ECD-01 Visualização de dados

10 de Abril de 2024

Estatística para Ciência de Dados[1]

por **Cibele Russo** e **Francisco Rodrigues - ICMC/USP - São Carlos SP**

Programa

Objetivos: Fornecer conhecimento em descrição e sumarização de dados, probabilidade, inferência estatística, inferência Bayesiana e modelos de regressão, necessários para o desenvolvimento de procedimentos em ciências de dados.

Ementa:

1. **Descritiva**: Medidas de posição, Medidas de dispersão, Agrupamento de dados, Apresentação tabular, Representação Gráfica
2. **Probabilidade**: Distribuições de probabilidade, esperança, variância e covariância, Resultados assintóticos e suas aplicações.
3. **Elementos de inferência estatística**: Funções de evidência e verossimilhança, Procedimentos de estimação pontual, Intervalos de confiança e testes de hipóteses, Inferência baseada em simulação.
4. **Inferência Bayesiana**: O paradigma Bayesiano, Os diferentes tipos de prioris, Distribuições conjugadas, Estimação Bayesiana, Densidade preditiva.
5. **Modelagem de Regressão**: Modelos lineares, Seleção de modelos, Regressão multivariada.

Referências:

1. Casella, G. and Berger, R. (2002). Statistical Inference. 2nd Edition, Duxbury Press, Florida.

2. Migon, H. S., Gamerman, D. and Louzada, F. (2014). Statistical Inference: An Integrated Approach, Second Edition, CRC Press.

[1] **Atenção.** Este material é complementar ao material principal da aula (notebooks ou slides) e pode ser utilizado para consultas.

3. Caffo, B. (2016). Statistical Inference for Data Science. Leanpub. Disponível em https://leanpub.com/LittleInferenceBook

Alguns vídeos complementares sugeridos:

- Playlist disciplina SME0803 Visualização e Exploração de Dados (Prof. Cibele Russo) https://youtube.com/playlist?list=PLt7qVSwRVn5YEIvaMb02IJVKCpauWV-s9
- Análise Exploratória de Dados: Correlação de Pearson e Spearman (Prof. Francisco Rodrigues) https://www.youtube.com/watch?v=qqRUsY2Fu0A

... e outras que serão citadas ao longo do curso.

1 Aula 1. Visualização de dados - Análise descritiva

1.1 Programa

a. Medidas de posição ou localização

b. Medidas de dispersão

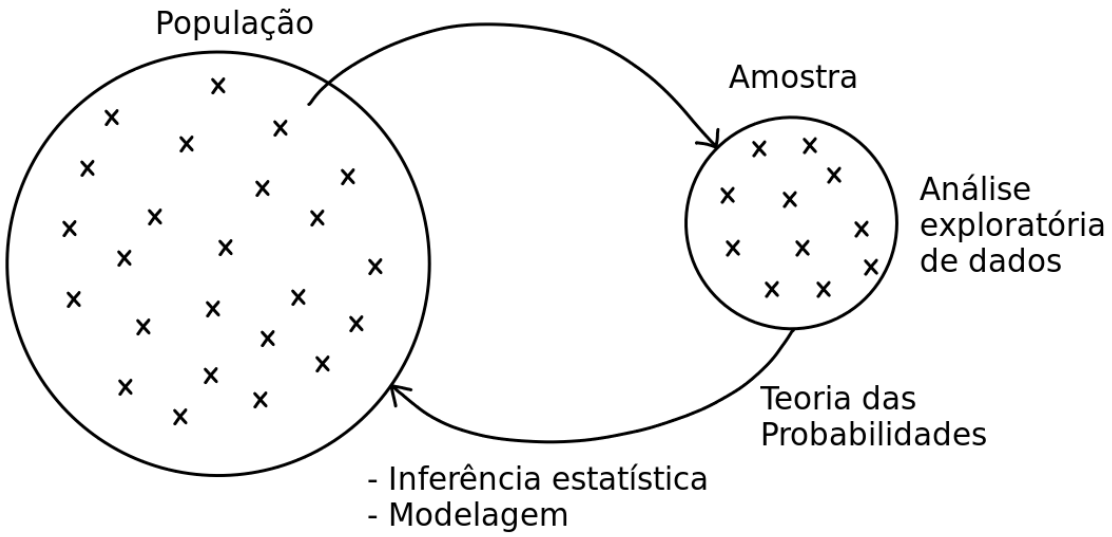
c. Agrupamento de dados

d. Apresentação tabular

e. Representação Gráfica

1.1.1 Referências e motivação:

- Seaborn: statistical data visualization: https://seaborn.pydata.org/index.html.
- COVID-19 Dashboard by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University: https://coronavirus.jhu.edu/map.html
- Capa do The New York Times em 21/02/2021: https://pbs.twimg.com/media/EuwfGryXAAE6zhc?format=jpg&name=large
- 20+ Electoral Maps Visualizing 2020 U.S. Presidential Election Results — DataViz Weekly Special Edition: https://www.anychart.com/blog/2020/11/06/election-maps-us-vote-live-results/
- **Gráfico de linhas:** Impacto da pandemia na educação https://twitter.com/gabrielbcor/status/1499122242369863691/photo/1
- Art of Stat: https://artofstat.com/web-apps
- Histograma humano: https://banderson02.files.wordpress.com/2014/05/f11-17_height_in_human__c.jpg



1.2 Análise exploratória de dados

Análise descritiva ou **análise exploratória de dados** (AED) tem como objetivos básicos:

- explorar os dados para descobrir ou identificar aspectos ou padrões de maior interesse,
- representar os dados de forma a destacar ou chamar a atenção para aspectos ou padrões que podem ou não se confirmar inferencialmente.

Tukey (1977) chama a **análise exploratória de dados** de **trabalho de detetive**, que busca pistas e evidência, e a **análise confirmatória de dados** é um **trabalho judicial ou quase-judicial**, que analisa e avalia a força das provas e da evidência.

Tukey também diz que: **"A análise exploratória de dados nunca conta a história toda, mas nada é tão perfeito para ser considerado a pedra fundamental, um primeiro passo para a análise de dados"**.

É importante salientar que a AED é um trabalho inicial, a pedra fundamental, e os resultados devem ser analisados com uma análise confirmatória.

Tukey, John W. (1977) Exploratory data analysis. Editora Addison-Wesley.

1.3 A natureza dos dados

Nesta aula, e quase sempre neste curso de Estatística para Ciência de Dados, trataremos de **dados retangulares**, que tem nas linhas as **unidades amostrais (exemplos, samples)** e nas colunas as **variáveis (atributos, features)**.

1.3.1 Tipos de variáveis

- **Qualitativas (não-numéricas)**

  - **Nominais:** sexo, cor da pele, fumante/não-fumante, adimplente/inadimplente
  - **Ordinais:** escolaridade (em categorias), grau de satisfação, idade (em faixas)

- **Quantitativas (numéricas)**

  - **Discretas:** número de defeitos em uma peça, número de produtos contratados
  - **Contínuas:** peso, idade, pressão sanguínea, valor contratado de um produto

1.4 Medidas-resumo

- **Medidas de posição**

  - **Média:** boas propriedades estatísticas
  - **Mediana:** medida resistente
  - **Moda:** valor mais frequente
  - **Quantis:** caracterização da distribuição dos dados

- **Medidas de dispersão**

  - **Desvio-padrão**
  - **Variância**
  - **Amplitude (range)**
  - **Coeficiente de variação: medida de dispersão relativa**

- **Assimetria**: Assimetria da distribuição dos dados

- **Curtose**: Achatamento da distribuição

- **Medidas de associação**: Covariância, Coeficiente de correlação de Pearson, Coeficiente de correlação de Spearman

1.4.1 Medidas de posição

Daqui em diante, vamos estabelecer $X_1, \ldots, X_n$ é uma amostra aleatória e $x_1, \ldots, x_n$ os **dados observados** dessa amostra. As medidas aqui apresentadas são **amostrais** e são obtidas a partir de $x_1, \ldots, x_n$.

A **média** (amostral observada) é definida como

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

Considere agora os dados ordenados $x_{(1)}, \ldots, x_{(n)}$, isto é, $x_{(1)} = min(x_1, \ldots, x_n)$ e $x_{(n)} = max(x_1, \ldots, x_n)$.

Se n é ímpar, a posição central é $c = (n+1)/2$. Se n é par, as posições centrais são $c = n/2$ e $c + 1 = n/2 + 1$.

A **mediana** é definida como

$$Md = \begin{cases} x_{(c)}, \text{se n é ímpar} \\ \frac{x_{(c)} + x_{(c+1)}}{2}, \text{se n é par} \end{cases}$$

A **moda** é o valor mais frequente da amostra. Não necessariamente existe.

Um **quantil** é o valor que provoca uma divisão conveniente nos valores ordenados. O quantil de 10% divide os dados de tal forma que 10% dos menores valores fiquem "à sua esquerda". O quantil de 50% é a mediana.

Os **quartis** dividem os dados em porções de 25%.

Os **decis** dividem os dados em porções de 10%.

Os **percentis** dividem os dados em porções de 1%.

1.4.2 Medidas de dispersão

A **variância amostral** é dada por $s^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}$.

O **desvio padrão** é dado por $s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}}$.

É comum, entretanto, utilizar as medidas corrigidas:

Variância amostral corrigida: $s^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}$

Desvio padrão corrigido: $s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}}$

A **amplitude** é dada por $A = x_{(n)} - x_{(1)}$.

O **coeficiente de variação** (amostral) é dado pela razão entre o desvio-padrão e a média $CV = \frac{s}{\bar{x}}$

1.4.3 Assimetria

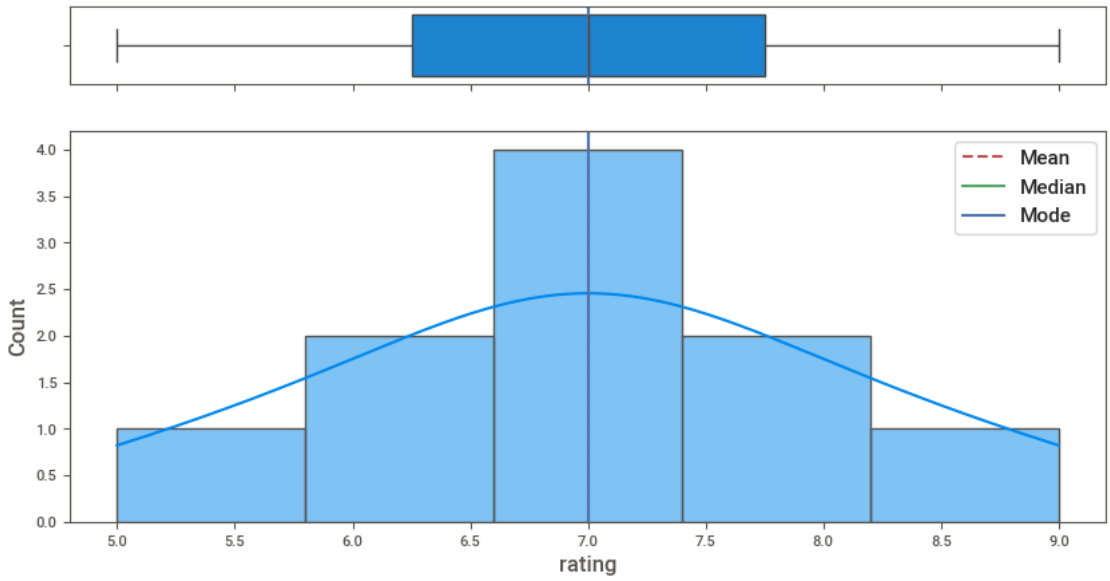
- Distribuição simétrica: média = mediana = moda
- Distribuição assimétrica à direita: moda < mediana < média
- Distribuição assimétrica à esquerda: média < mediana < moda

1.4.4 Curtose

- Distribuições mesocúrticas: achatamento da distribuição normal

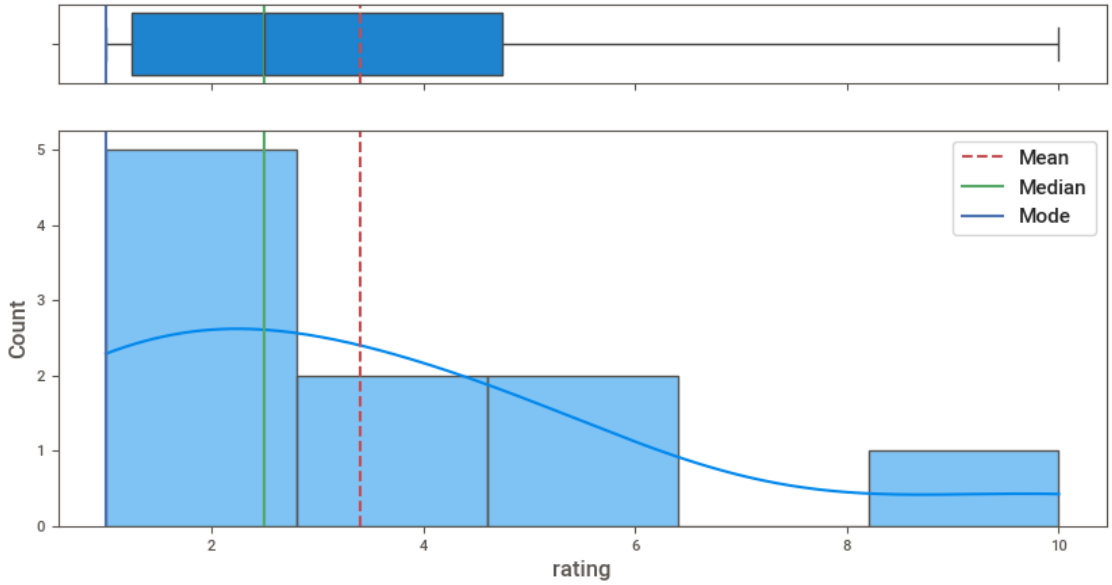
- Distribuições leptocúrticas: distribuição mais concentrada
- Distribuições platicúrticas: distribuição mais achatada

```
In [79]: # Ilustração das medidas média, moda, mediana para dados simétricos
# Adaptado de https://stackoverflow.com/questions/51417483/mean-median-mode-lines-showing-only-i

from matplotlib import pyplot as plt
import pandas as pd
import seaborn as sns

df = pd.DataFrame({"rating": [5, 6, 6, 7, 7, 7, 7, 8, 8, 9]})

f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw= {"height_ratios": (0.2, 1)})
mean=df['rating'].mean()
median=df['rating'].median()
mode=df['rating'].mode().values[0]

sns.boxplot(data=df, x="rating", ax=ax_box)
ax_box.axvline(mean, color='r', linestyle='--')
ax_box.axvline(median, color='g', linestyle='-')
ax_box.axvline(mode, color='b', linestyle='-')

sns.histplot(data=df, x="rating", ax=ax_hist, kde=True)
ax_hist.axvline(mean, color='r', linestyle='--', label="Mean")
ax_hist.axvline(median, color='g', linestyle='-', label="Median")
ax_hist.axvline(mode, color='b', linestyle='-', label="Mode")

plt.legend()

ax_box.set(xlabel='')
plt.show()
```



