



Escola Politécnica de Pernambuco

Especialização em Ciência de Dados e Analytics

Introdução à Ciência de Dados

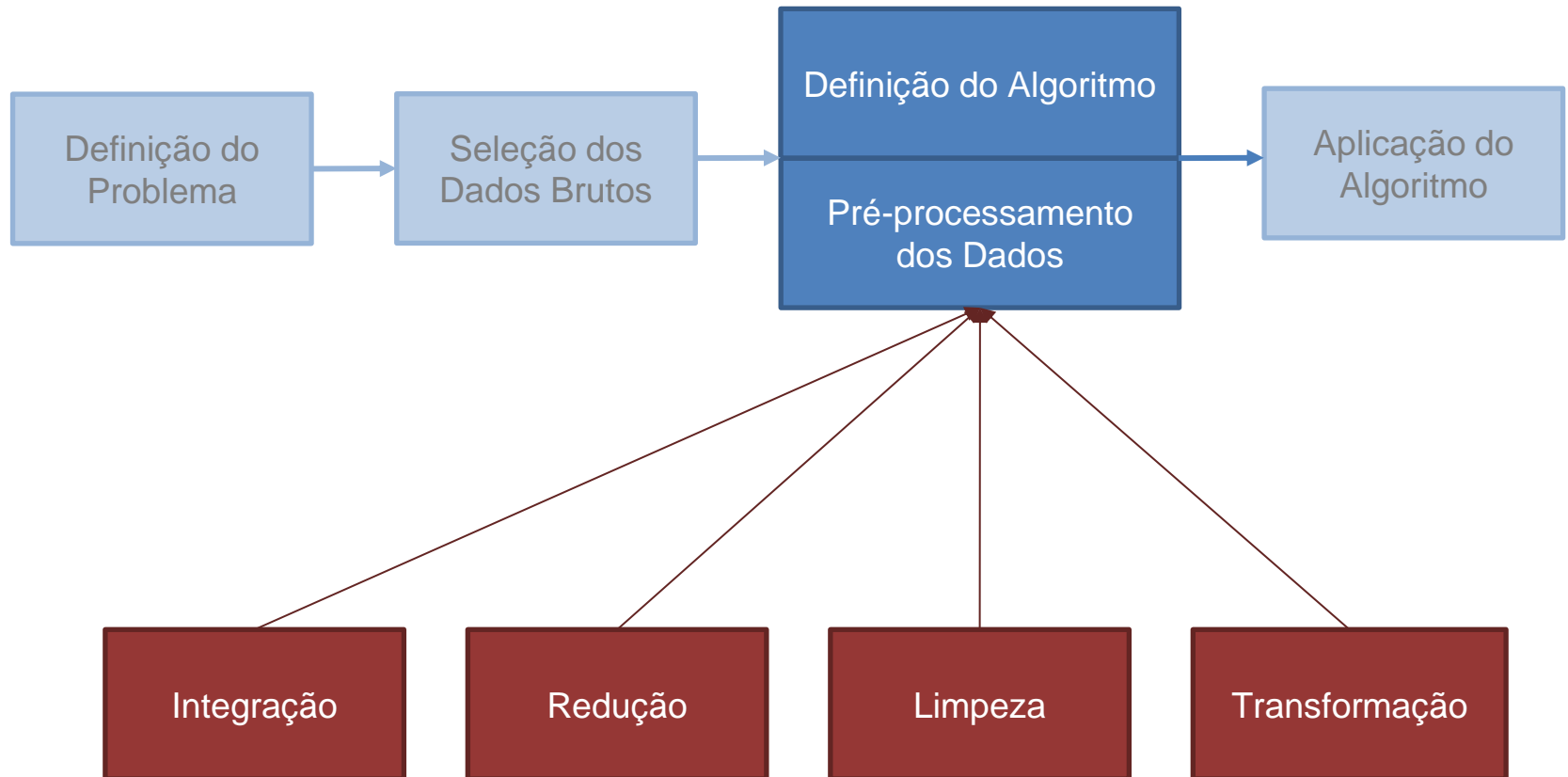
Aula 4

Prof. Dr. Alexandre Maciel
alexandre.maciel@upe.br


PRÉ-PROCESSAMENTO DOS DADOS

- Manipulação e transformação de dados brutos de modo que o conhecimento contidos neles possa ser mais fácil e corretamente obtido.
- Dados de mundo real obtidos por fontes automáticas, sensores, digitadores geralmente são incompletos, inconsistentes ou ruidosos.

