


# IMD0033 - Probabilidade

## Aula 17 - Visualização estatística de dados com Seaborn

Ivanovitch Silva  
Outubro, 2018



# Agenda

---

- Estudo de caso: competição kaggle
- Introdução ao Seaborn
- Instalação
- Histogramas, KDE
- Personalizando gráficos
- Distribuições condicionais

# Atualizar o repositório

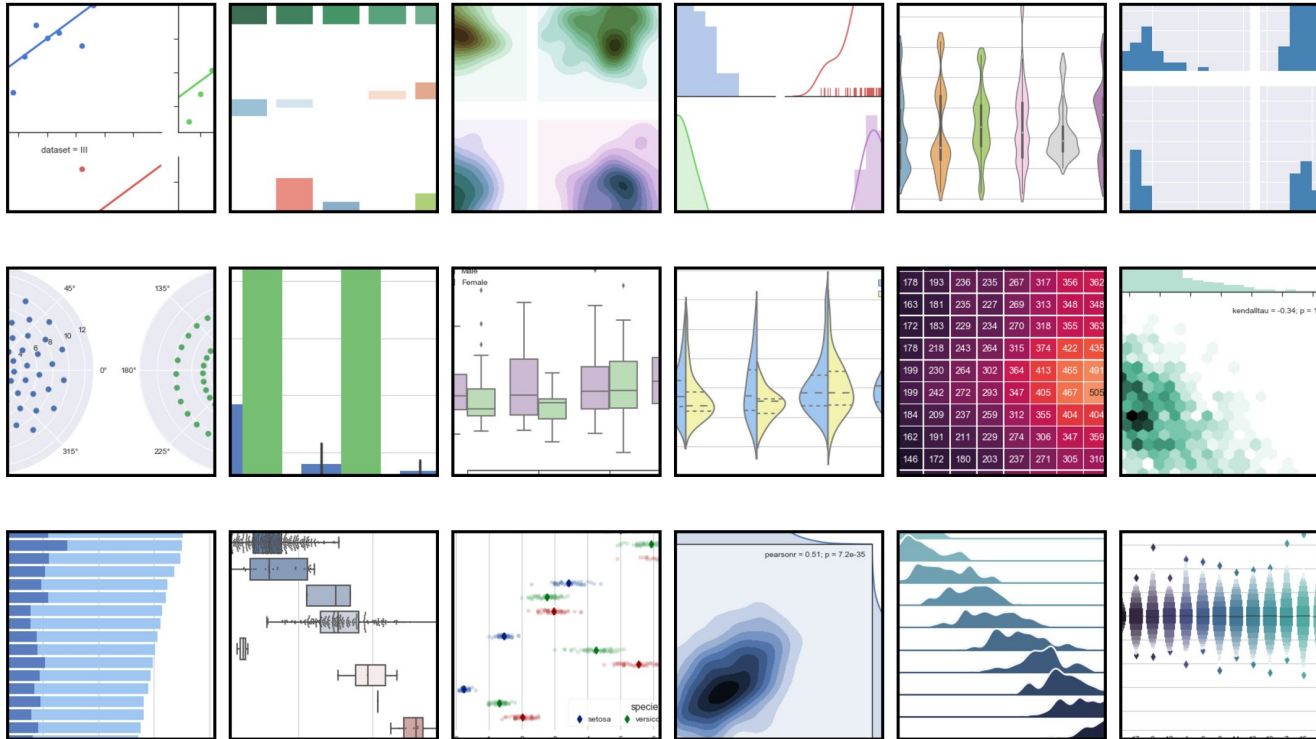
---

```
git clone https://github.com/ivanovitchm/imd0033_2018_2.git
```

