```

Dados desbalanceados

Como realizar o balanceamento das classes no Python?

Importante:

- Se você usou um método **undersampling**, é importante colocar no teste as instâncias 'descartadas' pelo método

Referências:

- Capítulo 3 do livro (Seção 3.4)

- Katti Faceli et al. Inteligência Artificial, Uma abordagem de Aprendizado de Máquina, LTC, 2015.

Artigo:

- https://dl.acm.org/doi/10.1145/1007730.1007735