ESQUEMA



BASE DE DADOS EXEMPLO



UC Irvine
Machine Learning
Repository


Datasets

Contribute Dataset

About

Search

Login



Mammographic Mass

Donated on 10/28/2007

Discrimination of benign and malignant mammographic masses based on BI-RADS attributes and the patient's age.

Dataset Characteristics

Multivariate

Subject Area

Life

Associated Tasks

Classification

Attribute Type

Integer

Instances

961

Attributes

6

Information

Additional Information

Mammography is the most effective method for breast cancer screening available today. However, the low positive predictive value of breast biopsy resulting from mammogram interpretation leads to approximately...

SHOW MORE

DOWNLOAD

CITE

3 citations

1877 views

Creators

Matthias Elter

DOI

[10.24432/C53K6Z](#)

License

This dataset is licensed under a [Creative Commons Attribution](#)

DESCRIÇÃO DA BASE

Number of Instances: 961

Number of Attributes: 6 (1 goal field, 1 non-predictive, 4 predictive attributes)

Attribute Information:

1. BI-RADS assessment: 1 to 5 (ordinal, non-predictive)
2. Age: patient's age in years (integer)
3. Shape: mass shape: round=1 oval=2 lobular=3 irregular=4 (nominal)
4. Margin: mass margin: circumscribed=1 microlobulated=2 obscured=3 ill-defined=4 spiculated=5 (nominal)
5. Density: mass density high=1 iso=2 low=3 fat-containing=4 (ordinal)
6. Severity: benign=0 or malignant=1 (binominal)

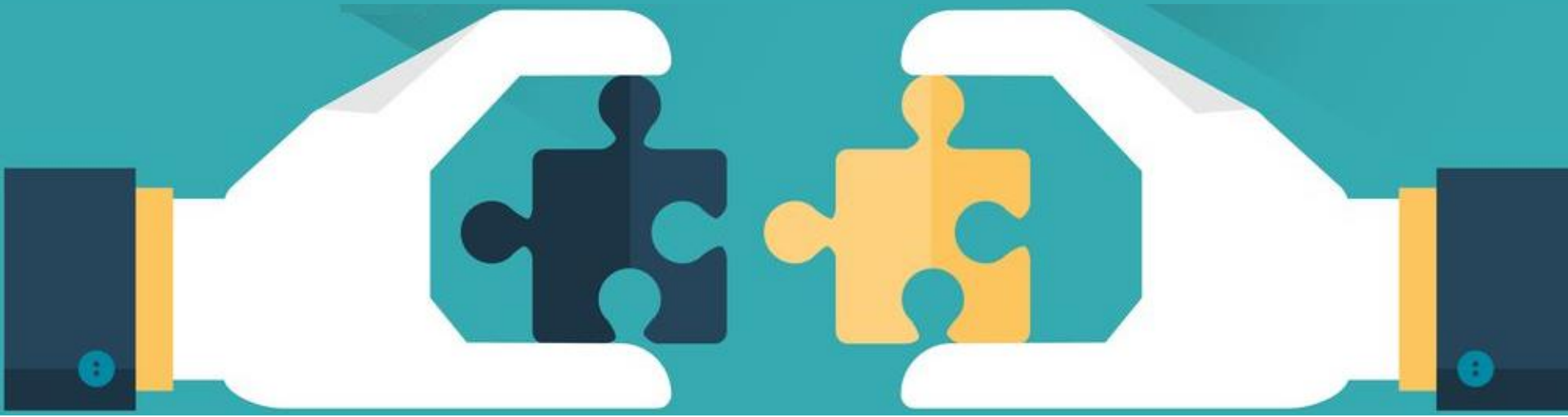
Missing Attribute Values: Yes

- BI-RADS assessment: 2
- Age: 5
- Shape: 31
- Margin: 48
- Density: 76
- Severity: 0

