



## AULA 3 - Análise Exploratória

**#BootcampMIA2022 #SomosMIA**



## Quem somos?



## Bárbara Barbosa

### Data Manager na Orderchamp

- Mestra em Sistemas de Informação - com foco em Inteligência Artificial e NLP
- Organizadora do Rails Girls SP, Women Dev Summit e Women in Data Science SP 2019/2020/2021



bahbbc



bahbbc



## Quem somos?



# Fernanda Wanderley

## Data Scientist na NeuralMed

- Co-fundadora da MIA
- Embaixadora do WiDS Rio
- Doutora em Inteligência Computacional - UFMG
- Google Developer Expert - ML



nandaw



nandaw

Agora acabaram as introduções...

**Finalmente vamos colocar a mão  
na massa!**





## Uma inspiração...

"Para mim Programação é mais do que uma arte prática importante. É também um empreendimento gigantesco nos fundamentos do conhecimento."

(Grace Hopper)

## O que veremos hoje:

- Ler, limpar e validar os dados
- Distribuições e estatísticas básicas
- Missing values e outliers
- Relações entre dados

# Ler, limpar e validar os dados - Dataframes

## Series

	apples
0	3
1	2
2	0
3	1

+

## Series

	oranges
0	0
1	3
2	7
3	2

=

## DataFrame

	apples	oranges
0	3	0
1	2	3
2	0	7
3	1	2

# Ler, limpar e validar os dados - colinha

Use this table for reference



## Python For Data Science Data Wrangling in Pandas Cheat Sheet

Learn Data Wrangling online at [www.DataCamp.com](https://www.datacamp.com)

### Reshaping Data

#### Pivot

```
pd.pivot(index='year', columns='country', values='pop')
pd.pivot(index='year', columns='country', values='pop')
```



#### Pivot Table

```
pd.pivot_table(index='year', columns='country', values='pop')
pd.pivot_table(index='year', columns='country', values='pop')
```

#### Stack / Unstack

```
df.stack()
df.unstack()
```



#### Melt

```
pd.melt(df, id_vars='year', value_vars=['pop', 'gdp'])
pd.melt(df, id_vars='year', value_vars=['pop', 'gdp'])
```



### Iteration

```
df.iterrows()
df.itertuples()
```

### Missing Data

```
df.isnull()
df.dropna()
df.fillna()
```

### Advanced Indexing

Also see NumPy Array

#### Selecting

```
df.loc[0:1, 'pop']
df.iloc[0:1, 0]
```

#### Indexing With iisla()

```
df.iisla('country', 'pop')
df.iisla('country', 'pop')
```

#### Where

```
df.where(df['pop'] > 1000000)
```

#### Query

```
df.query('pop > 1000000')
```

#### Setting/Resetting Index

```
df.set_index('year')
df.reset_index()
```

#### Reindexing

```
df.reindex(index=[1, 2, 3])
df.reindex(columns=['pop', 'gdp'])
```

#### Multindexing

```
df.index = pd.MultiIndex.from_tuples([('A', 1), ('A', 2), ('B', 1), ('B', 2)])
df.index.names = ['country', 'year']
```

### Duplicate Data

```
df.duplicated()
df.drop_duplicates()
```

### Grouping Data

#### Aggregation

```
df.groupby('year').sum()
df.groupby('year').mean()
```

#### Transformation

```
df.transform(lambda x: x**2)
```

### Combining Data



#### Merge

```
pd.merge(df1, df2, on='year')
pd.merge(df1, df2, left_on='year', right_on='year')
```

#### Join

```
df1.join(df2, how='outer')
```

#### Concatenate

##### Vertical

```
pd.concat([df1, df2])
```

##### Horizontal/Vertical

```
pd.concat([df1, df2], axis=1)
```

### Dates

```
df['year'] = pd.to_datetime(df['year'])
df['year'] = pd.to_datetime(df['year'])
```

### Visualization

Also see Matplotlib



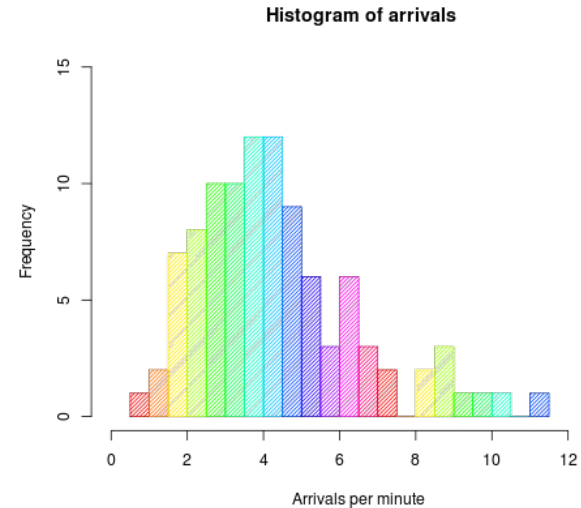


## Ler, limpar e validar os dados - tipos de variáveis

Pandas dtype	Python type	NumPy type	Usage
object	str	string_, unicode_	Text
int64	int	int_, int8, int16, int32, int64, uint8, uint16, uint32, uint64	Integer numbers
float64	float	float_, float16, float32, float64	Floating point numbers
bool	bool	bool_	True/False values
datetime64	NA	datetime64[ns]	Date and time values
timedelta[ns]	NA	NA	Differences between two datetimes
category	NA	NA	Finite list of text values