Ou ....

```
git pull
```

# Motivação - Seaborn



# Introdução a base de dados

---



Getting Started Prediction Competition

## **Titanic: Machine Learning from Disaster**

Start here! Predict survival on the Titanic and get familiar with ML basics



Kaggle · 8,380 teams · 3 years to go

<https://www.kaggle.com/c/titanic/data>

# Introdução a base de dados

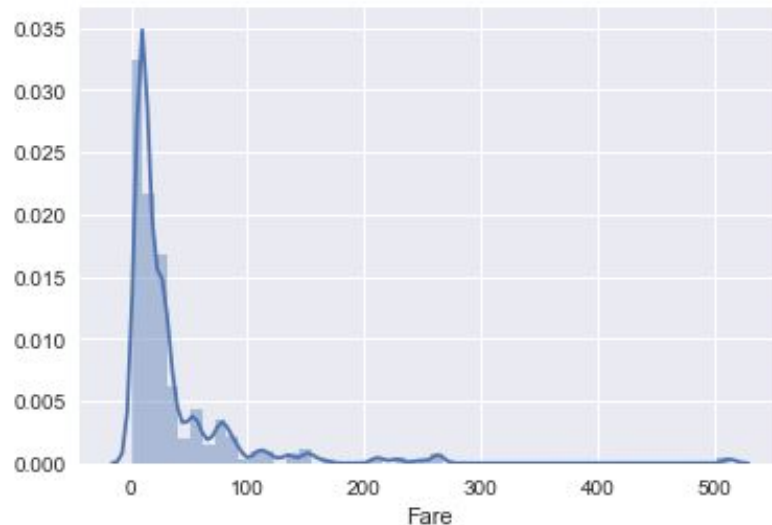
---

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500		S
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	female	38.0	1	0	PC 17599	71.2833	C85	C
3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250		S

# Criando um histograma com Seaborn

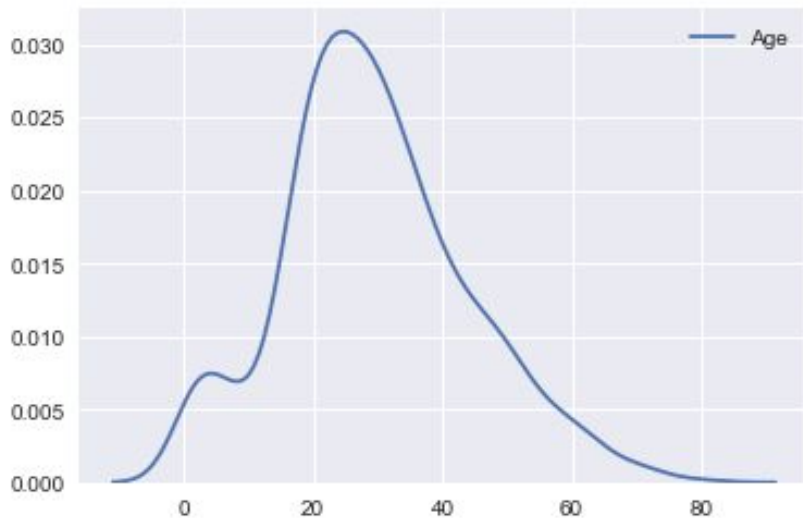
---

```
# seaborn is commonly imported as `sns`.  
import matplotlib.pyplot as plt  
import seaborn as sns  
  
#to switch to seaborn defaults, simply call the set()  
sns.set()  
  
# The four preset contexts, in order of relative size,  
sns.set_context("notebook")  
  
# plot a univariate distribution of observations.  
sns.distplot(titanic["Fare"])  
plt.show()
```



# Gerando uma distribuição KDE

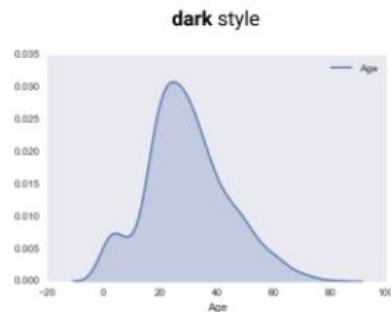
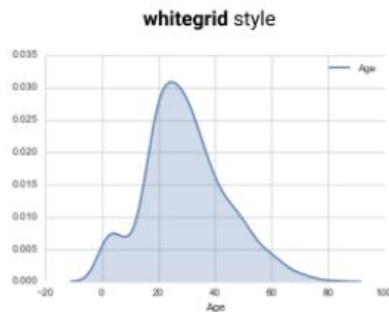
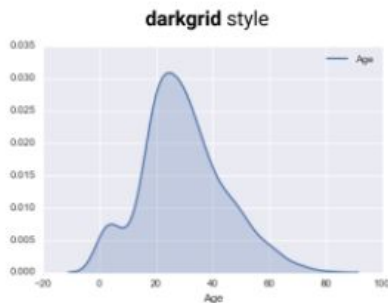
---



```
sns.kdeplot(titanic["Age"])  
plt.show()
```

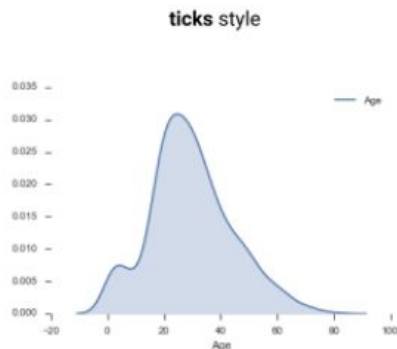
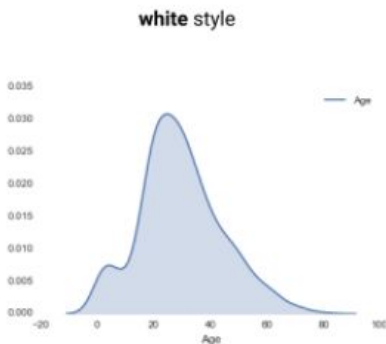