Class Distribution: benign: 516; malignant: 445

INTEGRAÇÃO DE DADOS

- **DUPLICIDADE** ... atributos aparecem repetidos na base.
- **CONFLITOS** ... para mesma entidade, diferentes valores na base.
- **GRANULARIDADE** ... tratamento para unidades maiores ou menores de dados.



DUPLICIDADE

```
df.value_counts()
```

BI-RADS	Idade	Forma	Margem	Densidade	Severidade	
5.0	66.0	4.0	4.0	3.0	1	9
4.0	45.0	1.0	1.0	3.0	0	6
	46.0	1.0	1.0	3.0	0	6
	54.0	1.0	1.0	3.0	0	6
	64.0	1.0	1.0	3.0	0	6
						..
	58.0	1.0	1.0	3.0	1	1
	57.0	4.0	4.0	4.0	1	1
				3.0	0	1
			3.0	3.0	1	1
55.0	46.0	4.0	3.0	3.0	1	1

Length: 564, dtype: int64



```
df_duplicatas = df.drop_duplicates()  
df_duplicatas
```

	BI-RADS	Idade	Forma	Margem	Densidade	Severidade
0	1.000000	0.628205	0.666667	1.00	0.666667	1.0
1	0.666667	0.320513	0.000000	0.00	NaN	1.0
2	1.000000	0.512821	1.000000	1.00	0.666667	1.0
3	0.666667	0.128205	0.000000	0.00	0.666667	0.0
4	1.000000	0.717949	0.000000	1.00	NaN	1.0
...
925	0.666667	0.666667	0.000000	0.00	0.000000	0.0
926	1.000000	0.666667	0.000000	0.75	0.666667	1.0
929	0.666667	0.448718	1.000000	1.00	0.666667	0.0
930	0.666667	0.461538	1.000000	0.75	0.666667	1.0
938	0.666667	0.487179	1.000000	1.00	0.666667	1.0

666 rows x 6 columns

REDUÇÃO DE DADOS

- **SEGMENTAÇÃO** dos dados.

Seleção de instâncias (filtros).

- **EXCLUSÃO** de atributos.

Redução de dimensionalidade.

- **AMOSTRAGEM** dos dados.

Técnica amplamente utilizada na estatística.



REDUÇÃO DE DADOS

```
df = df[df['BI-RADS'] > 0]
df = df[df['BI-RADS'] < 6]

df.shape

(942, 6) SEGMENTAÇÃO
```

```
df2 = df.drop(columns=['Severidade', 'Margem'])
df2
```

	BI-RADS	Idade	Forma	Densidade
0	5.0	67.0	3.0	3.0
1	4.0	43.0	1.0	NaN
2	5.0	58.0	4.0	3.0
3	4.0	28.0	1.0	3.0
4	5.0	74.0	1.0	NaN
...
956	4.0	47.0	2.0	3.0
957	4.0	56.0	4.0	3.0
958	4.0	64.0	4.0	3.0
959	5.0	66.0	4.0	3.0
960	4.0	62.0	3.0	3.0

942 rows x 4 columns

EXCLUSÃO

```
dfsample = df.sample(n=10, replace=False, random_state=123)
dfsample
```

	BI-RADS	Idade	Forma	Margem	Densidade	Severidade
783	4.0	69.0	2.0	1.0	3.0	1
349	4.0	45.0	1.0	2.0	3.0	0
564	5.0	79.0	1.0	4.0	3.0	1
593	5.0	53.0	4.0	5.0	3.0	0
276	5.0	70.0	4.0	5.0	3.0	1
543	4.0	45.0	2.0	1.0	3.0	1
162	4.0	23.0	3.0	1.0	3.0	0
139	5.0	67.0	3.0	5.0	3.0	1
341	5.0	61.0	1.0	1.0	3.0	1
319	4.0	64.0	4.0	4.0	3.0	1

AMOSTRAGEM

LIMPEZA DOS DADOS

- **Atribuir valores AUSENTES.**

Média, Mediana, Moda, Constante, Similaridade...

