

MBA em Inteligência Artificial

Big Data e Visualização de Dados

Prof. Diego Nogare



Faculdade de Computação e Informática
Mackenzie

Diego Nogare



Former MVP Artificial Intelligence
Microsoft Regional Director
Diretor no PASS.org
Chief Data Officer @ Lambda3
Membro notável na I2AI
Mestre em IA

www.diegonogare.net
www.livrosdonogare.com.br





Pacote Matplotlib

Uso das funções do Python para gerar gráficos

■ Gráficos no Matplotlib – Criação e Edição

Além da versatilidade de criar aplicações de ponta a ponta, muitos cientistas de dados usam Python pelas suas capacidades gráficas simples.

```
import pandas as pd

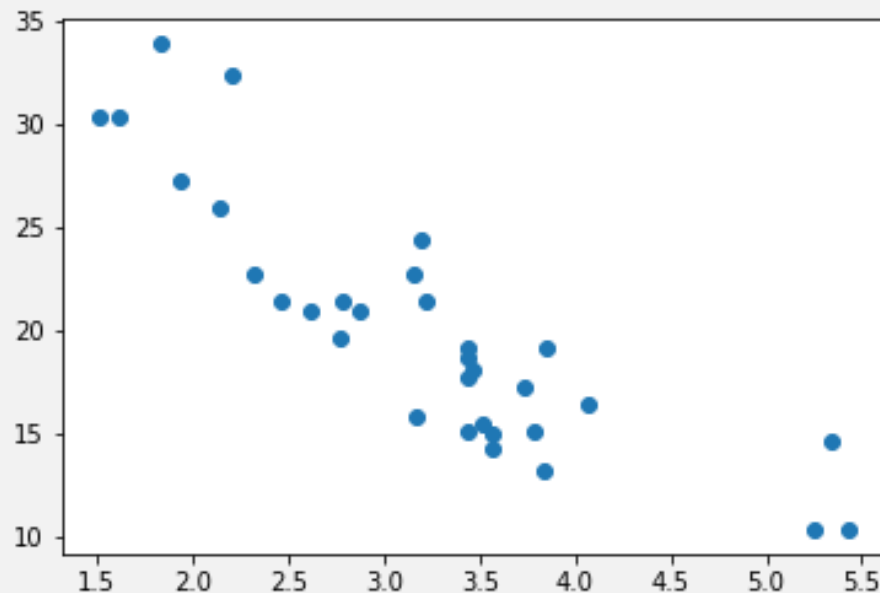
import matplotlib.pyplot as plt
%matplotlib inline

plt.scatter(df['wt'],df['mpg']);
```

■ Gráficos no Matplotlib – Criação e Edição

Além da versatilidade de criar aplicações de ponta a ponta, muitos cientistas de dados usam Python pelas suas capacidades gráficas simples.

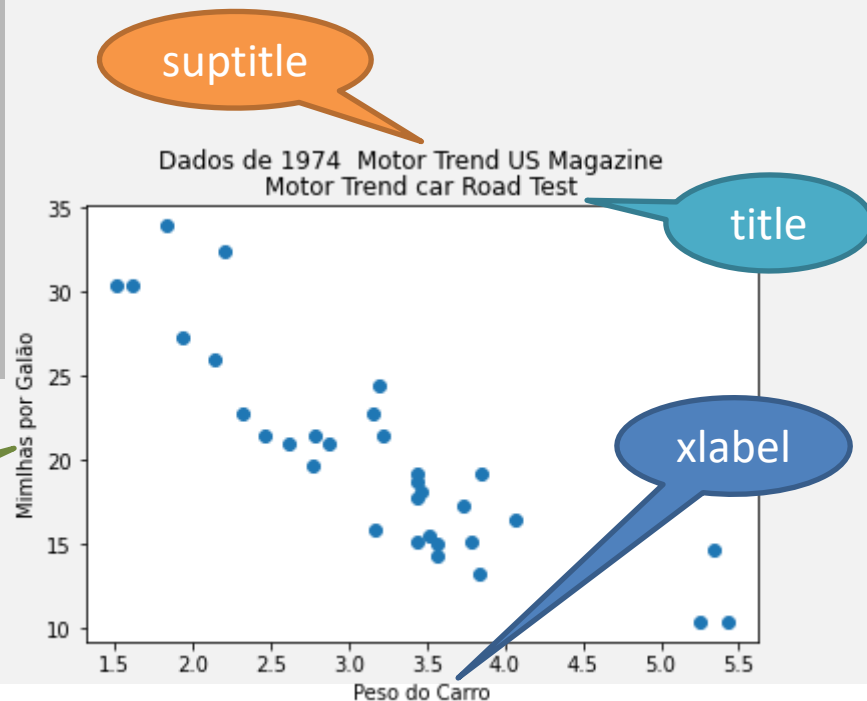
```
plt.scatter(df['wt'],df['mpg']);
```