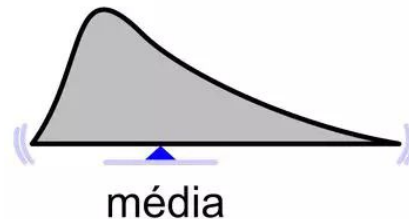
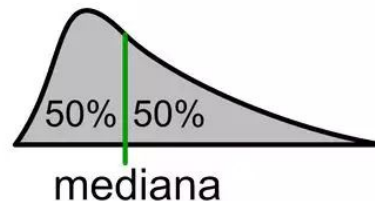
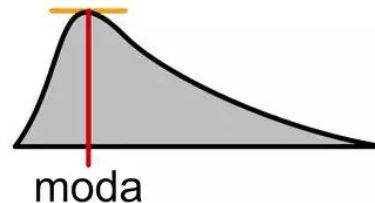
## Estatísticas Básicas - Distribuição de Dados

- É importante entender como os dados estão distribuídos para tirarmos informação deles
- Histogramas:** frequência com a qual os valores aparecem, divididos em *bins*



# Estatísticas Básicas - Medidas de tendência central

- **Moda:** valor mais frequente
- **Média:** valor resultante da soma de todos os valores dividido pela quantidade deles
- **Mediana:** valor que divide o conjunto dos dados em duas metades



## Estatísticas Básicas - Medidas de Dispersão

- **Range:** diferença entre o menor e maior valor na população
- **Variância:** distância de cada valor até a média
- **Desvio Padrão:** raiz quadrada da variância
- **Quartis:** valores que dividem a população em 4 intervalos iguais

## Normalização vs Padronização dos Dados

- Precisamos que os dados estejam todos na mesma escala
- **Padronização**: colocar os dados numa distribuição com média 0 e desvio padrão 1
- **Normalização**: colocar os dados no intervalo entre 0 e 1

## Missing values e outliers

O que podemos fazer com valores faltantes (missing values)?



## Substituindo valores faltantes

Delete os valores!

### Pros

- Você terá um modelo mais robusto

### Contras

- Muita perda de informação
- O modelo pode ficar terrível se houverem muitos valores faltantes

# Substituindo valores faltantes

Substitua pela média/mediana

## Pros

- Você não terá perda de dados
- Funciona bem com um dataset pequeno e é fácil de implementar

## Contras

- Só funciona com valores numéricos e contínuos
- Não leva em consideração a covariância dos fatores
- Pode acabar causando algum vazamento de informação (data leakage)



# Substituindo valores faltantes

Substitua pela moda ou crie uma nova categoria

## Pros

- Você não terá perda de dados
- Funciona bem com um dataset pequeno e é fácil de implementar
- Ao criar uma nova categoria você evita a perda de informação dos dados faltantes

## Contras

- Só funciona com valores categóricos
- Pode levar a baixa performance do modelo a depender do padrão dos valores faltantes

## Substituindo valores faltantes

Crie um modelo para isso!