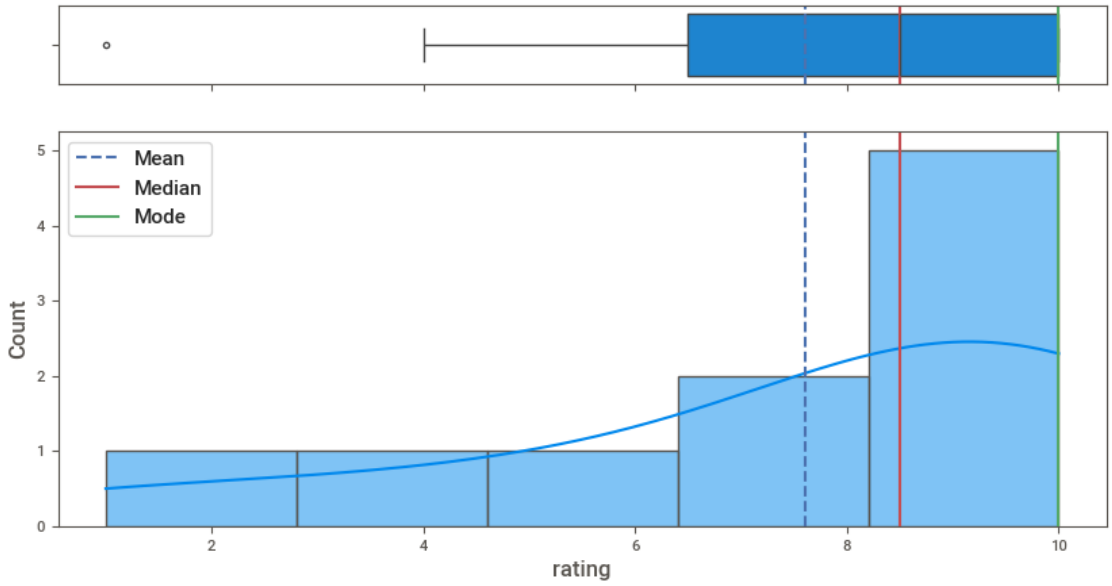

```
In [80]: # Ilustração das medidas média, moda, mediana para dados assimétricos à direita ou assi
# Adaptado de https://stackoverflow.com/questions/51417483/mean-median-mode-lines-showing-only-i

from matplotlib import pyplot as plt
import pandas as pd
import seaborn as sns
import statistics

df = pd.DataFrame({"rating": [1,1,1,2,2,3,4,5,5,10]})

f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw= {"height_ratios": (0.2, 1)})
mean=df['rating'].mean()
median=df['rating'].median()
mode=df['rating'].mode().values[0]

sns.boxplot(data=df, x="rating", ax=ax_box)
ax_box.axvline(mean, color='r', linestyle='--')
ax_box.axvline(median, color='g', linestyle='-')
ax_box.axvline(mode, color='b', linestyle='-')

sns.histplot(data=df, x="rating", ax=ax_hist, kde=True)
ax_hist.axvline(mean, color='r', linestyle='--', label="Mean")
```

```
ax_hist.axvline(median, color='g', linestyle='-', label="Median")
ax_hist.axvline(mode, color='b', linestyle='-', label="Mode")

plt.legend()


ax_box.set(xlabel='')
plt.show()
```




```
In [81]: # Ilustração das medidas média, moda, mediana para dados assimétricos à esquerda ou com
# Adaptado de: https://stackoverflow.com/questions/51417483/mean-median-mode-lines-showing-only-


from matplotlib import pyplot as plt
import pandas as pd
import seaborn as sns
import statistics

df = pd.DataFrame({"rating": [1, 4, 6, 8, 8, 9, 10, 10, 10, 10]})

f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw= {"height_ratios": (0.2, 1)})
mean=df['rating'].mean()
median=df['rating'].median()
```

```
mode = statistics.mode(df['rating'])

sns.boxplot(data=df, x="rating", ax=ax_box)
ax_box.axvline(mean, color='b', linestyle='--')
ax_box.axvline(median, color='r', linestyle='-')
ax_box.axvline(mode, color='g', linestyle='-')

sns.histplot(data=df, x="rating", ax=ax_hist, kde=True)
ax_hist.axvline(mean, color='b', linestyle='--', label="Mean")
ax_hist.axvline(median, color='r', linestyle='-', label="Median")
ax_hist.axvline(mode, color='g', linestyle='-', label="Mode")

plt.legend()

ax_box.set(xlabel='')
plt.show()
```



1.4.5 Curtose

Referências: - https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kurtosis.html - https://pt.wikipedia.org/wiki/Curtose

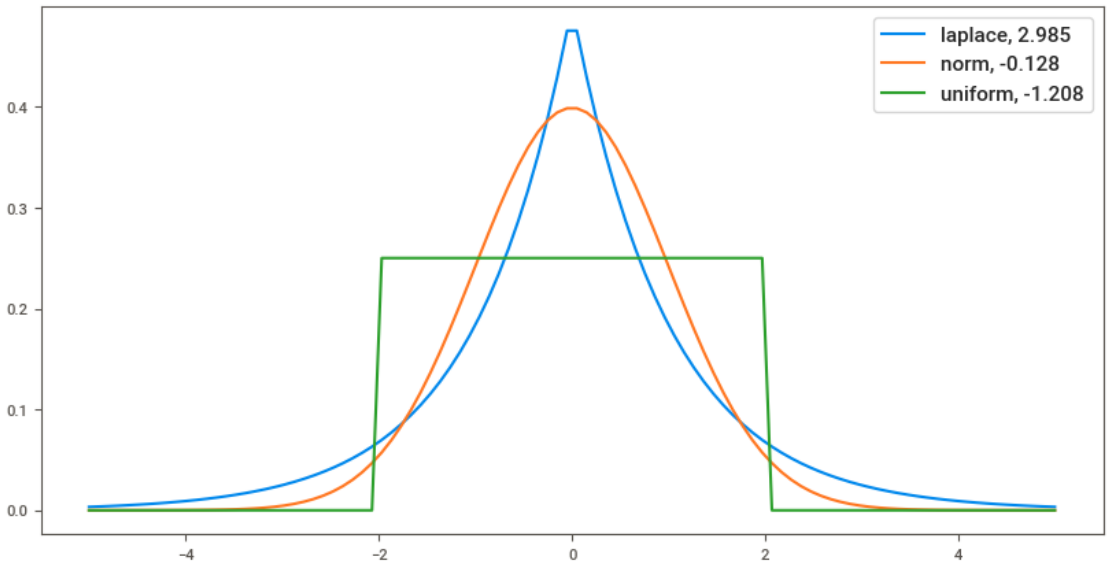
Medida que caracteriza o achatamento da curva.

- Curtose ≈ 0: achatamento da curva normal

- Curtose > 0: leptocúrtica, distribuição mais afunilada
- Curtose < 0: platicúrtica, distribuição mais achatada

Obs: Distribuição normal https://www.spss-tutorials.com/normal-distribution/

```
In [82]: from scipy.stats import norm, kurtosis

data = norm.rvs(size=100000)

kurtosis(data)

Out[82]: -0.007890326386671198

In [83]: # Fonte: https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kurtosis.html

import matplotlib.pyplot as plt
import scipy.stats as stats
from scipy.stats import kurtosis
import numpy as np


x = np.linspace(-5, 5, 100)
ax = plt.subplot()
distnames = ['laplace', 'norm', 'uniform']

for distname in distnames:
    if distname == 'uniform':
        dist = getattr(stats, distname)(loc=-2, scale=4)
    else:
        dist = getattr(stats, distname)
    data = dist.rvs(size=1000)
    kur = kurtosis(data, fisher=True)
    y = dist.pdf(x)
    ax.plot(x, y, label="{}, {}".format(distname, round(kur, 3)))
    ax.legend()

# Normal: mesocúrtica
# Laplace: leptocúrtica
# Uniforme: platicúrtica
```



1.5 Medidas de associação entre variáveis quantitativas

Sejam X e Y variáveis quantitativas de interesse e as amostras aleatórias observadas $x_1, \ldots, x_n$ e $y_1, \ldots, y_n$, respectivamente. As medidas de associação mais utilizadas são:

1.5.1 Covariância (amostral)

$$s_{XY} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

1.5.2 Coeficiente de correlação linear (amostral) de Pearson

Referência: https://pt.wikipedia.org/wiki/Coeficiente_de_correla%C3%A7%C3%A3o_de_Pearson

$$r = \frac{s_{XY}}{\sqrt{s_X^2 s_Y^2}}$$

Propriedade: $-1 \leq r \leq 1$
É comum usar as seguintes classificações:

1. $r = 1$ indica uma correlação perfeita e positiva
2. $r = -1$ indica uma correlação perfeita e negativa
3. $0.7 \leq |r| \leq 1$ indica uma correlação forte

4. $0.5 \leq |r| \leq 0.69$ indica uma correlação moderada

5. $0 \leq |r| \leq 0.49$ indica uma correlação fraca

1.5.3 Coeficiente de correlação de Spearman

Avalia relações monótonas entre duas variáveis

Referência: https://pt.wikipedia.org/wiki/Coeficiente_de_correla%C3%A7%C3%A3o_de_postos_de_Spearman

1.6 Associação entre variáveis qualitativas e quantitativas

Alguns casos que veremos mais adiante:

- Associação entre variáveis quantitativas e qualitativas: Testes para comparação de médias em duas populações.
- Associação entre variáveis qualitativas: Teste qui-quadrado, teste exato de Fisher, entre outros.

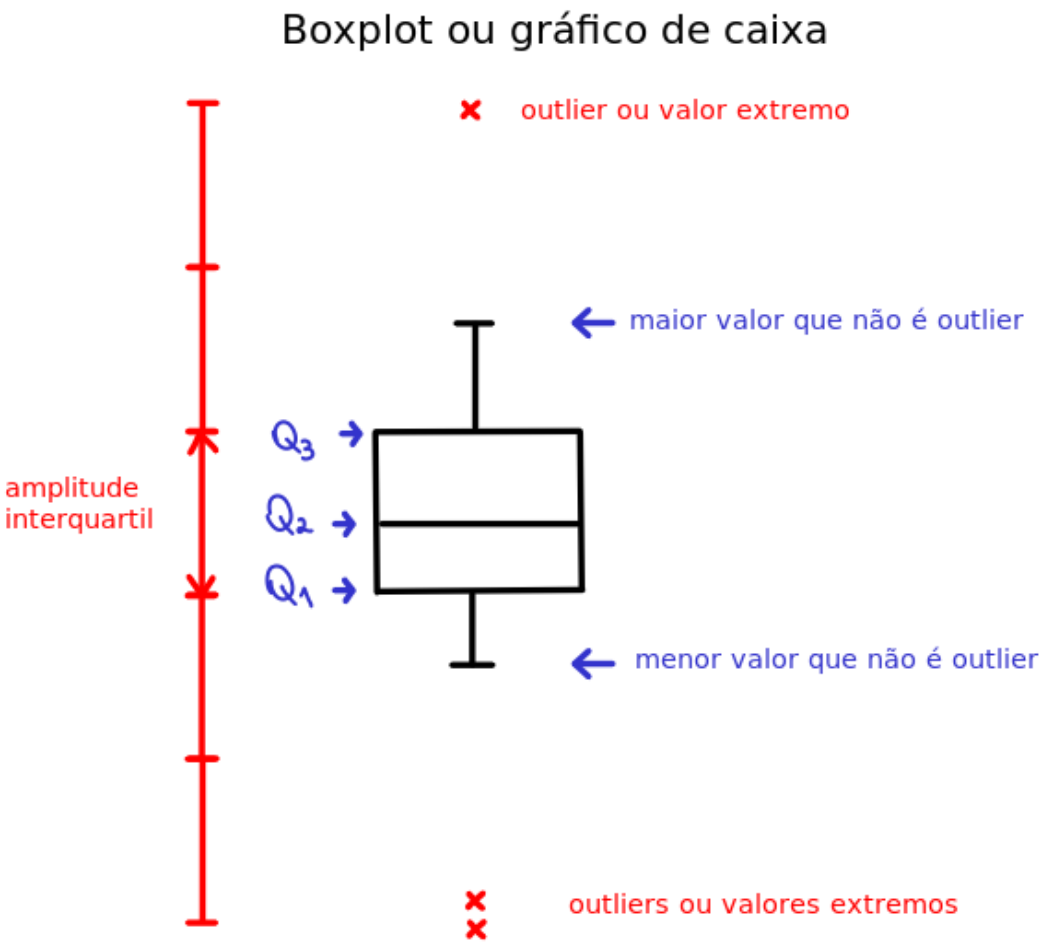
1.7 Representação gráfica e tabular de dados

- **Variáveis qualitativas:**
  - **Tabela de frequência:** Resume a informação dos dados de forma a possibilitar a observação de frequências absolutas ou relativas de cada categoria das variáveis qualitativas (ou valores assumidos pelas variáveis quantitativas discretas).
  - **Gráfico de barras** Representação gráfica das frequências de cada categoria das variáveis qualitativas (ou valores assumidos pelas variáveis quantitativas discretas). As barras são separadas.
  - **Gráfico de Pareto**: Gráfico de barras + frequências acumuladas das categorias.
  - **Gráfico de setores (pizza)**: Representação gráfica das proporções das categorias das variáveis quantitativas discretas.
- **Variáveis quantitativas discretas:**
  - **Tabelas de frequências**
  - **Gráficos de barras**
  - **Gráficos de pontos**
- **Variáveis quantitativas contínuas:**

– **Histogramas**: Representação gráfica para uma aproximação da distribuição de uma variável quantitativa contínua, discretizada em classes de tamanhos convenientes. As barras são adjacentes. Permitem observar a localização, dispersão, assimetria, número de picos, curtose dos dados.

– **Gráficos de linhas (dados coletados ao longo do tempo)**

– **Boxplots (gráficos de caixas)**: Representação gráfica inteligente que permite a observação da localização, dispersão, assimetria, pontos discrepantes (outliers). Além disso, permite comparar visualemente a distribuição de dados em dois grupos. Pode indicar evidências sobre a igualdade das médias entre os dados de dois grupos, pendente de análise confirmatória inferencial.



1.7.1 Representação tabular

Tabelas que resumem a informação da base completa de dados.

- **Tabelas de frequências:** Resumo dos dados originais considerando as frequências observadas na amostra, de variáveis qualitativas ou variáveis que foram categorizadas

- **Tabelas de dupla entrada:** Avaliação da associação entre variáveis qualitativas ou que foram categorizadas.

1.8 Aplicação com visualização e exploração de dados

Considere uma amostra de 10 mil clientes de um banco no arquivo dados_banco.csv. Estão disponíveis as variáveis:

- Cliente: Identificador do cliente.
- Sexo: Feminino (F) ou Masculino (M)
- Idade: Idade do cliente, em anos completos.
- Empresa: Tipo da empresa em que trabalha: Pública, Privada ou Autônomo
- Salário: Salário declarado pelo cliente na abertura da conta, em reais.
- Saldo_cc: Saldo em conta corrente, em reais.
- Saldo_poupança: Saldo em poupança, em reais.
- Saldo_investimento: Saldo em investimentos, em reais.
- Devedor_cartao: Valor em atraso no cartão de crédito, em reais.
- Inadimplente: Se o cliente é considerado inadimplente atualmente (1) ou não (0), de acordo com critérios preestabelecidos.

Desenvolva a exploração e visualização dos dados. Verifique possíveis associações entre variáveis.

```
In [84]: import os.path
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from scipy import stats

%matplotlib inline

# Modifique o diretório para fazer a leitura dos dados em dados_banco.csv

# Dados banco - Leitura dos dados
# Caso necessário, leia a partir de um diretório da sua máquina
# pkgdir = '/hdd/MBA/ECD/Data'
# dados = pd.read_csv(f'{pkgdir}/dados_banco.csv', index_col=0)

dados = pd.read_csv('https://raw.githubusercontent.com/cibelerusso/Estatistica-Ciencia-Dados/mai

dados
```

```
Out[84]:Sexo  Idade   Empresa  Salario  Saldo_cc  Saldo_poupança  \
Cliente
75928      M     32   Privada  5719.00    933.79    0.0
52921      F     28   Privada  5064.00    628.37    0.0
8387       F     24  Autônomo  4739.00    889.18    0.0
54522      M     30   Pública  5215.00   1141.47    0.0
45397      M     30  Autônomo  5215.56    520.70    0.0
...      ...    ...       ...      ...       ...    ...
33487      F     31   Pública  5016.00    498.96    0.0
71360      M     29   Pública  5329.00   1142.82    0.0
92455      M     34   Privada  5581.00    885.34    0.0
61296      F     28   Privada  5061.00    660.74    0.0
52862      M     33  Autônomo  5519.00   1147.71    0.0

Saldo_investimento  Devedor_cartao  Inadimplente
Cliente
75928 0.06023.68    0
52921 0.01578.24    0
8387  0.02578.70    0
54522 0.04348.96    0
45397 0.01516.78    1
...   ...    ...  ...
33487 0.01263.34    0
71360 0.05613.71    0
92455 0.01199.22    0
61296 0.01152.97    0
52862 0.04684.66    0

[10000 rows x 9 columns]

In [85]: dados.head()

Out[85]:Sexo  Idade   Empresa  Salario  Saldo_cc  Saldo_poupança  \
Cliente
75928      M     32   Privada  5719.00    933.79    0.0
52921      F     28   Privada  5064.00    628.37    0.0
8387       F     24  Autônomo  4739.00    889.18    0.0
54522      M     30   Pública  5215.00   1141.47    0.0
45397      M     30  Autônomo  5215.56    520.70    0.0
```

```
Saldo_investimento  Devedor_cartao  Inadimplente
Cliente
75928 0.06023.68    0
52921 0.01578.24    0
8387  0.02578.70    0
54522 0.04348.96    0
45397 0.01516.78    1
```

1.8.1 Classificação das variáveis por tipo

- Sexo: qualitativa nominal
- Idade: quantitativa contínua
- Empresa: qualitativa nominal
- Salário: quantitativa contínua
- Saldo_cc: quantitativa contínua
- Saldo_poupança: quantitativa contínua
- Saldo_investimento: quantitativa contínua
- Devedor_cartão: quantitativa contínua
- Inadimplente: qualitativa nominal (embora numérica)

1.8.2 Tabela de frequências (absolutas e relativas)

(para a Empresa, repetir para outras variáveis qualitativas)

```
In [86]: # Tabela de frequências absolutas

tab = pd.crosstab(index=dados['Empresa'], columns='count')

tab

Out[86]: col_0     count
Empresa
Autônomo   1447
Privada    6103
Pública    2450

In [87]: tab = pd.crosstab(index=dados['Empresa'], columns='count')

# Tabela de frequências relativas
tab/tab.sum()
```

```
Out[87]: col_0      count
Empresa
Autônomo  0.1447
Privada   0.6103
Pública   0.2450
```

Análise: Na base de dados, cerca de 61% dos clientes trabalham em empresas privadas, 24% em empresas públicas e 15% são autônomos.