# Modificando a aparência da visualização



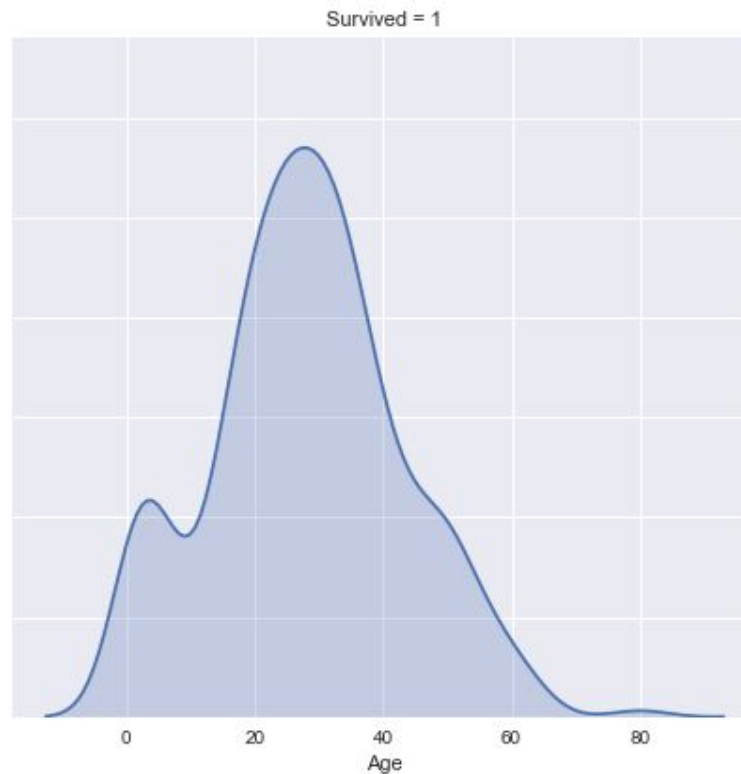
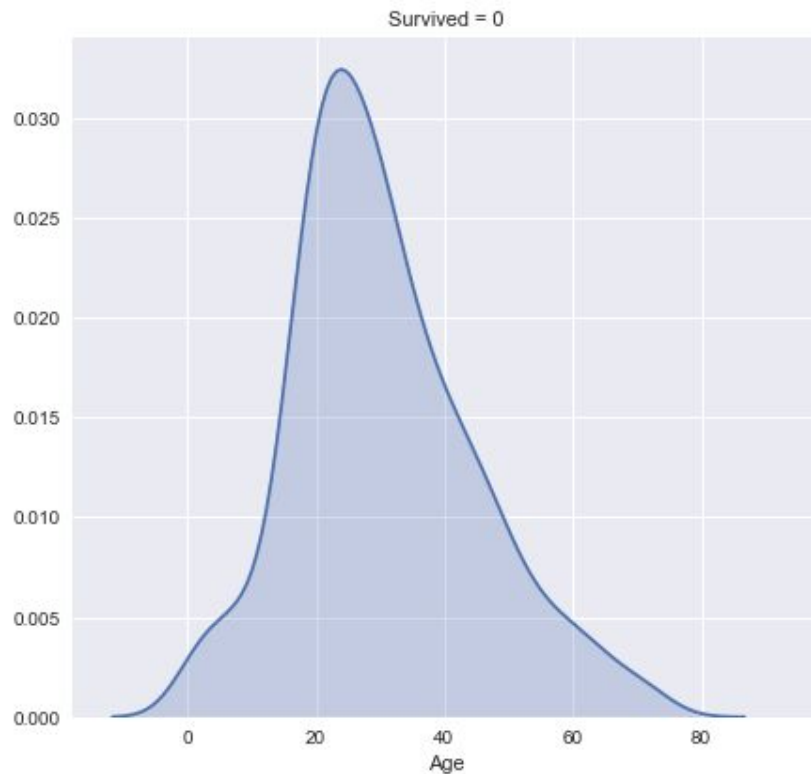
```
sns.set_style("darkgrid")
```

```
sns.despine()
```



