

Programa de Pós-graduação em Sistemas de Informação

SIN5007 - Reconhecimento de Padrões (2023)

Atividade 2: One-Hot Encode, normalização e PCA

MSc. Leonardo Cunha dos Santos
Gabriel Francisco dos Santos Silva

São Paulo / 2023



Agenda

- ❶ Conjunto de dados
- ❷ One-hot encoding
- ❸ StandardScaler
- ❹ PCA

Conjunto de dados

Versão depura

```
RangeIndex: 19158 entries, 0 to 19157
```

```
Data columns (total 17 columns):
```

| # | Column | Non-Null Count | Dtype |
|---|-----------------------------|----------------|--------|
| 0 | NO_REGIAO | 19158 non-null | object |
| 1 | NO_UF | 19158 non-null | object |
| 2 | IN_CAPITAL_DEPARA | 19158 non-null | object |
| 3 | NO_CURSO_DEPARA | 19158 non-null | object |
| 4 | TP_GRAU_ACADEMICO_DEPARA | 19155 non-null | object |
| 5 | IN_GRATUITO_DEPARA | 19158 non-null | object |
| 6 | TP_MODALIDADE_ENSINO_DEPARA | 19158 non-null | object |
| 7 | TP_NIVEL_ACADEMICO_DEPARA | 19158 non-null | object |

| | | | |
|----|------------------------------------|----------------|--------|
| 8 | TP_DIMENSAO_DEPARA | 19158 non-null | object |
| 9 | TP_ORGANIZACAO_ACADEMICA_DEPARA | 19158 non-null | object |
| 10 | TP_CATEGORIA_ADMINISTRATIVA_DEPARA | 19158 non-null | object |
| 11 | TP_REDE_DEPARA | 19158 non-null | object |
| 12 | QT_VG_TOTAL | 19158 non-null | int64 |
| 13 | QT_INSCRITO_TOTAL | 19158 non-null | int64 |
| 14 | QT_ING | 19158 non-null | int64 |
| 15 | QT_MAT | 19158 non-null | int64 |
| 16 | QT_CONC | 19158 non-null | int64 |

```
dtypes: int64(5), object(12)
```

```
memory usage: 2.5+ MB
```

One-hot encoding

Utilização do pacote Pandas: get_dummies

```
categories = ['NO_REGIAO', 'NO_UF', 'IN_CAPITAL_DEPARA', 'NO_CURSO_DEPARA',  
             'TP_GRAU_ACADEMICO_DEPARA', 'IN_GRATUITO_DEPARA', 'TP_MODALIDADE_ENSINO_DEPARA',  
             'TP_NIVEL_ACADEMICO_DEPARA', 'TP_DIMENSAO_DEPARA', 'TP_ORGANIZACAO_ACADEMICA_DEPARA',  
             'TP_CATEGORIA_ADMINISTRATIVA_DEPARA']  
  
encoded_data = pd.get_dummies(df[categories], prefix=categories, prefix_sep='_')  
encoded_data = encoded_data.apply(lambda x: x.astype(bool).astype(int))  
  
encoded_data = encoded_data.merge(df[['QT_VG_TOTAL', 'QT_INSCRITO_TOTAL', 'QT_ING',  
                                     'QT_MAT', 'QT_CONC', 'TP_REDE_DEPARA']],  
                                left_index=True, right_index=True, how='left')
```

One-hot encoding

Result set

```
##### One hot encode #####  
      NO_REGIAO_Centro-Oeste  NO_REGIAO_Nordeste  ...  QT_CONC  TP_REDE_DEPARA  
0                0                0  ...      0      Pública  
1                0                0  ...      0      Pública  
2                0                0  ...      0      Pública  
3                0                0  ...      0      Pública  
4                0                0  ...      0      Privada  
...                ...                ...  ...      ...      ...  
19153            0                0  ...      0      Privada  
19154            0                0  ...      0      Privada  
19155            0                0  ...      1      Privada  
19156            0                0  ...      0      Privada  
19157            0                0  ...      0      Privada  
  
[19158 rows x 75 columns]
```

StandardScaler

Utilização do pacote StandardScaler do Sklearn

```
#####  
print(f' Normalização de variáveis '.center( __width: 80 __fillchar: '#'))  
  
scaler = StandardScaler()  
  
columns=['QT_VG_TOTAL', 'QT_INSCRITO_TOTAL', 'QT_ING', 'QT_MAT', 'QT_CONC']  
  
normalized_data = scaler.fit_transform(encoded_data[columns])  
normalized_data = pd.DataFrame(normalized_data, columns=columns)  
print(normalized_data)
```