1.8.3 Medidas resumo

(para a idade, poderia repetir para as outras variáveis quantitativas)

```
In [88]: # Média

dados['Idade'].mean()

Out[88]: 31.8019

In [89]: # Mediana

dados['Idade'].median()

Out[89]: 32.0

In [90]: # Desvio-padrão

round(dados['Idade'].std(),2)

Out[90]: 2.93

In [91]: # Média de idade por grupos

dados.groupby('Sexo')['Idade'].mean()

Out[91]: Sexo
F    30.130466
M    33.027734
Name: Idade, dtype: float64
```

Análise: A média de idade nos dados é 31.8 anos, a mediana é 32 anos. O desvio-padrão da idade na base de dados geral é 2.93 anos. Entre mulheres, a média de idade é 30.1 anos e entre homens, 33 anos.

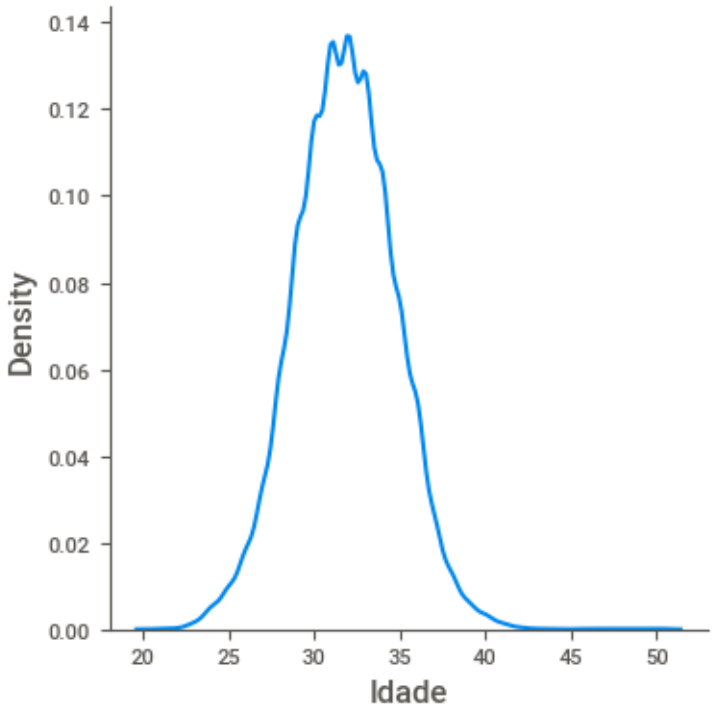
```
In [92]: # Média de idade por grupos

dados.groupby('Empresa')['Idade'].mean()

Out[92]: Empresa
Autônomo    29.163787
Privada     32.867115
Pública     30.706531
Name: Idade, dtype: float64
```

Análise: A média de idade entre os clientes autônomos é de 29.1 anos, entre clientes que trabalham em empresas privadas é 32.9 anos e para clientes que trabalham em empresas públicas é 30.7 anos.

```
In [93]: # Moda - para a Empresa
import statistics

statistics.mode(dados['Empresa'])

Out[93]: 'Privada'
```

Análise: Na base de dados, o tipo de empresa mais comum é a empresa privada.

```
In [94]: # Ordenação dos dados

np.sort(dados['Idade'])

Out[94]: array([21, 22, 22, ..., 49, 50, 50])

In [95]: # Quantis de 95% e 25%

np.percentile(dados['Idade'],95)

Out[95]: 36.0

In [96]: np.percentile(dados['Idade'],25)

Out[96]: 30.0

In [97]: sns.displot(x=dados['Idade'], height=4, kind='kde');
```




```
In [98]: tab1 = pd.crosstab(index=dados['Sexo'], columns='count')
tab1/tab1.sum()
```

```
Out[98]: col_0   count
Sexo
F       0.4231
M       0.5769
```

1.8.4 Estatísticas descritivas dos dados com describe()

```
In [99]: dados.describe()
```

```
Out[99]:       Idade       Salario      Saldo_cc  Saldo_poupança  \
count  10000.000000  10000.000000  10000.000000    10000.000000
mean      31.801900   5482.880238    773.441611     2224.517679
std        2.931913    393.779438    246.932963     5668.740769
min       21.000000   4325.720000   -280.670000        0.000000
25%       30.000000   5207.540000    599.425000        0.000000
50%       32.000000   5498.780000    766.000000        0.000000
75%       34.000000   5738.220000    941.470000        0.000000
```

```
max       50.000000   8582.000000   2007.260000    23336.420000

       Saldo_investimento  Devedor_cartao  Inadimplente
count        10000.000000    10000.000000  10000.000000
mean 1476.939508     2737.210731      0.246100
std  3920.049185     1994.877093      0.430759
min     0.000000        0.000000      0.000000
25%     0.000000     1186.807500      0.000000
50%     0.000000     2692.935000      0.000000
75%     0.000000     4058.565000      0.000000
max 21810.520000    12312.220000      1.000000
```

```
In [100]: dados.loc[:,dados.columns != 'Cliente'].describe()
```

```
Out[100]:       Idade       Salario      Saldo_cc  Saldo_poupança  \
 count  10000.000000  10000.000000  10000.000000    10000.000000
 mean      31.801900   5482.880238    773.441611     2224.517679
 std        2.931913    393.779438    246.932963     5668.740769
 min       21.000000   4325.720000   -280.670000        0.000000
 25%       30.000000   5207.540000    599.425000        0.000000
 50%       32.000000   5498.780000    766.000000        0.000000
 75%       34.000000   5738.220000    941.470000        0.000000
 max       50.000000   8582.000000   2007.260000    23336.420000

        Saldo_investimento  Devedor_cartao  Inadimplente
 count        10000.000000    10000.000000  10000.000000
 mean 1476.939508     2737.210731      0.246100
 std  3920.049185     1994.877093      0.430759
 min     0.000000        0.000000      0.000000
 25%     0.000000     1186.807500      0.000000
 50%     0.000000     2692.935000      0.000000
 75%     0.000000     4058.565000      0.000000
 max 21810.520000    12312.220000      1.000000
```

1.8.5 Gráfico de setores (pizza)

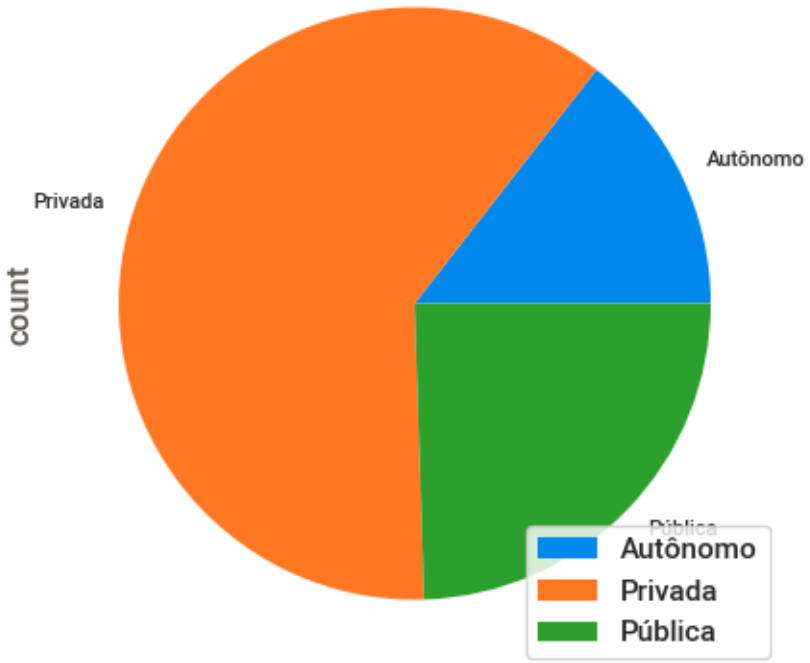
- https://blog.algorexhealth.com/2018/03/almost-10-pie-charts-in-10-python-libraries/

```
In [101]: tab
```

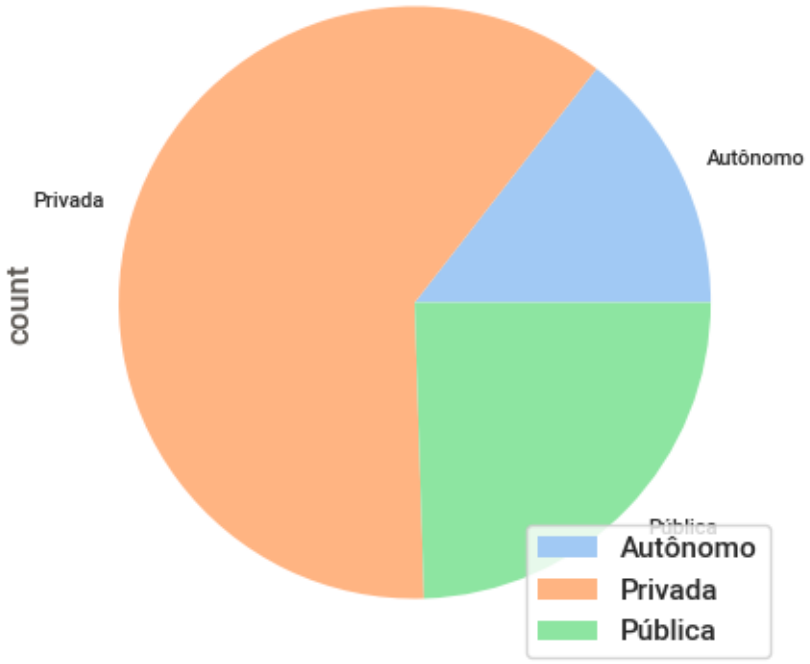
```
Out[101]: col_0     count
 Empresa
```

```
Autônomo    1447
Privada     6103
Pública     2450
```

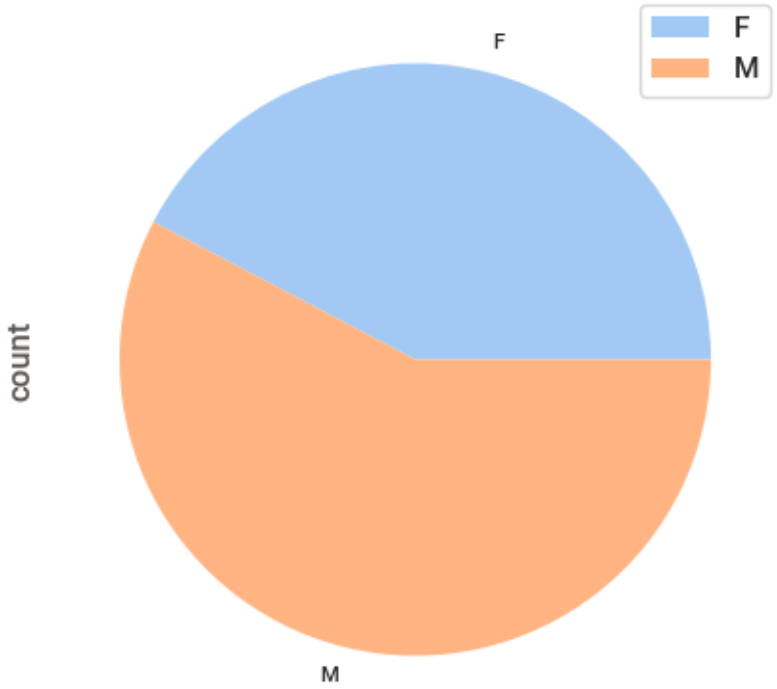
```
In [102]: plot = tab.plot.pie(y='count')
```




```
In [103]: #define Seaborn color palette to use
 colors = sns.color_palette('pastel')[0:5]
 plot = tab.plot.pie(y='count', colors=colors)
```

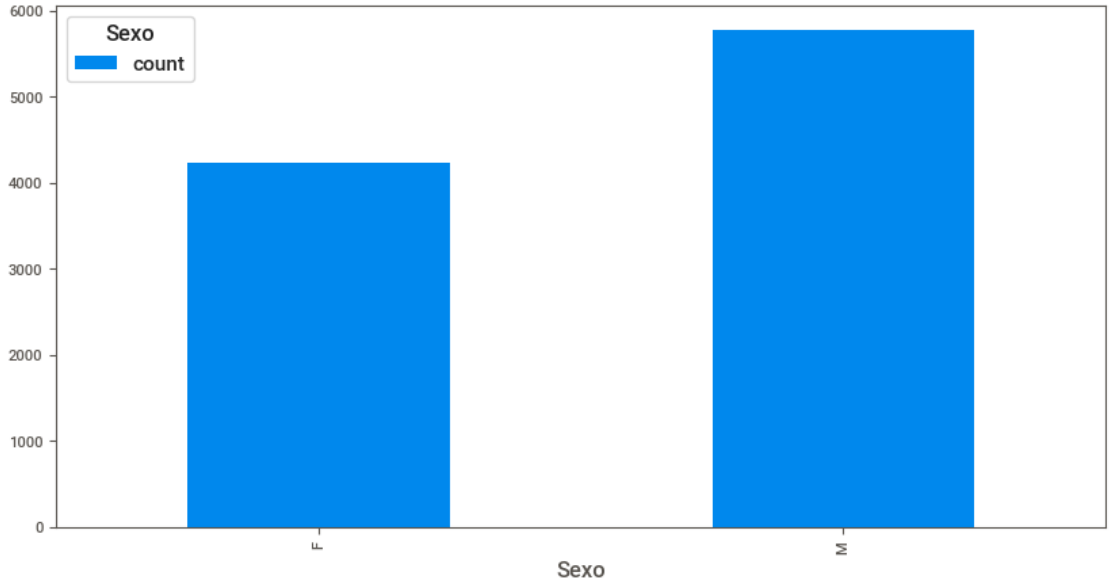



```
In [104]: # Tabela de frequências absolutas

 tab = pd.crosstab(index=dados['Sexo'], columns='count')

 tab

Out[104]: col_0  count
 Sexo
 F       4231
 M       5769

In [105]: plot = tab.plot.pie(y='count', colors=colors)
```



1.8.6 Gráfico de barras

```
In [106]: tab.plot.bar()
 plt.legend(title='Sexo')

 plt.show()
```



1.8.7 Boxplot

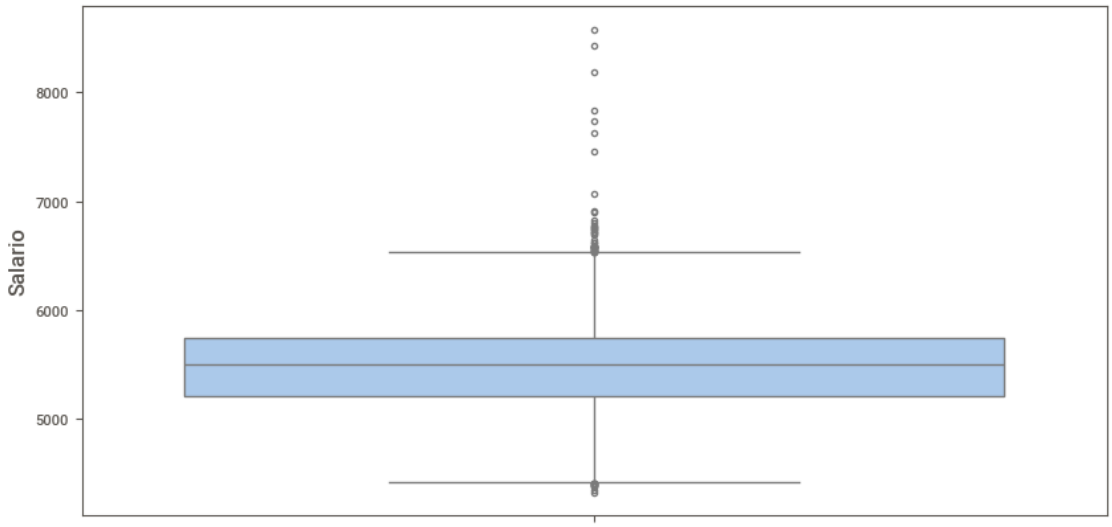
- Posição
- Dispersão
- Outliers
- Assimetria

https://seaborn.pydata.org/generated/seaborn.boxplot.html

```
In [107]: dados['Salario']
```