- **Suavizar RUÍDOS.**

Encaixotamento.

- **Identificar OUTLIERS.**

Análise “manual”.

- **Corrigir INCONSISTÊNCIAS.**

Análise automatizada.



DADOS AUSENTES

```
df_fill = df.fillna(df.mean())  
df_fill
```

	BI-RADS	Idade	Forma	Margem	Densidade	Severidade
0	5.0	67.0	3.0	5.0	3.000000	1
1	4.0	43.0	1.0	1.0	2.911085	1
2	5.0	58.0	4.0	5.0	3.000000	1
3	4.0	28.0	1.0	1.0	3.000000	0
4	5.0	74.0	1.0	5.0	2.911085	1
...
956	4.0	47.0	2.0	1.0	3.000000	0
957	4.0	56.0	4.0	5.0	3.000000	1
958	4.0	64.0	4.0	5.0	3.000000	0
959	5.0	66.0	4.0	5.0	3.000000	1
960	4.0	62.0	3.0	3.0	3.000000	0

942 rows x 6 columns

MÉDIA

```
df['Forma'] = df['Forma'].fillna(df['Forma'].mode()[0])  
df['Forma']
```

```
0    3.0  
1    1.0  
2    4.0  
3    1.0  
4    1.0  
...  
956   2.0  
957   4.0  
958   4.0  
959   4.0  
960   3.0  
Name: Forma, Length: 942, dtype: float64
```

MODA

TRANSFORMAÇÃO DOS DADOS

- **CODIFICAÇÃO**... Conversão categórico- numérico, numérico-categórico...
- **NORMALIZAÇÃO**... Ajustar escala de valores.
- **BALANCEAMENTO**... Ajustar quantidade de instâncias.
- **PARTIÇÃO**... Essencial para modelagem.



TRANSFORMAÇÃO

```
enc = OrdinalEncoder()  
enc.fit(df)  
  
encoded_data = enc.transform(df)  
  
df_encoded = pd.DataFrame(encoded_data, columns=df.columns)  
df_encoded
```

	BI-RADS	Idade	Forma	Margem	Densidade	Severidade
0	4.0	49.0	1.0	1.0	1.0	1.0
1	3.0	25.0	3.0	0.0	NaN	1.0
2	4.0	40.0	0.0	1.0	1.0	1.0
3	3.0	10.0	3.0	0.0	1.0	0.0
4	4.0	56.0	3.0	1.0	NaN	1.0

CODIFICAÇÃO

```
x = df_fill.values  
min_max_scaler = preprocessing.MinMaxScaler()  
x_scaled = min_max_scaler.fit_transform(x)  
df_normalized = pd.DataFrame(x_scaled, columns=df.columns)
```

```
df_normalized.head(5)
```

	BI-RADS	Idade	Forma	Margem	Densidade	Severidade
0	1.000000	0.628205	0.666667	1.0	0.666667	1.0
1	0.666667	0.320513	0.000000	0.0	0.666667	1.0
2	1.000000	0.512821	1.000000	1.0	0.666667	1.0
3	0.666667	0.128205	0.000000	0.0	0.666667	0.0
4	1.000000	0.717949	0.000000	1.0	0.666667	1.0

NORMALIZAÇÃO

```
x = df.drop(columns='Severidade')  
y = df['Severidade']  
  
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3)
```

PARTIÇÃO

ATIVIDADE 4

1. **Faça o pré-processamento de dados do seu projeto.**
 - Apresente os steps para redução, limpeza e transformação.
2. **Entregue os slides (+3) de sua apresentação.**
3. **Faça o upload no Google Classroom.**