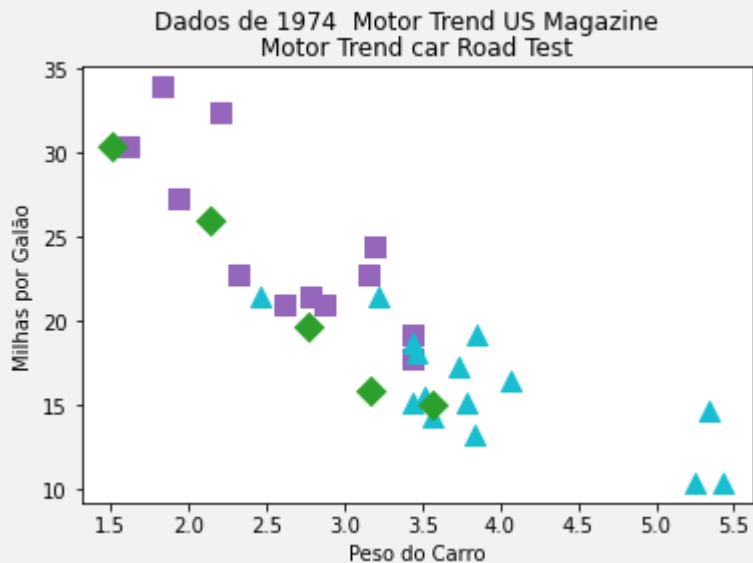
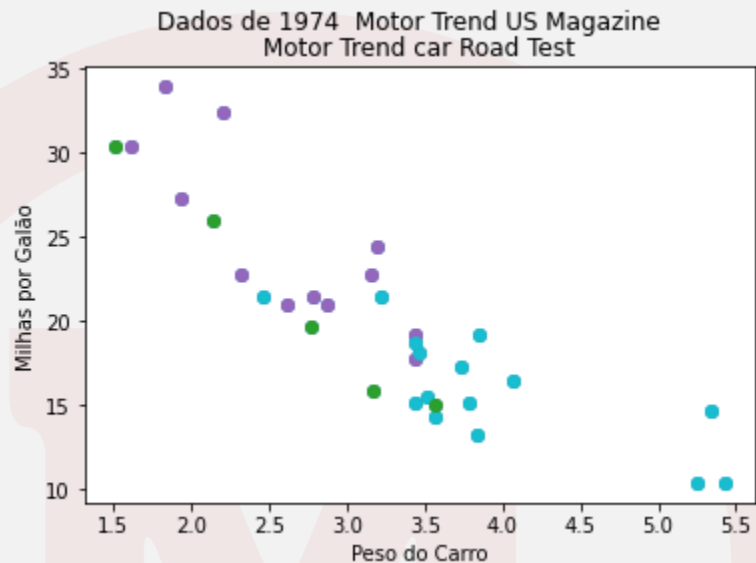
■ Gráficos no Python

```
plt.figure(figsize=(15,8))
plt.scatter(df['wt'],df['mpg']);

plt.title("Motor Trend car Road Test")
plt.suptitle("Dados de 1974 Motor Trend US Magazine")
plt.xlabel("Peso do Carro")
plt.ylabel("Milhas por Galão")
```