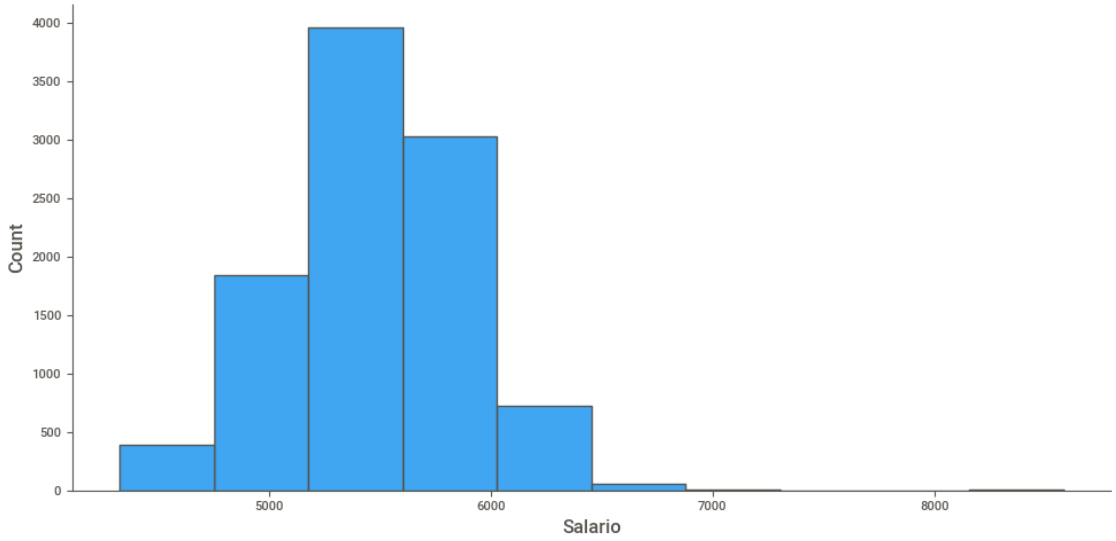
```
Out[107]: Cliente
 75928    5719.00
 52921    5064.00
 8387     4739.00
 54522    5215.00
 45397    5215.56
  ...
 33487    5016.00
 71360    5329.00
 92455    5581.00
 61296    5061.00
 52862    5519.00
 Name: Salario, Length: 10000, dtype: float64
```

```
In [108]: sns.boxplot(y=dados['Salario'], palette='pastel')

<ipython-input-108-267c2bf6a195>:1: FutureWarning:


Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign t


Out[108]: <Axes: ylabel='Salario'>
```

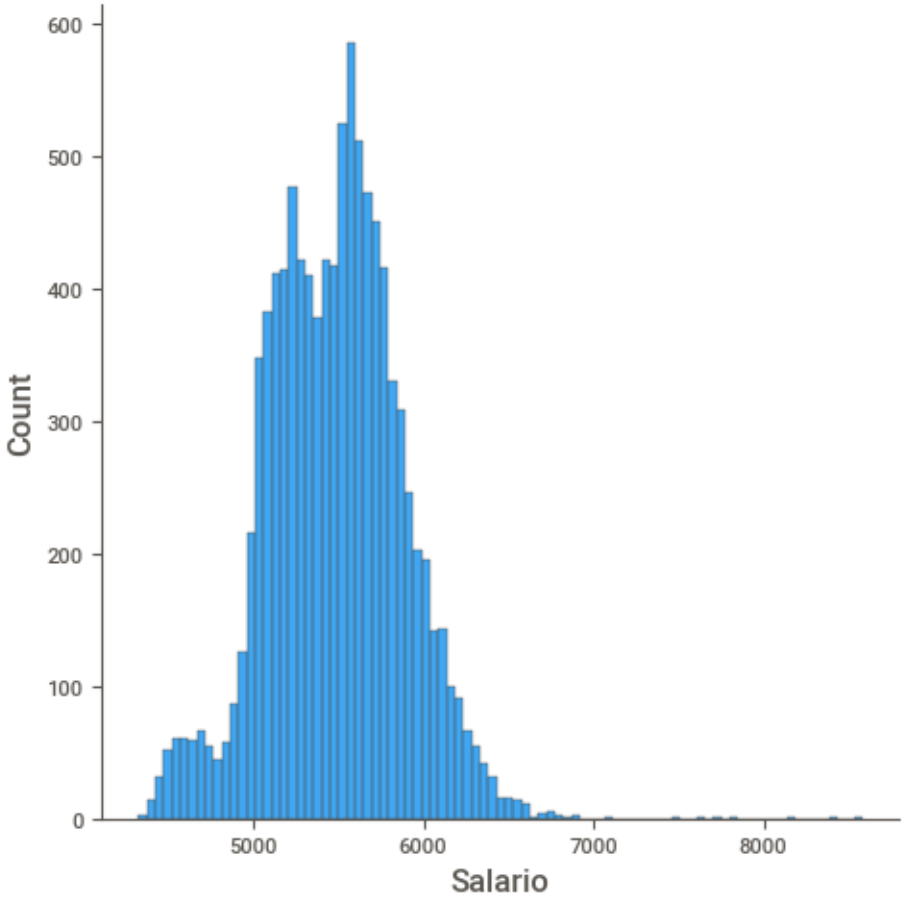


Histograma

```
In [109]: sns.displot(dados['Salario'],kde=False, bins=10, height=5, aspect=2);
```

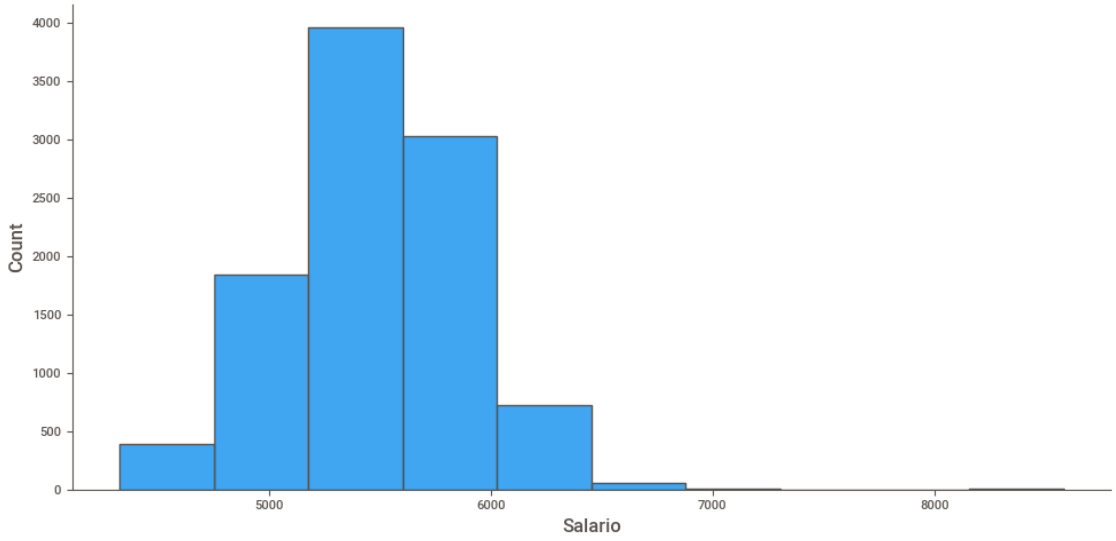



```
In [110]: sns.displot(dados['Salario']);
```



```
In [111]: sns.displot(dados['Salario'], bins=10, height=5, aspect=2)
```

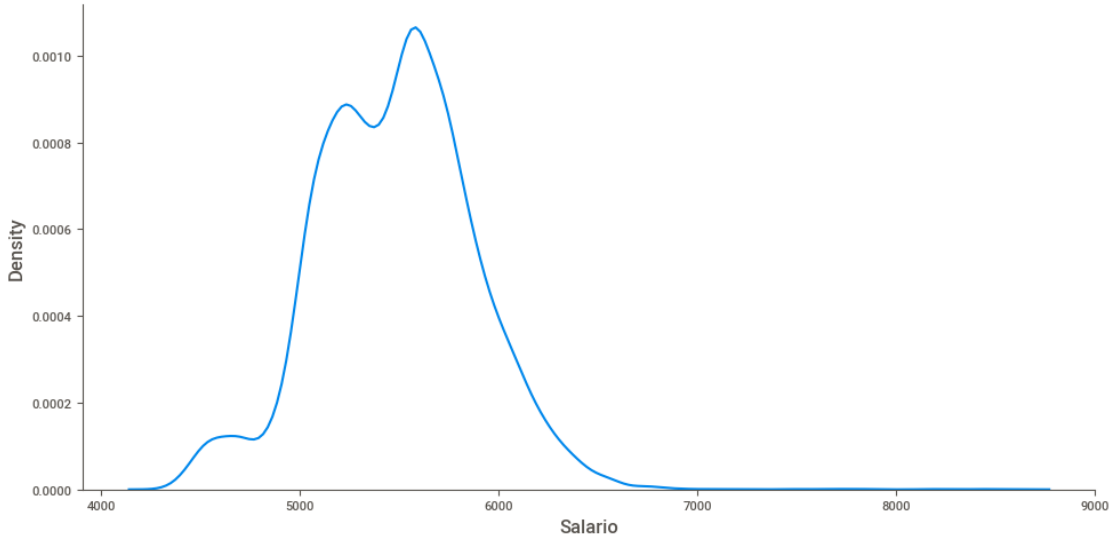
```
Out[111]: <seaborn.axisgrid.FacetGrid at 0x7dcb628f0ca0>
```



Densidade alisada

```
In [112]: sns.displot(dados['Salario'], kind='kde', height=5, aspect=2)
```

```
Out[112]: <seaborn.axisgrid.FacetGrid at 0x7dcb62c720e0>
```

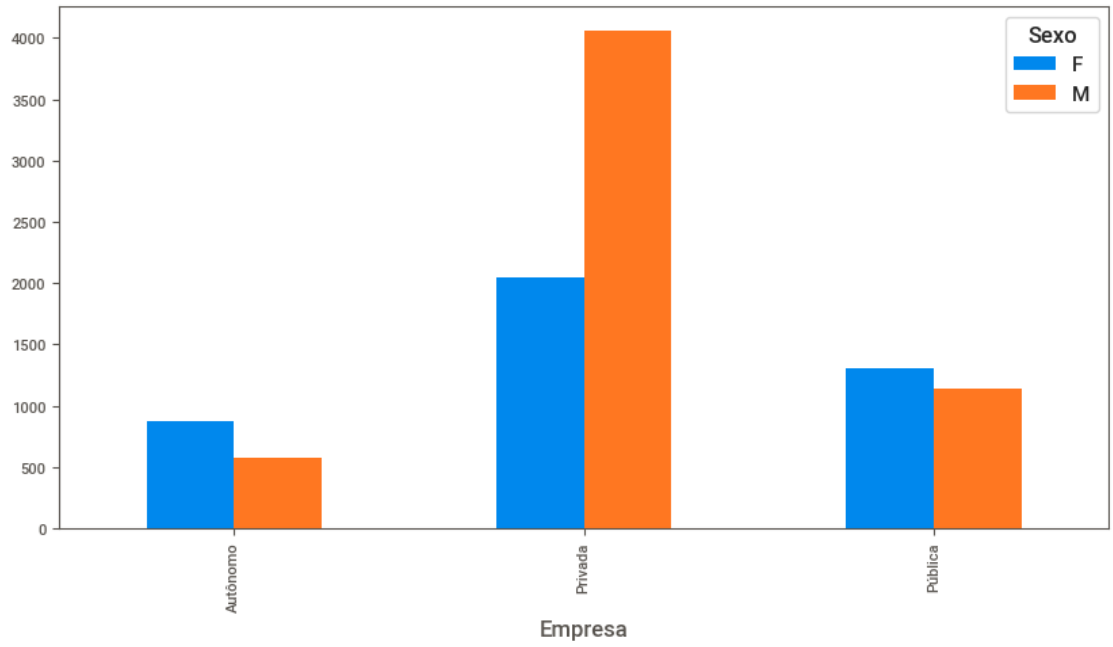


1.8.8 Associação entre duas variáveis qualitativas

```
In [113]: # Tabela de dupla entrada

 tabela_dupla = pd.crosstab(index=dados['Empresa'], columns=dados['Sexo'])

 tabela_dupla
```

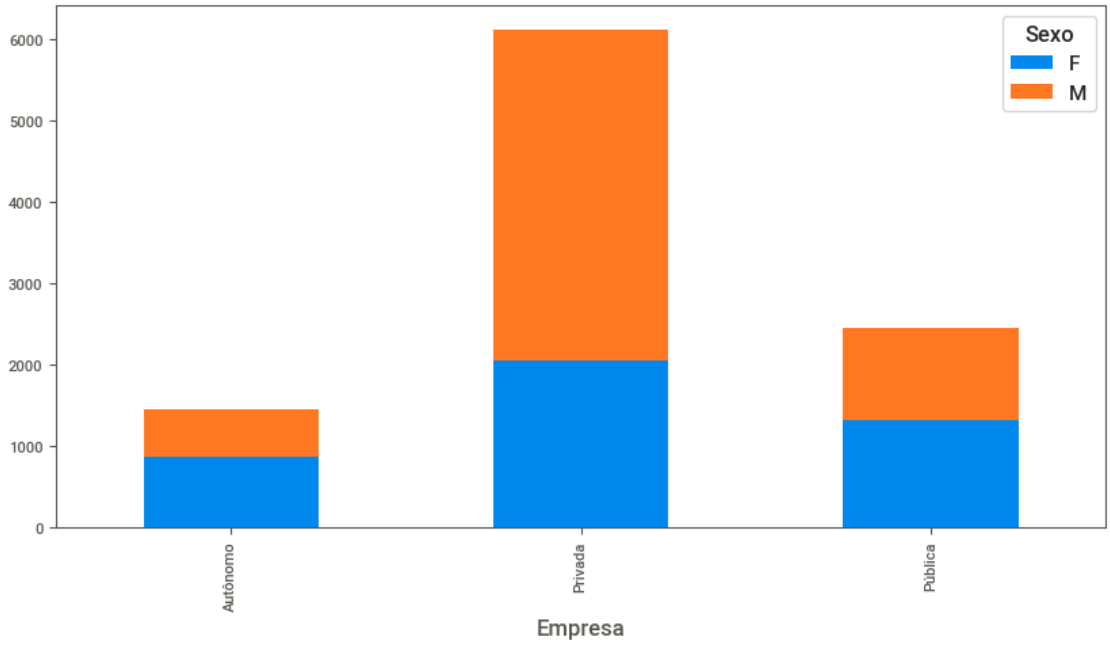
```
Out[113]: SexoF     M
 Empresa
 Autônomo   875   572
 Privada   2047  4056
 Pública   1309  1141
```

```
In [114]: tabela_dupla.plot.bar()

 plt.legend(title='Sexo')

 plt.show()
```



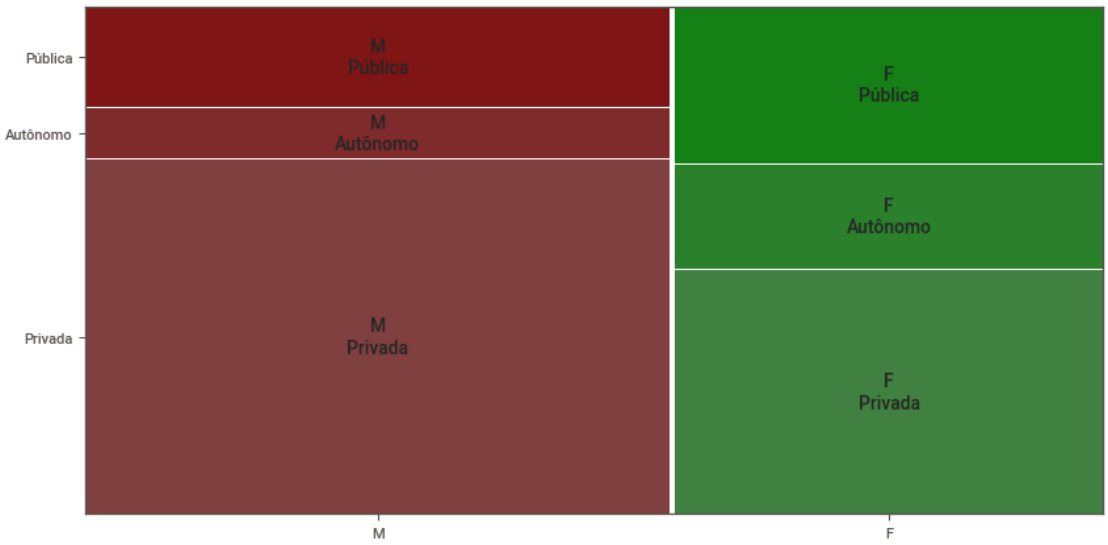

```
In [115]: tabela_dupla.plot.bar(stacked=True)
```

```
plt.legend(title='Sexo')

plt.show()
```