# Distribuições condicionais

---



# Distribuições condicionais

---

```
# Condition on unique values of the "Survived" column.
```

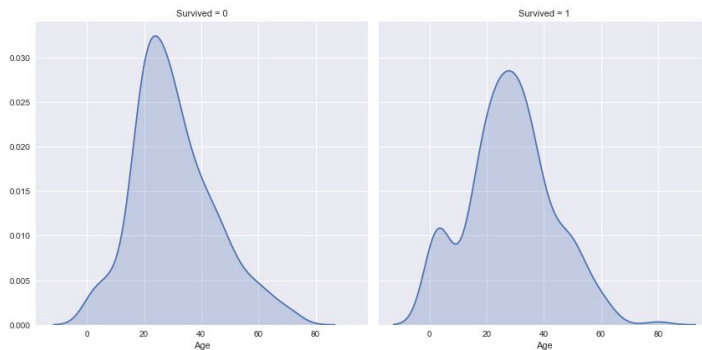
```
g = sns.FacetGrid(titanic, col="Survived", size=6)
```

```
# For each subset of values, generate a kernel density plot of the "Age" columns.
```

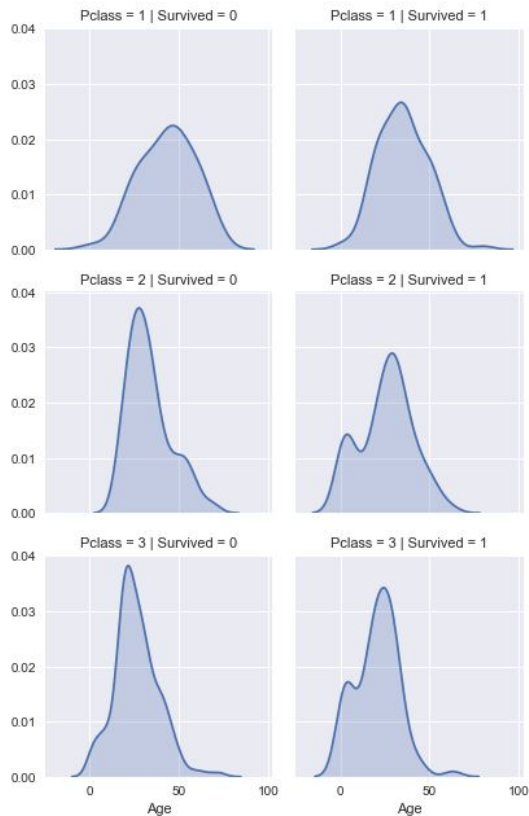
```
g.map(sns.kdeplot, "Age", shade=True)
```

```
# Plot the graph
```

```
plt.show()
```

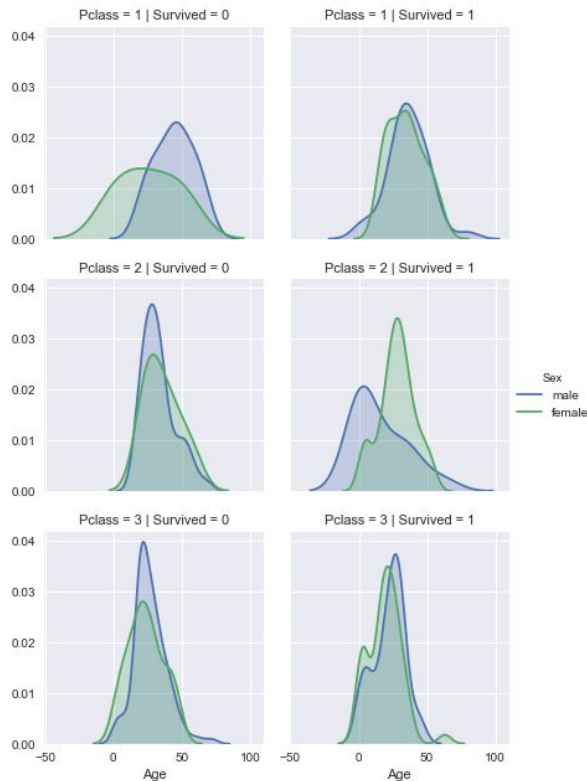


# Distribuições condicionais com três variáveis



```
g = sns.FacetGrid(titanic, col="Survived", row="Pclass")
g.map(sns.kdeplot, "Age", shade=True)
sns.despine(left=True, bottom=True)
plt.show()
```

# Distribuição condicional com três condições



```
g = sns.FacetGrid(titanic, col="Survived", row="Pclass", hue="Sex", size=3)
g.map(sns.kdeplot, "Age", shade=True)
sns.despine(left=True, bottom=True)
g.add_legend()
plt.show()
```