Gráficos no Python



```
cores = {3:'tab:cyan',4:'tab:purple',5:'tab:green'}
marcadores = {3:'^',4:'s',5:'D'}

for cambio in df['gear'].unique():
    tmp = df[df['gear'] == cambio]
    plt.scatter(tmp['wt'], tmp['mpg'], color=cores[cambio],
                marker=marcadores[cambio], s=100)

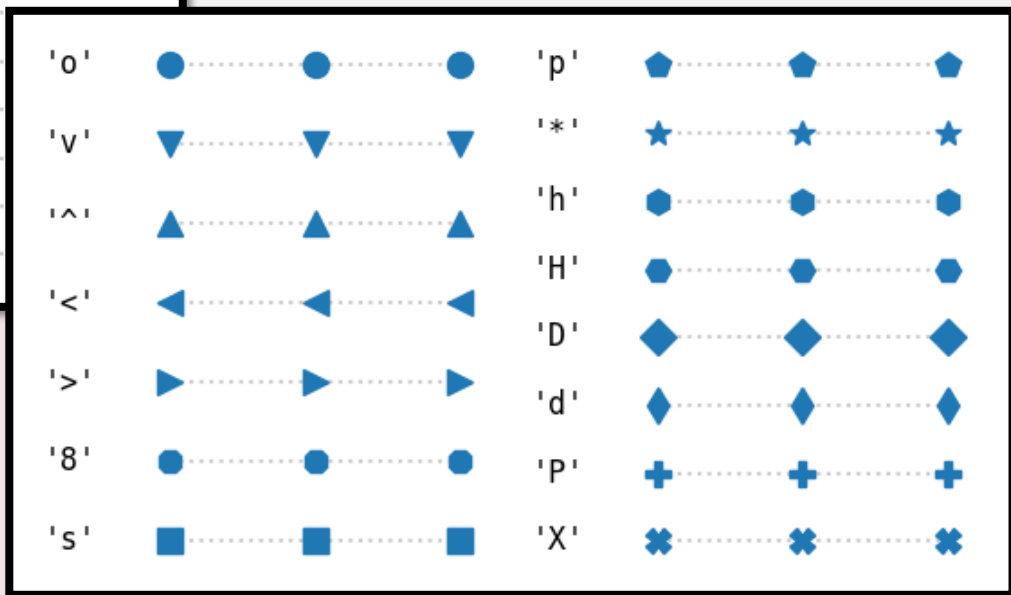
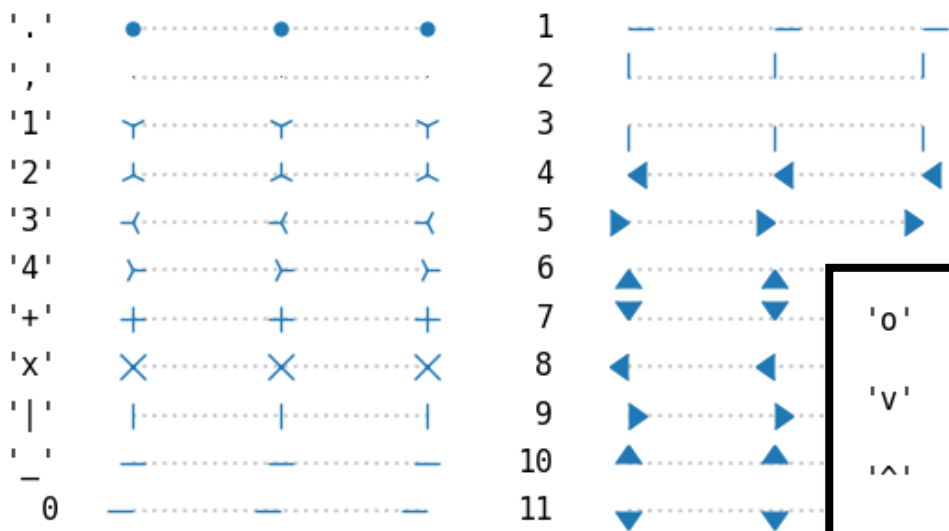
plt.title("Motor Trend car Road Test")
plt.suptitle("Dados de 1974 Motor Trend US Magazine")
plt.xlabel("Peso do Carro")
plt.ylabel("Milhas por Galão")
```

Tupla de
cores

Tupla de
marcadores

uso de
cores

uso de
marcadores



https://matplotlib.org/api/markers_api.html?highlight=marker

CSS Colors

black	bisque	forestgreen	slategrey
dimgray	darkorange	limegreen	lightsteelblue
dimgray	burlywood	darkgreen	cornflowerblue
gray	antiquewhite	green	royalblue
grey	tan	lime	ghostwhite
darkgray	navajowhite	seagreen	lavender
darkgrey	blanchedalmond	mediumseagreen	midnightblue
silver	papayawhip	springgreen	navy
lightgray	moccasin	mintcream	darkblue
lightgrey	orange	mediumspringgreen	mediumblue
gainsboro	wheat	mediumaquamarine	blue
whitesmoke	oldlace	aquamarine	slateblue
white	floralwhite	turquoise	darkslateblue
snow	darkgoldenrod	lightseagreen	mediumslateblue
rosybrown	goldenrod	mediumturquoise	mediumpurple
lightcoral	cornsilk	azure	rebeccapurple
indianred	gold	lightcyan	blueviolet
brown	lemonchiffon	paleturquoise	indigo
firebrick	khaki	darkslategray	darkorchid
maroon	palegoldenrod	darkslategrey	darkviolet
darkred	darkkhaki	teal	mediumorchid
red	ivory	darkcyan	thistle
mistyrose	beige	aqua	plum
salmon	lightyellow	cyan	violet
tomato	lightgoldenrodyellow	darkturquoise	purple
darksalmon	olive	cadetblue	darkmagenta
coral	yellow	powderblue	fuchsia
orangered	olivedrab	lightblue	magenta
lightsalmon	yellowgreen	deepskyblue	orchid
sienna	darkolivegreen	skyblue	mediumvioletred
seashell	greenyellow	lightskyblue	deeppink
chocolate	chartreuse	steelblue	hotpink
saddlebrown	lawngreen	aliceblue	lavenderblush
sandybrown	honeydew	dodgerblue	palevioletred
peachpuff	darkseagreen	lightslategray	crimson
peru	palegreen	lightslategrey	pink
linen	lightgreen	slategray	lightpink