1.8.9 Gráfico de mosaico

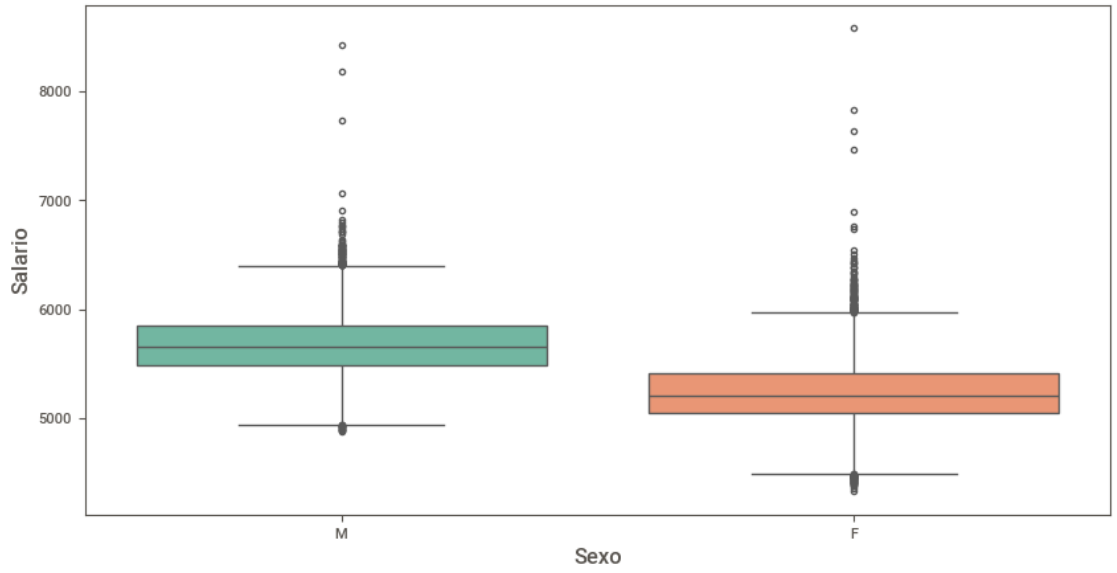
```
In [116]: from statsmodels.graphics.mosaicplot import mosaic

plt.rcParams["figure.figsize"] = [10, 5]

mosaic(dados,['Sexo', 'Empresa'] );
```



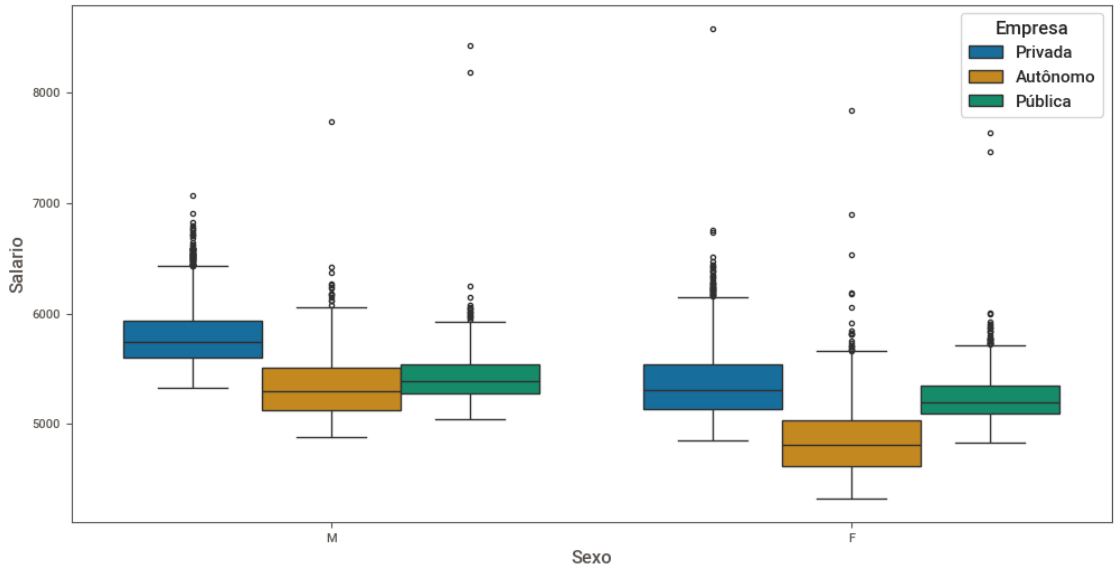
1.8.10 Associação entre variáveis quantitativas e qualitativas

```
In [117]: ax = sns.boxplot(x='Sexo', y='Salario', data=dados, palette='Set2')
```

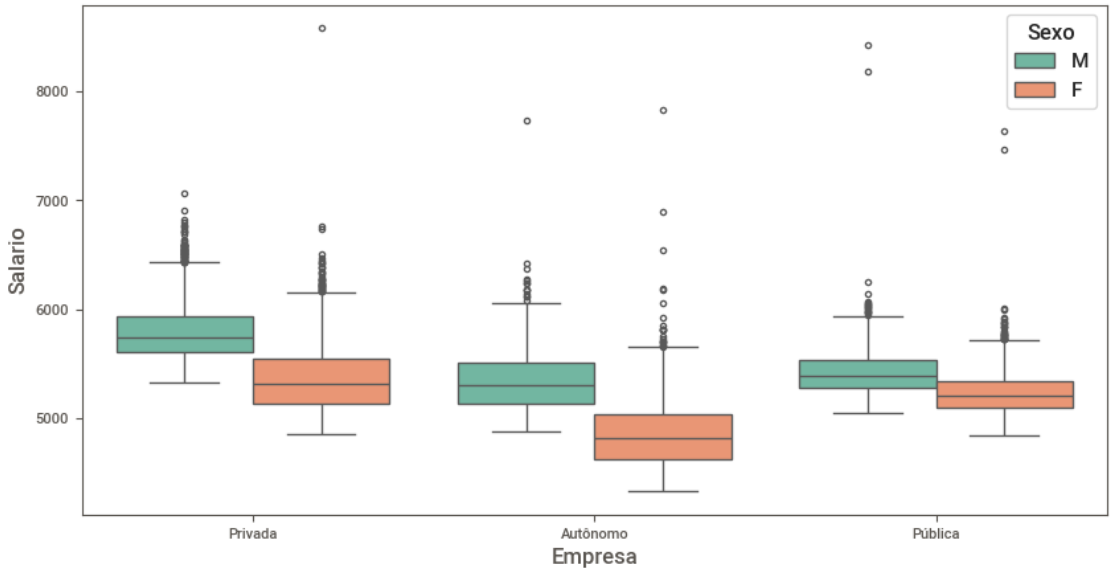
```
<ipython-input-117-f885d1a815c5>:1: FutureWarning:


Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign t
```




```
In [118]: plt.figure(figsize=(12,6))
 ax = sns.boxplot(x='Sexo', y='Salario', hue='Empresa', data=dados, palette='colorblind')
```

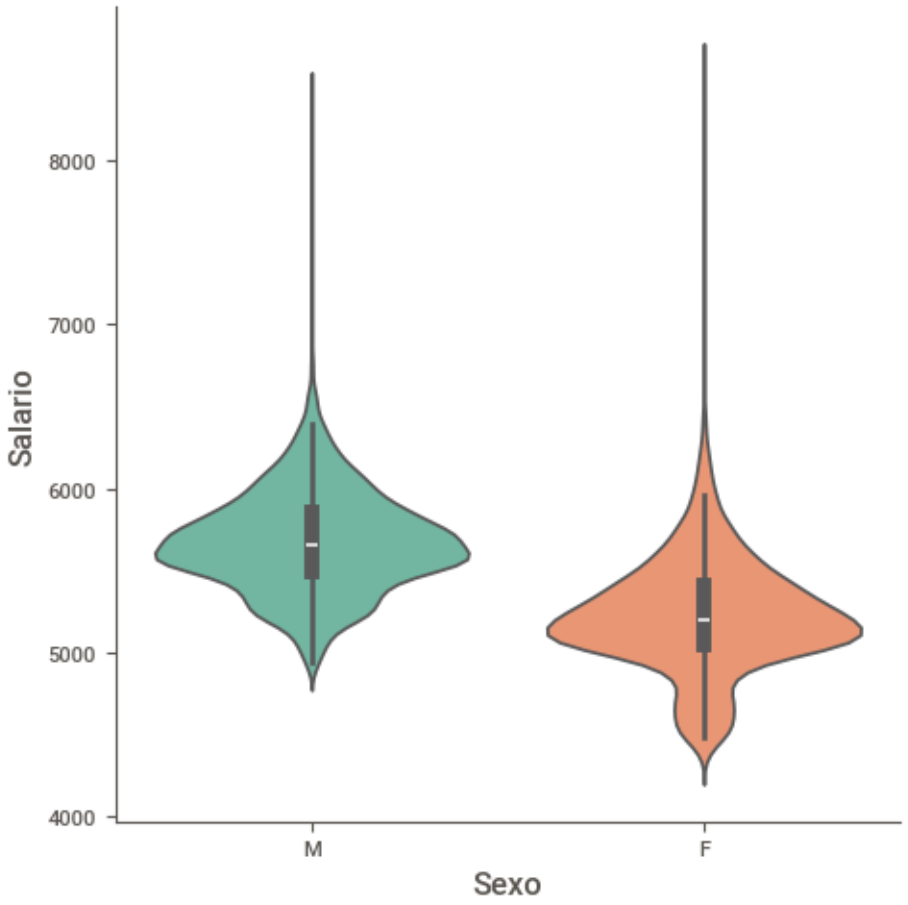



```
In [119]: ax = sns.boxplot(x='Empresa', y='Salario', hue='Sexo', data=dados, palette='Set2')
```

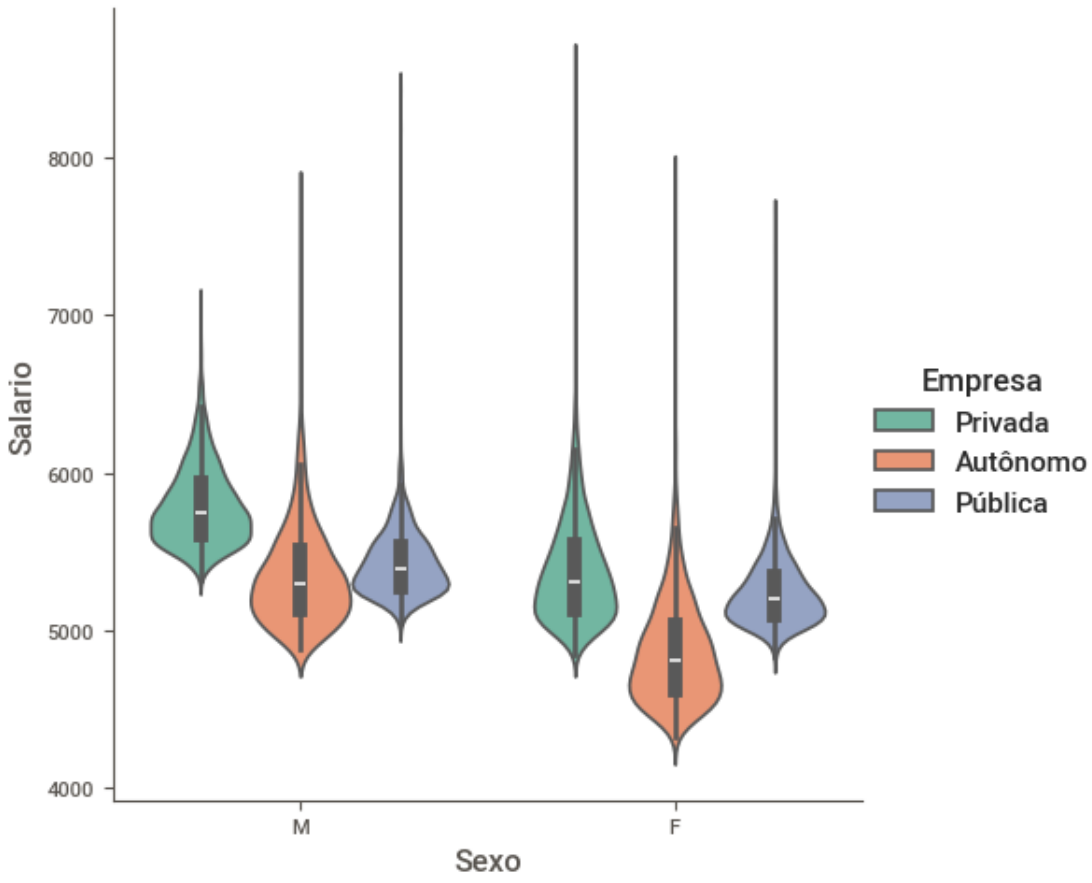



```
In [120]: ax = sns.catplot(x='Sexo', y='Salario', kind='violin', data=dados, palette='Set2')
```

```
<ipython-input-120-b7ace95513f6>:1: FutureWarning:


Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign t
```

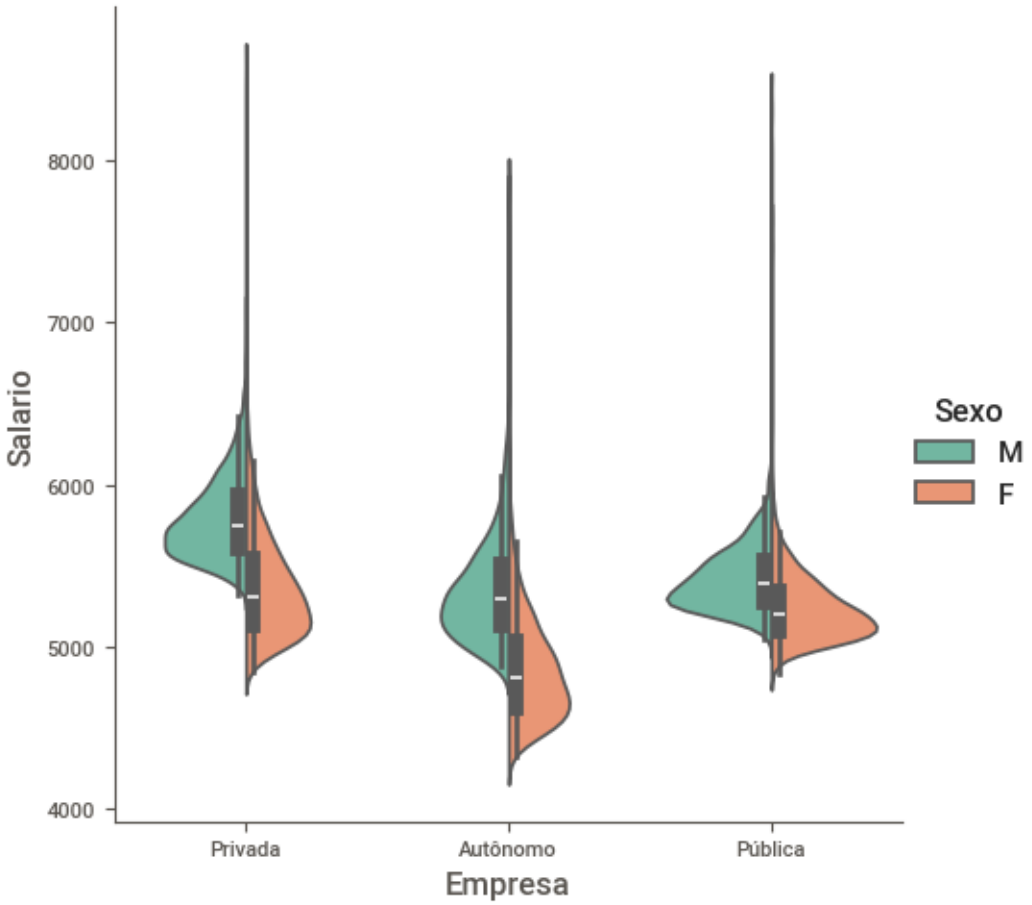



```
In [121]: ax = sns.catplot(x='Sexo', y='Salario', hue='Empresa', kind='violin', data=dados, pale
```




```
In [122]: sns.catplot(x='Empresa', y='Salario', hue='Sexo', kind='violin', split=True, data=dado
```

```
Out[122]: <seaborn.axisgrid.FacetGrid at 0x7dcb633a3400>
```