### Pros

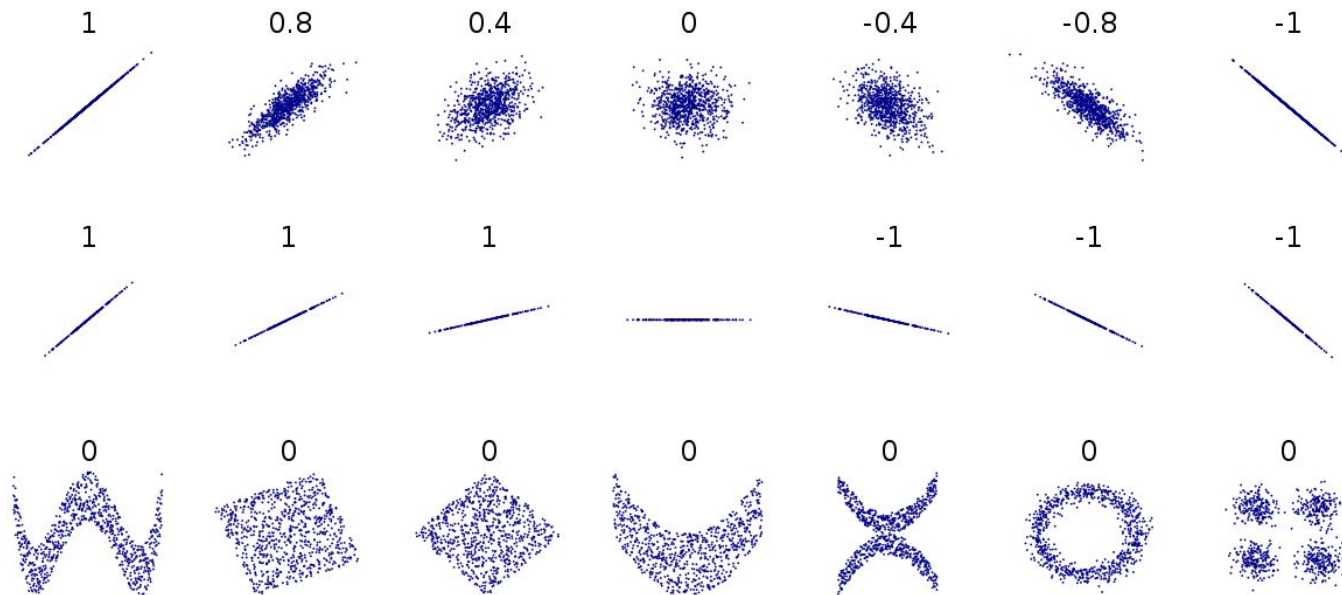
- Pode dar ótimos resultados
- Leva em consideração a covariância dos dados

### Contras

- Muito mais trabalhoso
- É um proxy para os valores verdadeiros

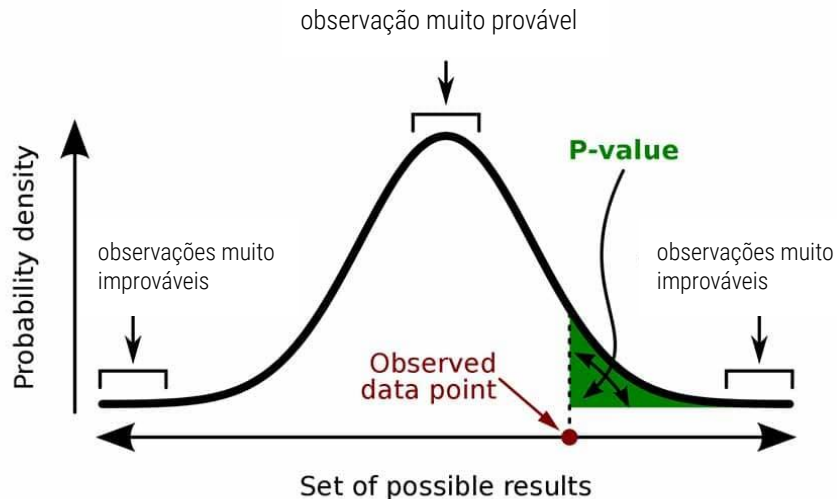
# Relações entre dados - Correlação

Correlação descreve como 2 variáveis se movem em conjunto.



# Relações entre dados - Chi quadrado e p-valor

O teste de chi-quadrado é um teste de independência para variáveis categóricas. Você pode testar uma ou mais variáveis.



**P-valor** (em verde) é a probabilidade do resultado observado assumir a hipótese nula

Este teste confronta a hipótese nula: Não há relação entre as variáveis.

A hipótese nula significa que é tudo acaso, que o que você está tentando provar não acontece.

O p-valor é a probabilidade de tudo ser obra do acaso. Quanto menor maior as chances de você descartar a hipótese nula (as variáveis têm uma relação).

0.05 é um valor com relevância estatística.

***Vamos praticar!!!***



# Para praticar...

- ▷ Boas EDAs no Kaggle:
  - Existe desigualdade de gênero em dados?
- ▷ Veja esse curso do Kaggle (em inglês, grátis)
- ▷ Esse curso do Udemy também é bastante interessante para se aprofundar
- ▷ Essa série de vídeos do Programação Dinâmica sobre análise de dados





## Nossos contatos



**mulheres.em.ia@gmail.com**



**mulheres-em-ia**



**@mulheres.em.ia**



**@mulheres.em.ia**



**@MulheresemInteligenciaArtificial**



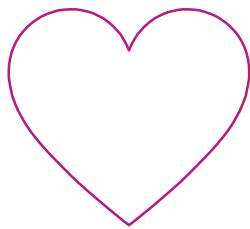
**Canal: Mulheres em IA**

## Linktree

<https://linktr.ee/mulheres.em.ia>

## Grupo Telegram para Mulheres

[https://t.me/mulheres\\_em\\_ia](https://t.me/mulheres_em_ia)



# Muito obrigada!

Dúvidas? Podem nos procurar! 🙄