Seleção de Características

Aprendizado de Máquina

elias.rodrigues@paulista.ifpe.edu.br

Dimensionalidade do problema

Representação matricial da base de dados

- Soluções de AM manipulam dados como matrizes
  - Linha - padrão
  - Coluna - variável do problema (dimensão)

$$B \in \mathbb{R}^{N \times d}$$

- Base de dados “B”
- Possui “N” padrões
- Cada padrão é representado por “d” variáveis

Cenários de elevada dimensionalidade

- Soluções são mais dificilmente obtidas
- Tempo de execução é significativamente maior
- Não é possível visualizar a dispersão dos padrões

Maldição da dimensionalidade

“O erro da inferência aumenta com o aumento da dimensionalidade do problema”

- Requer um exponencialmente maior número de padrões para manter um patamar de inferência

Redução de dimensionalidade

Implica em encontrar uma nova forma de representar a base de dados cuja dimensionalidade seja menor

$$B \equiv \hat{B}$$

$$B \in \mathbb{R}^{N \times d}$$

$$\hat{B} \in \mathbb{R}^{N \times \hat{d}}$$

$$\hat{d} \ll d$$

- Seleção de características
- Extração de características

Redução de dimensionalidade

Seleção de características

- Descarta dimensões consideradas inúteis
- Não modifica as dimensões remanescentes
- Não cria novas dimensões
- É possível compreender porque da decisão de manutenção ou descarte
- Resultante de método de otimização

Extração de características

- Cria um novo conjunto de dimensões
  - A partir da combinação das dimensões originais
- Não é possível compreender facilmente a composição das novas dimensões
- Resultante de uma transformação matemática

Seleção de características: Scikit Learn

Scikit Learn: Feature Selection

1. Removendo características com baixa variância
2. Seleção univariada de características
3. Eliminação recursiva de características
4. Seleção de características usando SelectFromModel
5. Seleção de Característica Sequencial

1. Removendo características com baixa variância

- Remover dos dados as dimensões com valores de variância menor ou igual a limiar
- Usuário precisa escolher o limiar

$$\sigma^2 = \frac{\sum_{i=1}^{N} (x_i - \bar{x})^2}{N}$$

	x_1	x_2	x_3	x_4
	1	2	3	5
	1	3	2	2
	1	1	3	0
	1	2	2	3
σ^2	0	0,5	0,25	3,25

1. Removendo características com baixa variância

```
from sklearn.feature_selection import VarianceThreshold

X = [[1,2,3,5], [1,3,2,2], [1,1,3,0], [1,2,2,3]]

sel = VarianceThreshold( threshold=0.4999 )
sel.fit_transform(X)
```

```
array([[2, 5],
       [3, 2],
       [1, 0],
       [2, 3]])
```

2. Seleção univariada de características

- Seleciona as dimensões que melhor correspondem a um teste estatístico
- Analisa cada dimensão individualmente e atribui valor de adequação ao teste
- O valor retornado indica quão significante é o relacionamento entre a dimensão e a saída esperada

- Scikit Learn:
  - SelectKBest
  - SelectPercentile
  - SelectFpr
  - SelectFdr
  - SelectFwe
  - GenericUnivariateSelect

2. Seleção univariada de características

SelectKBest

- Usuário define o valor “k” de dimensões a serem mantidas
- As “d-k” dimensões com menores valores do teste estatístico são descartadas

Features	Regression	Classification
Continuous	Linear Regression F-test f_regression	ANOVA F-test f_classif
Categorical		Chi-squared test chi2

2. Seleção univariada de características

SelectKBest

```
from sklearn.datasets import load_iris
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2
X, y = load_iris(return_X_y=True)
X.shape
```

```
(150, 4)
```

```
X_new = SelectKBest(chi2, k=2).fit_transform(X, y)
X_new.shape
```

```
(150, 2)
```

3. Eliminação recursiva de características

- Busca por subconjunto das dimensões que melhor responde a um estimador
- São realizadas muitas avaliações de subconjunto das dimensões
- Processo pode ser lento caso hajam muitas dimensões
- O usuário define a quantidade final de dimensões

- O estimador é treinado com o conjunto original de dimensões
- A importância de cada dimensão é calculada pelo estimador
- A dimensão de menor importância é descartada
- Recursivamente o procedimento é repetido para as dimensões restantes
- A recursão é interrompida quando se alcança um determinado número de dimensões

3. Eliminação recursiva de características

```
from sklearn.datasets import load_digits

# Load the digits dataset
digits = load_digits()
X = digits.images.reshape((len(digits.images), -1))
y = digits.target

X.shape
```

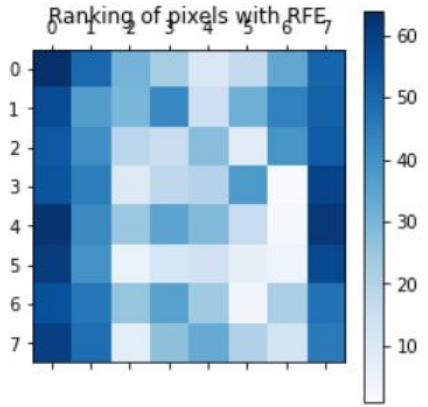
```
(1797, 64)
```

3. Eliminação recursiva de características

```
from sklearn.svm import SVC
from sklearn.feature_selection import RFE

# Create the RFE object and rank each pixel
svc = SVC(kernel="linear", C=1)
rfe = RFE(estimator=svc, n_features_to_select=1, step=1)
rfe.fit(X, y)

rfe.ranking_
```




```
array([64, 50, 31, 23, 10, 17, 34, 51, 57, 37, 30, 43, 14, 32, 44, 52, 54,
       41, 19, 15, 28,  8, 39, 53, 55, 45,  9, 18, 20, 38,  1, 59, 63, 42,
       25, 35, 29, 16,  2, 62, 61, 40,  5, 11, 13,  6,  4, 58, 56, 47, 26,
       36, 24,  3, 22, 48, 60, 49,  7, 27, 33, 21, 12, 46])
```

Examine a documentação

4. Seleção de características usando SelectFromModel

https://scikit-learn.org/stable/modules/feature_selection.html#feature-selection-using-selectfrommodel

5. Seleção de Característica Sequencial

https://scikit-learn.org/stable/modules/feature_selection.html#sequential-feature-selection

Seg às 09:00

https://meet.google.com/ngn-vjwh-qhd

Dúvidas: elias.rodrigues@paulista.ifpe.edu.br