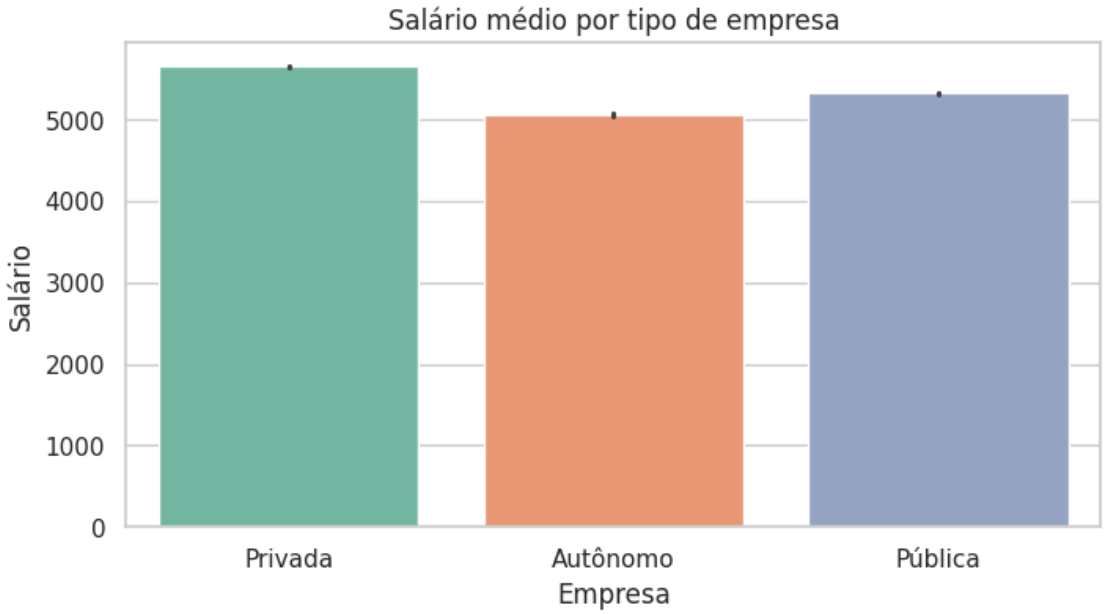



```
In [123]: # Salário médio por tipo de empresa

 sns.set_theme(style="whitegrid")

 # Estabelecendo o tamanho do gráfico
 plt.figure(figsize=(8,4))

 # Título
 plt.title("Salário médio por tipo de empresa")

 # Gráfico de barras com salário médio por tipo de empresa
 sns.barplot(x='Empresa', y='Salario', data=dados, palette='Set2')
 #sns.barplot(x='Empresa', y='Salario', hue='Sexo', data=dados, palette='Set2')
```

```
 # Label para eixo vertical
 plt.ylabel("Salário");
```

```
<ipython-input-123-eb7ff1310357>:12: FutureWarning:
```

```
Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign t
```

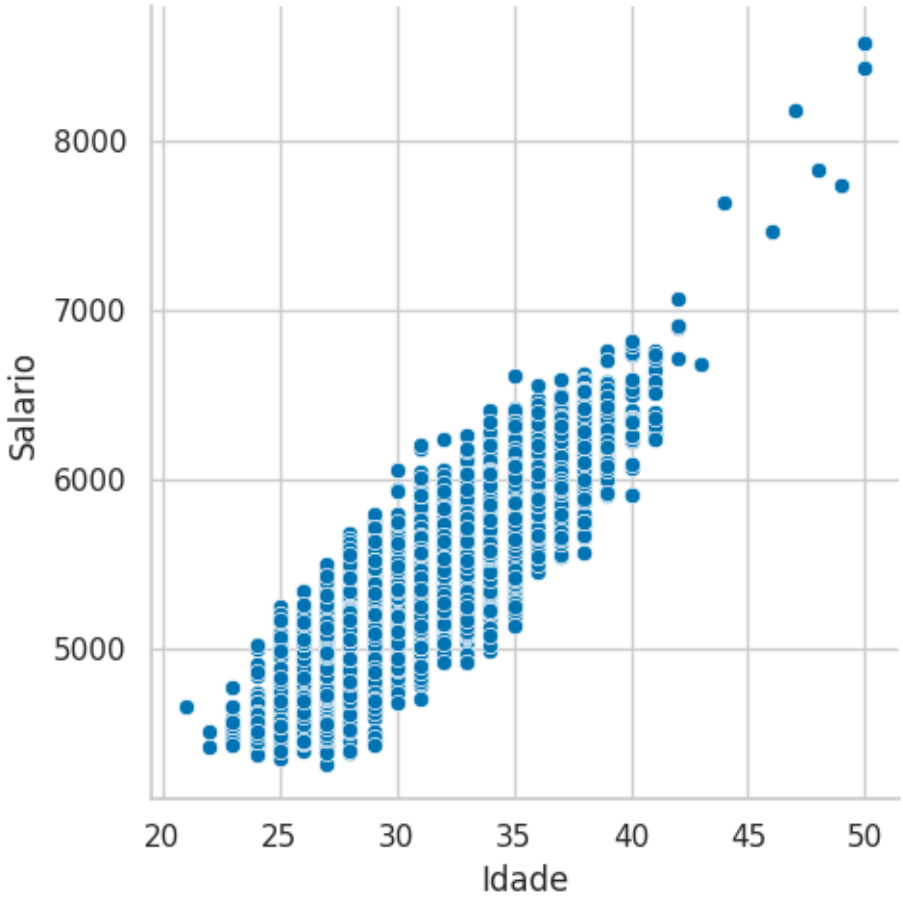


1.9 Associação entre variáveis quantitativas

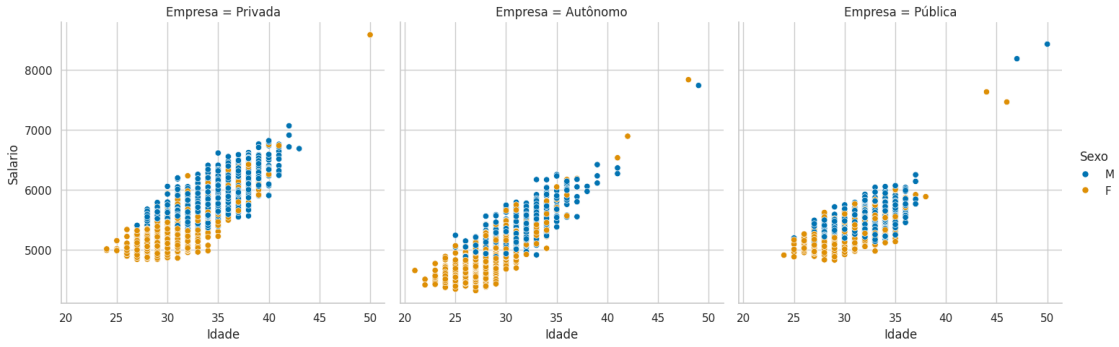
Gráfico de dispersão

```
In [124]: sns.set_palette('colorblind')
 sns.relplot(x='Idade', y='Salario', data=dados)
```

```
Out[124]: <seaborn.axisgrid.FacetGrid at 0x7dcb6353ed40>
```

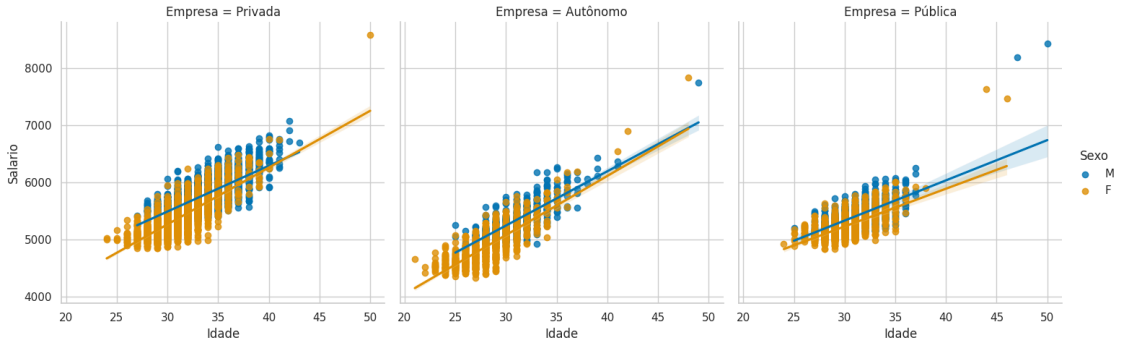



```
In [125]: sns.set_palette('colorblind')
 sns.relplot(x='Idade', y='Salario', hue='Sexo', col='Empresa',  data=dados);
```

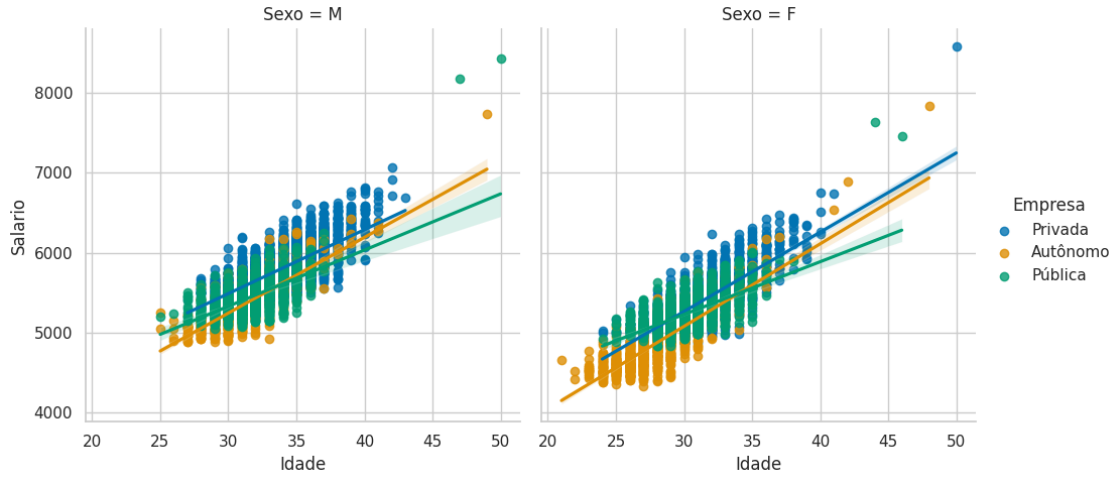


Gráficos com Regressão

```
In [126]: sns.lmplot(x='Idade', y='Salario', hue='Sexo', col='Empresa',  data=dados, aspect=1, c
```

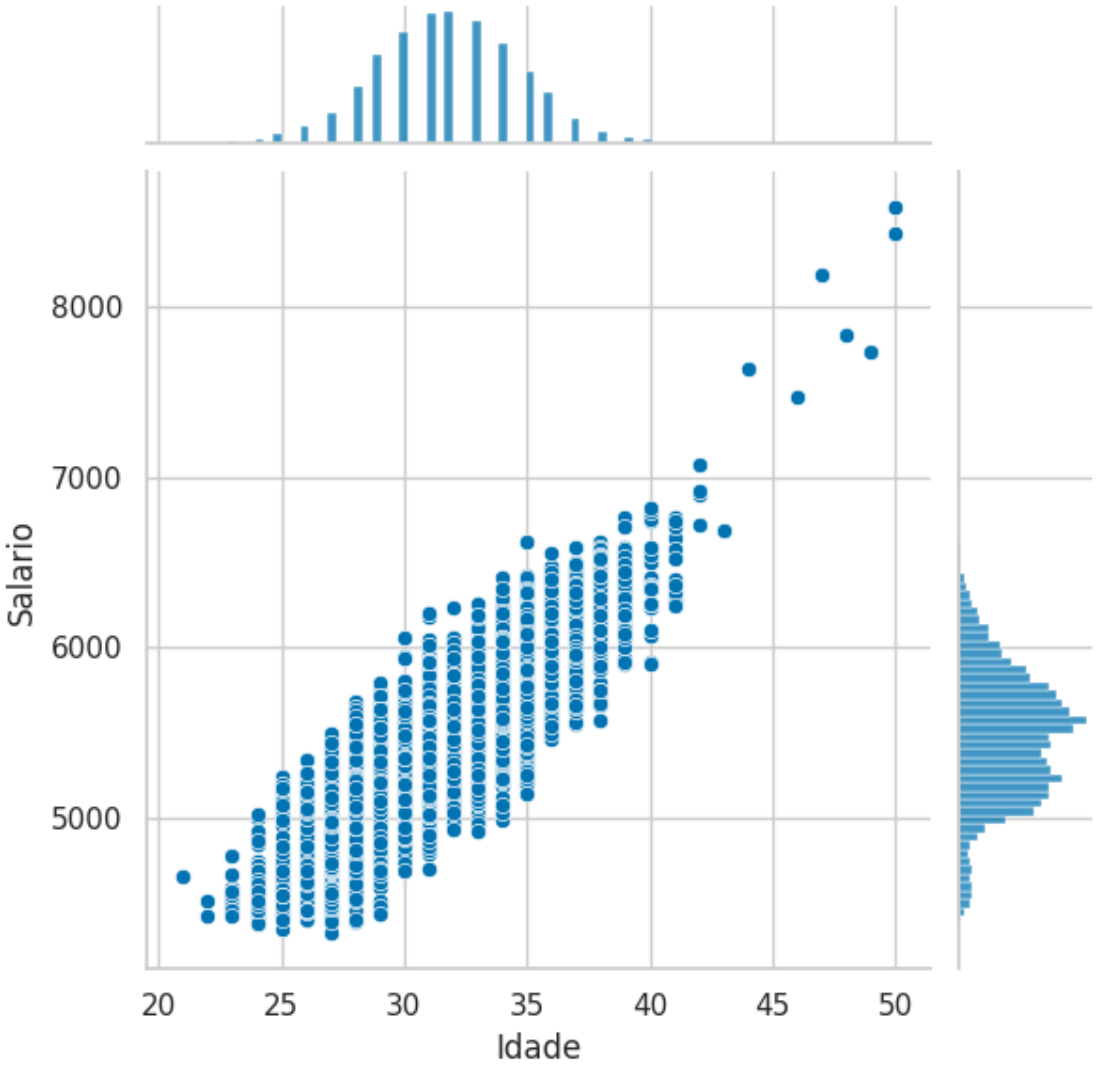



```
In [127]: sns.lmplot(x='Idade', y='Salario', hue='Empresa', col='Sexo',  data=dados, aspect=1, c
```