StandardScaler

Result set

```
##### Normalização de variáveis #####
```

| | QT_VG_TOTAL | QT_INSCRITO_TOTAL | QT_ING | QT_MAT | QT_CONC |
|-------|-------------|-------------------|-----------|-----------|----------|
| 0 | 0.724341 | 3.745327 | -0.269880 | -0.287079 | -0.20961 |
| 1 | -0.056820 | -0.077259 | -0.269880 | -0.287079 | -0.20961 |
| 2 | -0.056820 | -0.077259 | -0.269880 | -0.287079 | -0.20961 |
| 3 | -0.056820 | -0.077259 | -0.269880 | -0.287079 | -0.20961 |
| 4 | 12.420536 | 12.534500 | -0.269880 | -0.287079 | -0.20961 |
| ... | ... | ... | ... | ... | ... |
| 19153 | -0.056820 | -0.077259 | -0.243467 | -0.272439 | -0.20961 |
| 19154 | -0.056820 | -0.077259 | -0.164229 | -0.169958 | -0.20961 |
| 19155 | -0.056820 | -0.077259 | -0.111404 | -0.228518 | -0.11534 |
| 19156 | -0.056820 | -0.077259 | -0.243467 | -0.272439 | -0.20961 |
| 19157 | -0.056820 | -0.077259 | -0.243467 | -0.272439 | -0.20961 |

Dimensões do conjunto de dados

Após a padronização

- Variáveis qualitativas (binárias): 69
- Variáveis quantitativas (padronizadas): 5
- Alvo: TP_REDE_DEPARA
- Total de variáveis no result set: 75
- Instâncias: 19158

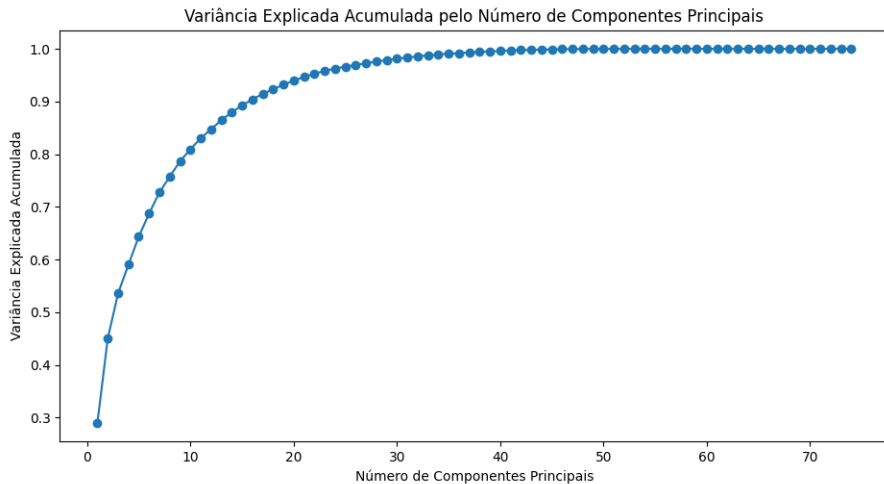
PCA

Principal Component Analysis

```
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

data_X = normalized_data_vf.drop('TP_REDE_DEPARA', axis=1)

pca = PCA()
pca.fit(data_X)
explained_variance_ratio = pca.explained_variance_ratio_
```



Número de Componentes Principais para 95% de Variância: 22

PCA com número de componentes esperado

| | PC1 | PC2 | PC3 | ... | PC20 | PC21 | PC22 |
|-------|-----------|-----------|-----------|-----|-----------|-----------|-----------|
| 0 | 0.190351 | 3.345125 | -0.140522 | ... | -1.051111 | -1.318289 | -0.026662 |
| 1 | -0.271789 | 0.161091 | -0.924383 | ... | -1.180369 | -1.477429 | -0.046666 |
| 2 | -0.309983 | 0.141671 | 0.358834 | ... | -1.032111 | -1.547045 | -0.003615 |
| 3 | -0.309983 | 0.141671 | 0.358834 | ... | -1.032111 | -1.547045 | -0.003615 |
| 4 | 1.720391 | 17.651678 | -0.612151 | ... | -0.292294 | -0.570287 | -0.030889 |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 19153 | -0.535000 | -0.050070 | 0.392255 | ... | -0.039530 | -0.041791 | 0.950386 |
| 19154 | -0.439304 | -0.069639 | 0.916337 | ... | 0.018221 | -0.060235 | 0.949860 |
| 19155 | -0.392884 | -0.077730 | 0.929753 | ... | -0.017430 | -0.033971 | 0.951916 |
| 19156 | -0.447089 | -0.046038 | 0.713141 | ... | 0.101151 | 0.011569 | 0.959502 |
| 19157 | -0.535000 | -0.050070 | 0.392255 | ... | -0.039530 | -0.041791 | 0.950386 |

[19158 rows x 22 columns]

Obrigado!

Thanks! / ¡Gracias!

Leonardo Cunha dos Santos

`lattes.cnpq.br/5620610314140397`

`leonardo.cunha.santos@usp.br`

Gabriel Francisco dos Santos Silva

`gabfssilva@gmail.com`