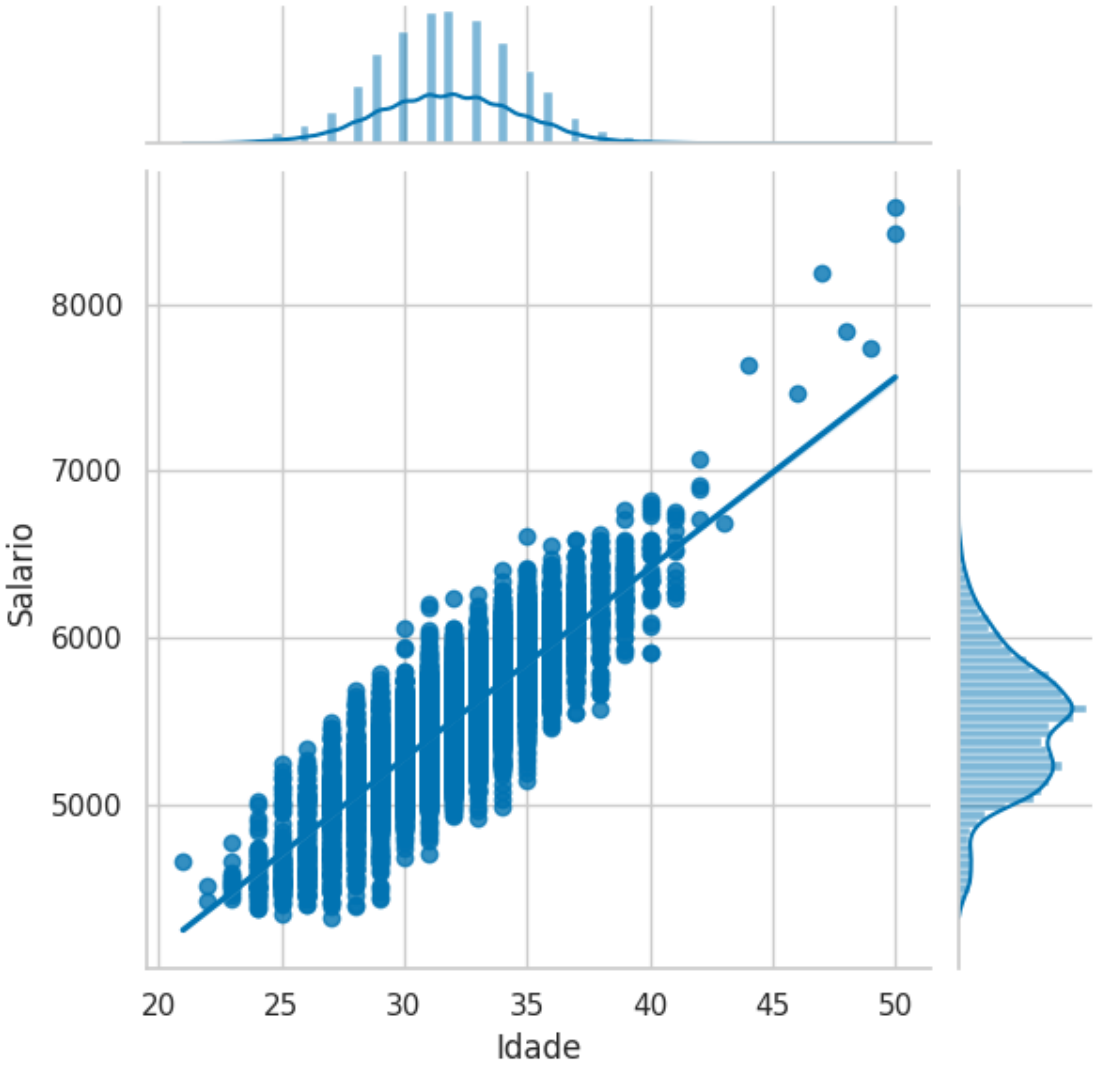


Joint plot

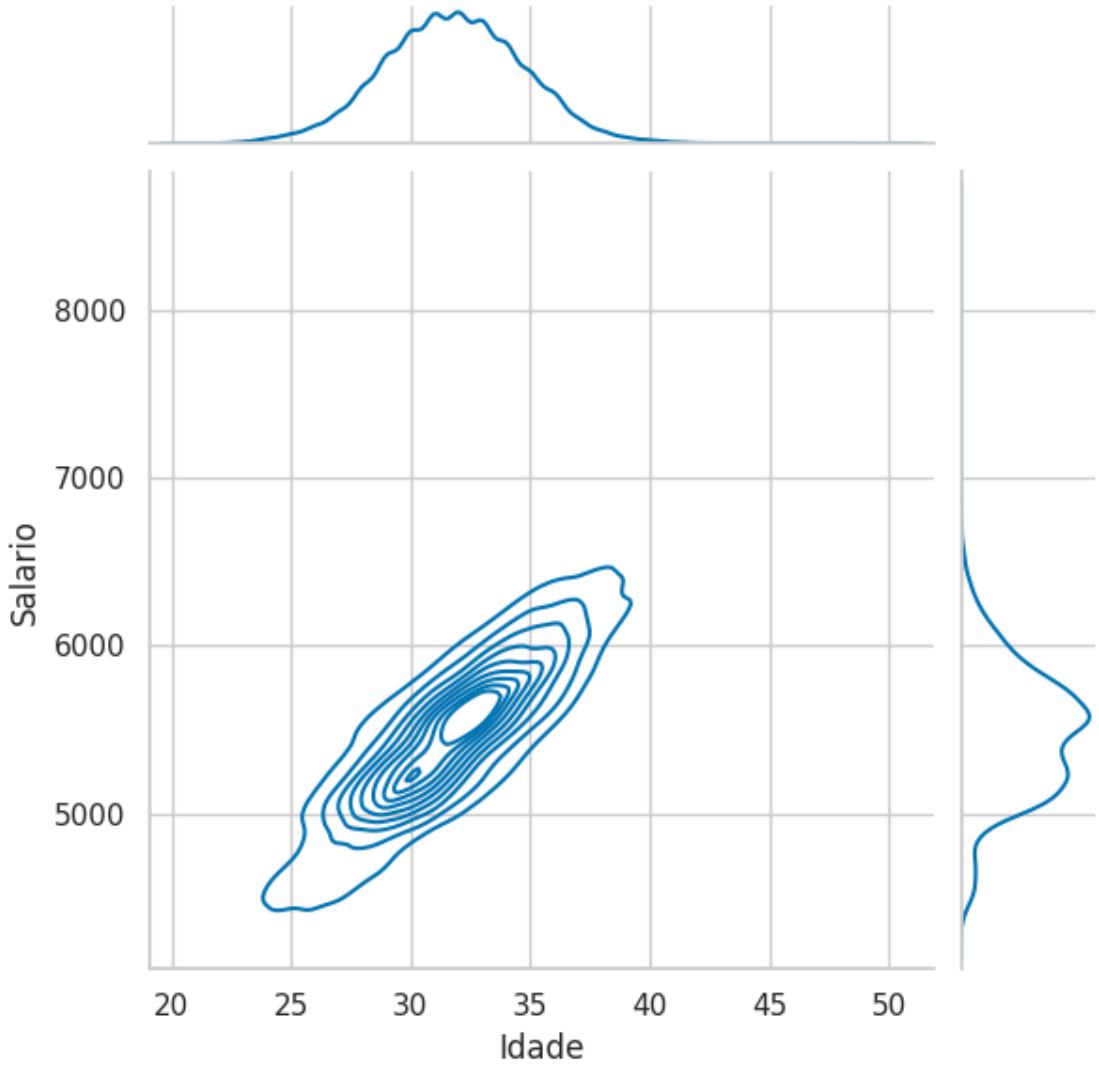
```
In [128]: sns.jointplot(x='Idade', y='Salario',   data=dados);
```




```
In [129]: sns.jointplot(x='Idade', y='Salario', kind='reg',  data=dados);
```




```
In [130]: sns.jointplot(x='Idade', y='Salario', kind='kde',  data=dados);
```



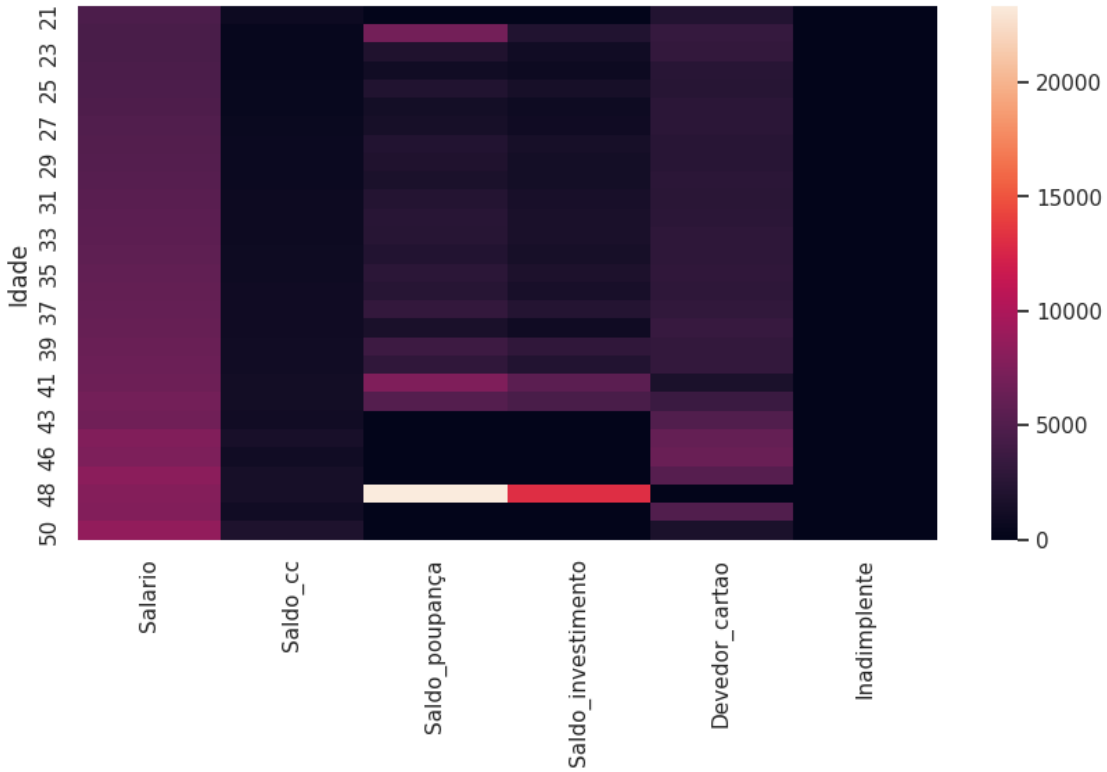
Coeficiente de correlação de Pearson

```
In [131]: #rom scipy.stats import pearsonr
 #pearsonr(dados['Idade'], dados['Salario'])

 # Em google colab use corrcoef

 np.corrcoef(dados['Idade'], dados['Salario'])[0,1]
```

```
Out[131]: 0.8506660825874651
```

1.10 Heatmap (mapa de calor)

```
In [132]: dados = pd.read_csv('https://raw.githubusercontent.com/cibelerusso/Estatistica-Ciencia

In [133]: dados_heatmap = dados_heatmap = dados.drop(['Sexo','Empresa'],axis=1).groupby(by='Idad

 dados_heatmap.head()

 # Estabelecendo o tamanho do gráfico
 plt.figure(figsize=(10,5))

 ax = sns.heatmap(dados_heatmap)
 # ax = sns.heatmap(dados_heatmap, cmap="BuPu")
```



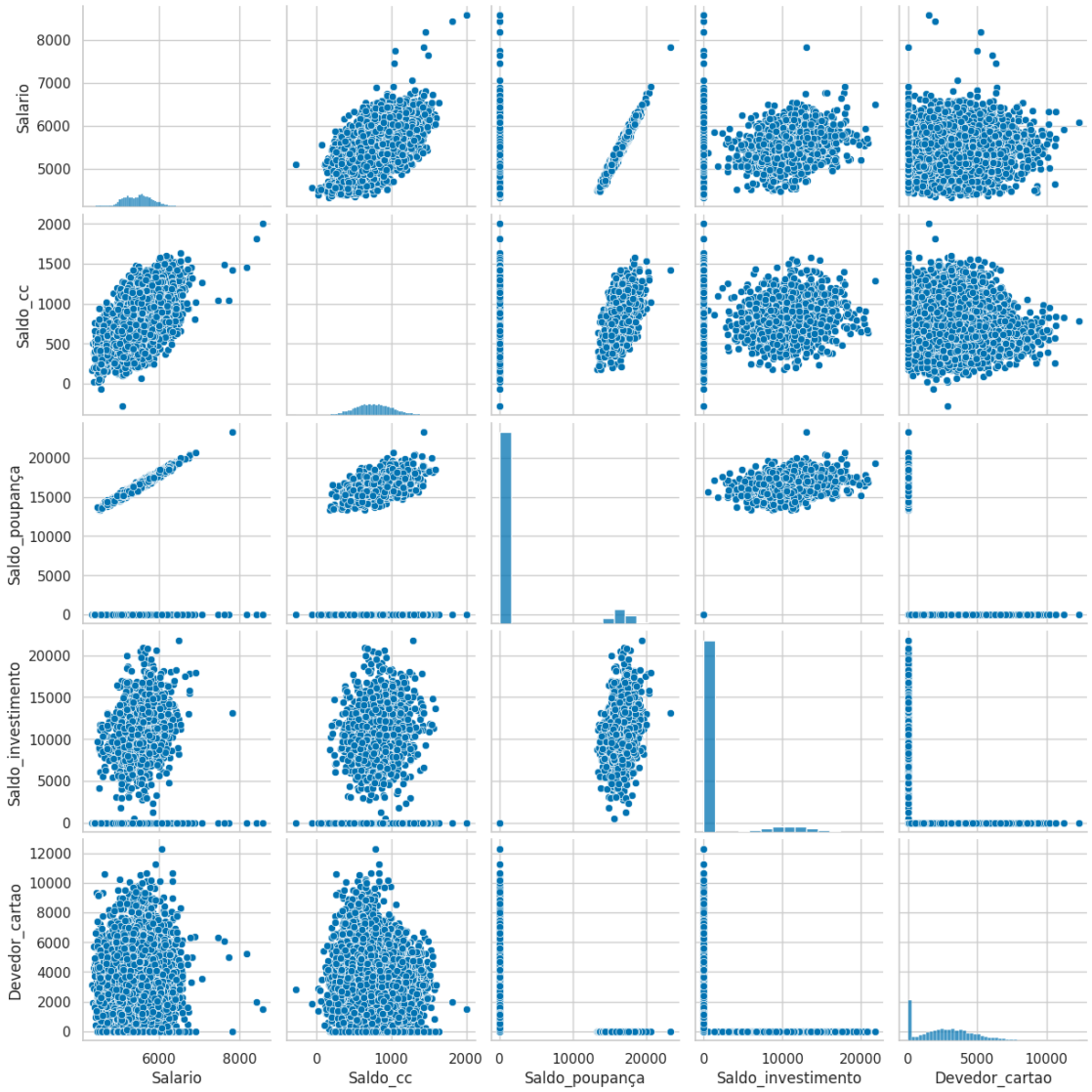

```
In [134]: dados

Out[134]:Sexo  Idade   Empresa  Salario  Saldo_cc  Saldo_poupança  \
 Cliente
 75928       M     32   Privada  5719.00    933.79     0.0
```

```
52921      F     28   Privada  5064.00    628.37    0.0
8387       F     24  Autônomo  4739.00    889.18    0.0
54522      M     30   Pública  5215.00   1141.47    0.0
45397      M     30  Autônomo  5215.56    520.70    0.0
...      ...    ...       ...      ...       ...    ...
33487      F     31   Pública  5016.00    498.96    0.0
71360      M     29   Pública  5329.00   1142.82    0.0
92455      M     34   Privada  5581.00    885.34    0.0
61296      F     28   Privada  5061.00    660.74    0.0
52862      M     33  Autônomo  5519.00   1147.71    0.0

Saldo_investimento  Devedor_cartao  Inadimplente
Cliente
75928 0.06023.68    0
52921 0.01578.24    0
8387  0.02578.70    0
54522 0.04348.96    0
45397 0.01516.78    1
...   ...    ...  ...
33487 0.01263.34    0
71360 0.05613.71    0
92455 0.01199.22    0
61296 0.01152.97    0
52862 0.04684.66    0

[10000 rows x 9 columns]
```

1.10.1 Gráficos multivariados

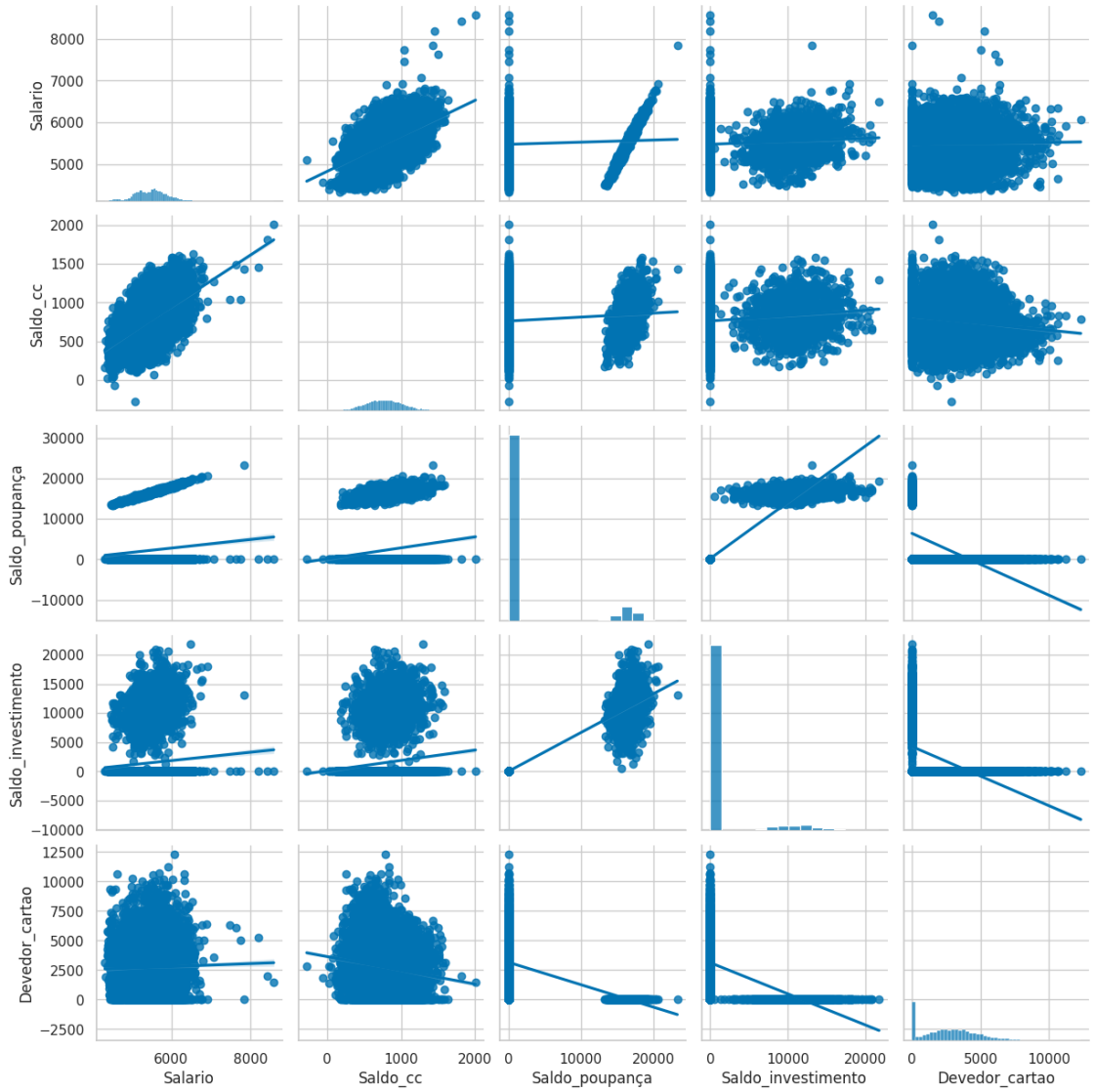
```
In [135]: sns.pairplot(dados[['Salario','Saldo_cc', 'Saldo_poupança', 'Saldo_investimento', 'Dev
```

```
Out[135]: <seaborn.axisgrid.PairGrid at 0x7dcb703973a0>
```

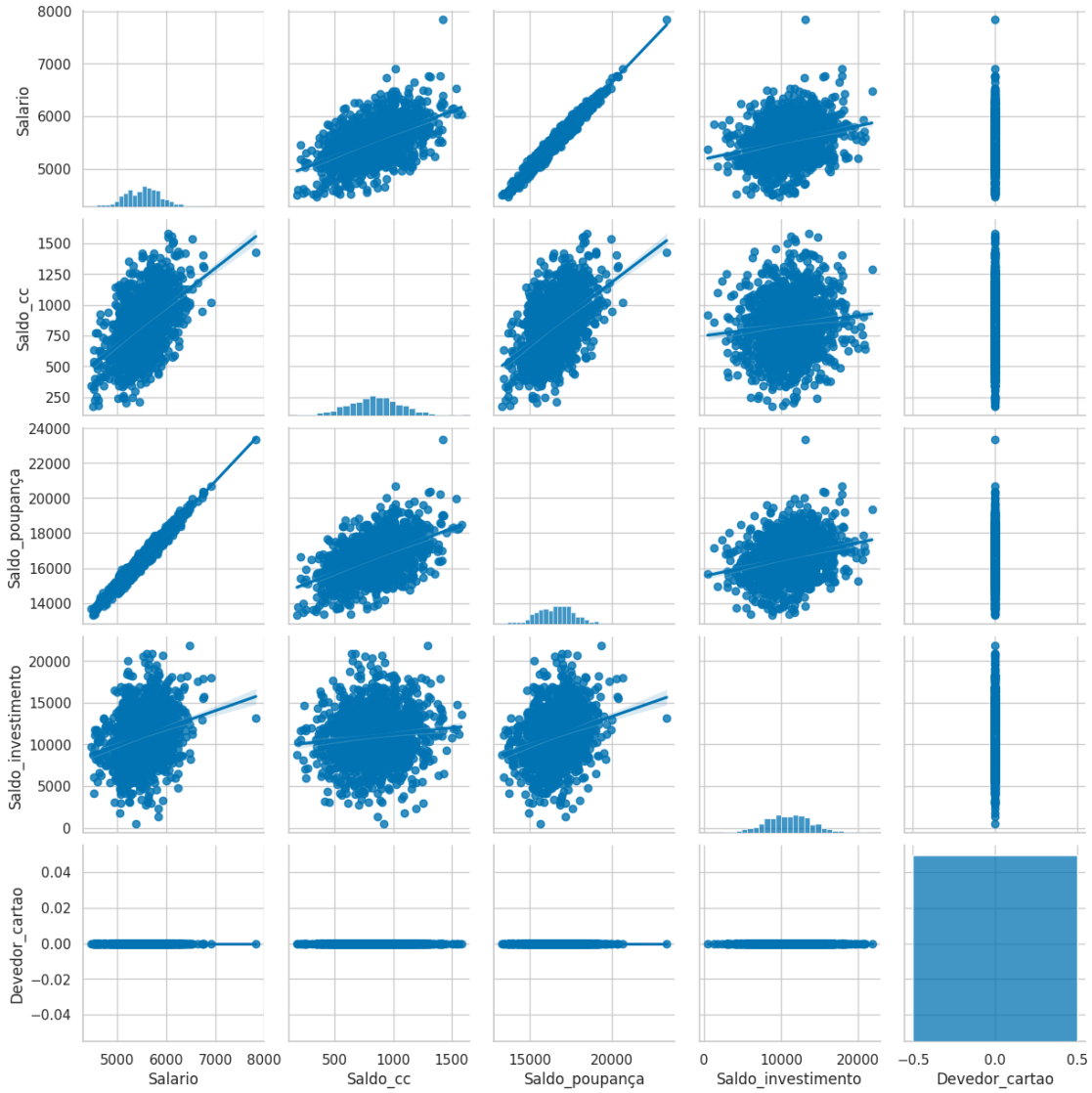



```
In [136]: sns.pairplot(dados[['Salario','Saldo_cc', 'Saldo_poupança', 'Saldo_investimento', 'Dev
```

```
Out[136]: <seaborn.axisgrid.PairGrid at 0x7dcb814e0c70>
```




```
In [137]: dados_nozeros = dados[dados['Saldo_investimento']*dados['Saldo_poupança']!=0]

In [138]: sns.pairplot(dados_nozeros[['Salario','Saldo_cc', 'Saldo_poupança', 'Saldo_investiment

Out[138]: <seaborn.axisgrid.PairGrid at 0x7dcb5fff9180>
```



1.11 Agrupamento de dados

- Agrupamento hierárquico (dendrograma)
- Agrupamento não-hierárquico (k-médias)

Referências:

Aulas no contexto de Análise Multivariada e Aprendizado Não-supervisionado (Profa. Cibele Russo):

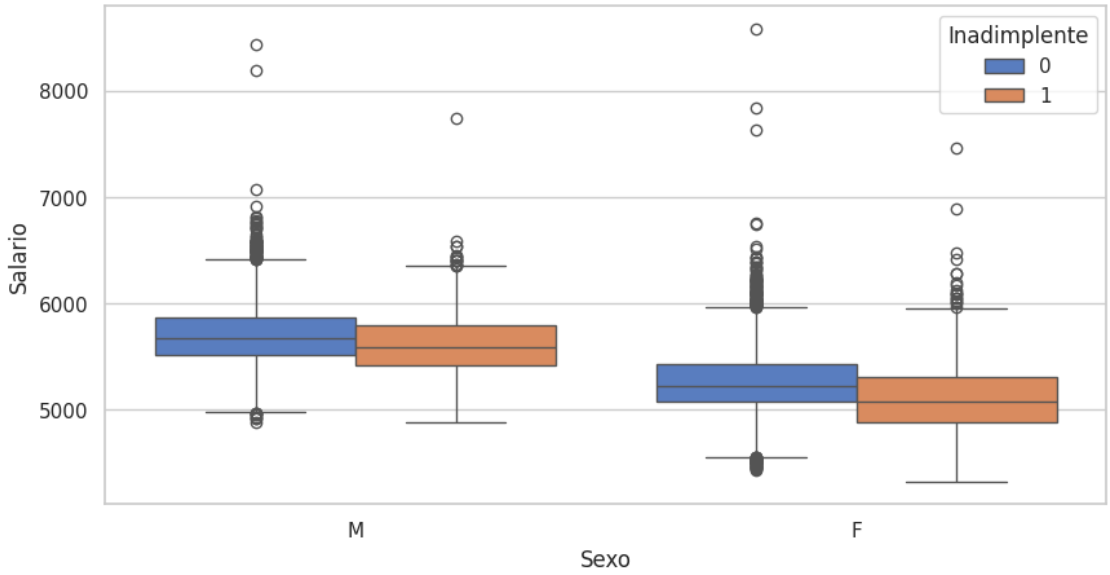
- Análise de Agrupamentos: https://youtu.be/zyLDAnQMnbo

- Análise de Agrupamentos - Um exemplo passo a passo: https://youtu.be/Re97VX6ZhPA
- Análise de Agrupamentos - Aplicação em Python: https://youtu.be/d_CJGaAbC7o

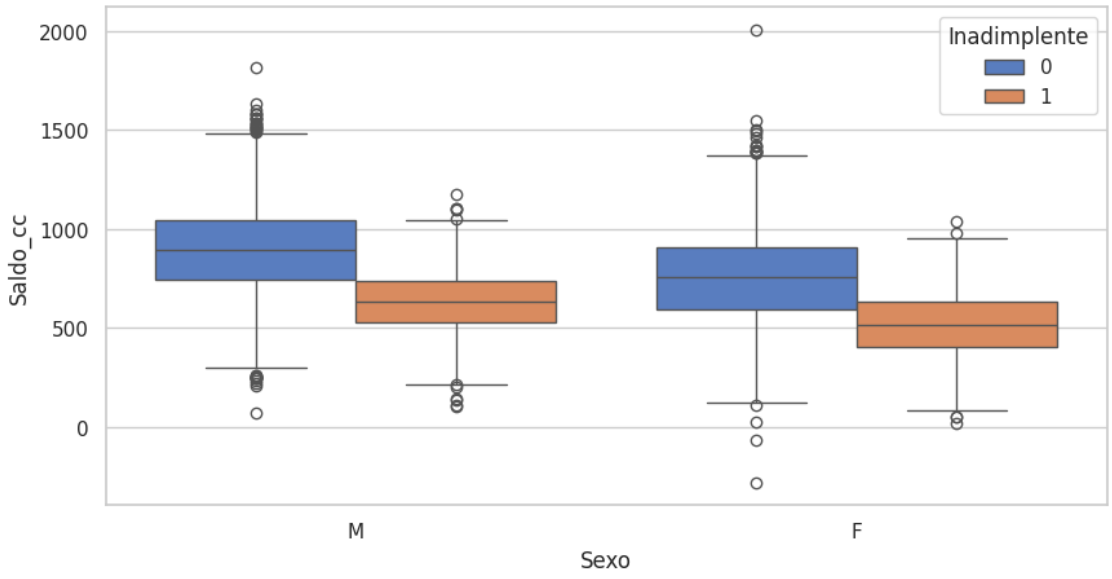
Exercício

Analise as possíveis associações entre o sexo, idade, empresa, salário, saldo em conta corrente, saldo em conta poupança, saldo em investimento e devedor no cartão com a variável Inadimplente.

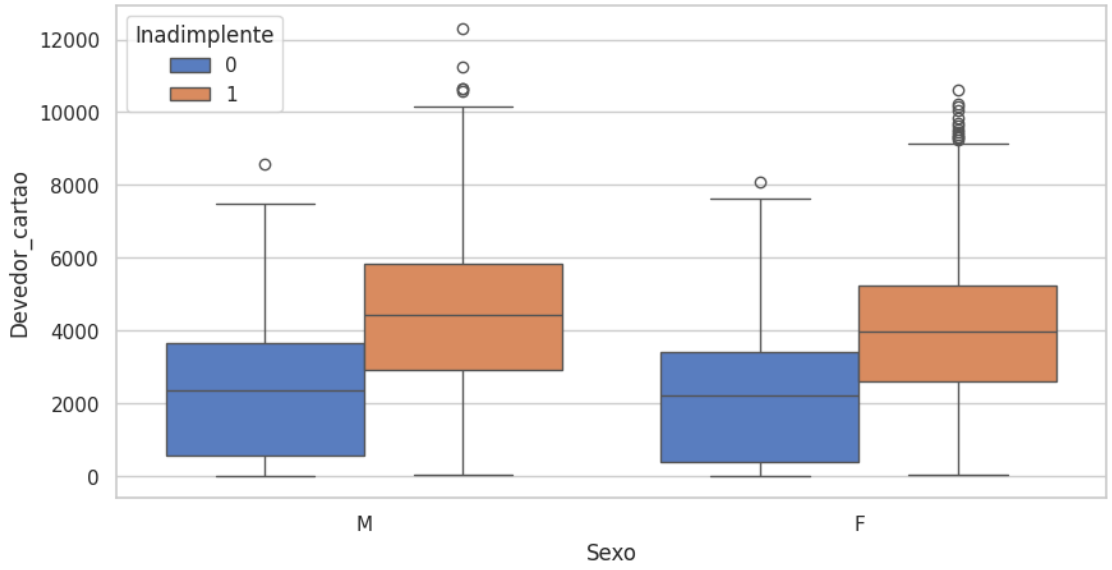
```
In [139]: ax = sns.boxplot(x='Sexo', y='Salario', hue='Inadimplente', data=dados, palette='muted
```




```
In [140]: ax = sns.boxplot(x='Sexo', y='Saldo_cc', hue='Inadimplente', data=dados, palette='mute
```

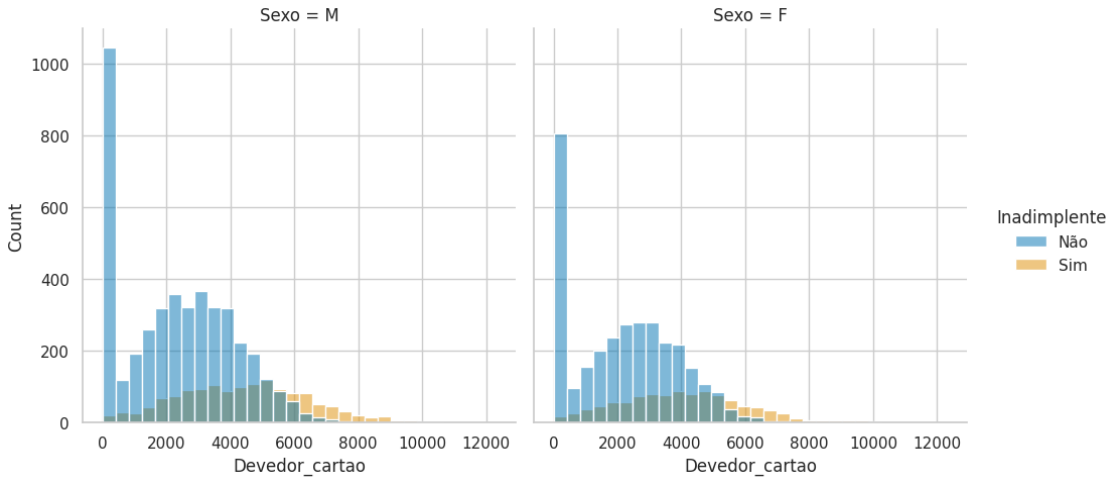



```
In [141]: ax = sns.boxplot(x='Sexo', y='Devedor_cartao', hue='Inadimplente', data=dados, palette
```




```
In [142]: dados.loc[dados['Inadimplente']==0, 'Inadimplente']= 'Não'
 dados.loc[dados['Inadimplente']==1, 'Inadimplente']= 'Sim'
```

```
In [143]: sns.displot(dados, x='Devedor_cartao', col='Sexo', hue='Inadimplente', bins=30);
```


Sexo = M
Sexo = F
Count
Devedor_cartao
Devedor_cartao
Inadimplente
Não
Sim