Base Colors

b	c	k
g	m	w
r	y	

Tableau Palette

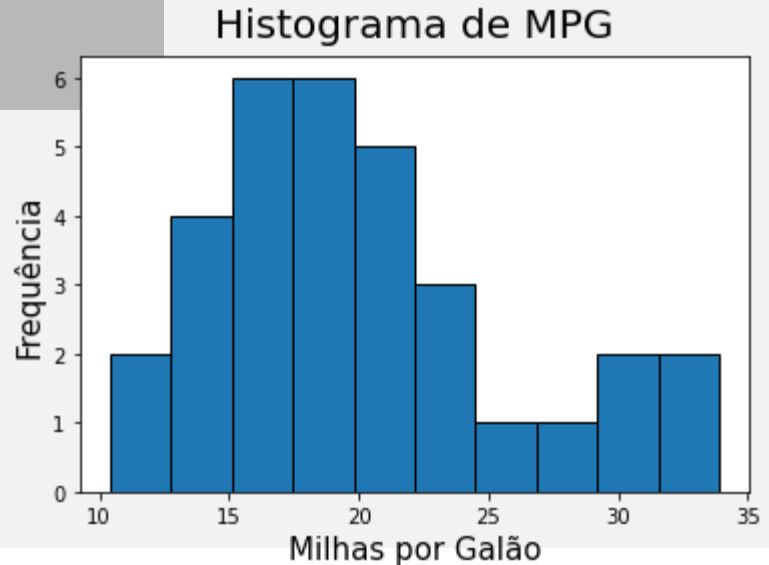
tab:blue	tab:brown
tab:orange	tab:pink
tab:green	tab:gray
tab:red	tab:olive
tab:purple	tab:cyan

https://matplotlib.org/gallery/color/named_colors.html

Tamanho da fonte

```
plt.hist(df['mpg'], edgecolor="black");  
  
plt.title("Histograma de MPG", fontsize=20, pad=10)  
plt.xlabel("Milhas por Galão", fontsize=15)  
plt.ylabel("Frequência", fontsize=15)
```

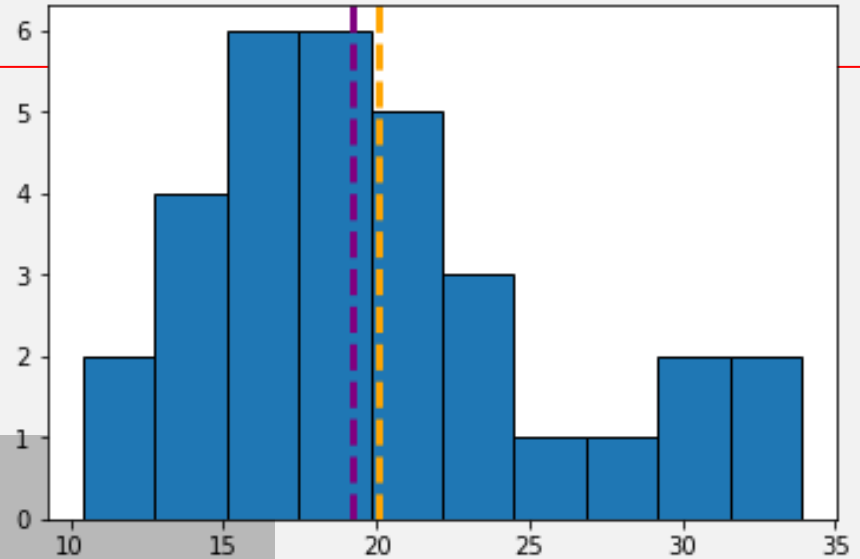
afastamento



```
cores = ["orange","purple"]
```

```
plt.hist(df['mpg'], edgecolor="black");  
media, mediana = df['mpg'].mean(), df['mpg'].median()
```

```
plt.axvline(media, color=cores[0], lw=3, ls="--")  
plt.axvline(mediana, color=cores[1], lw=3, ls="--")
```

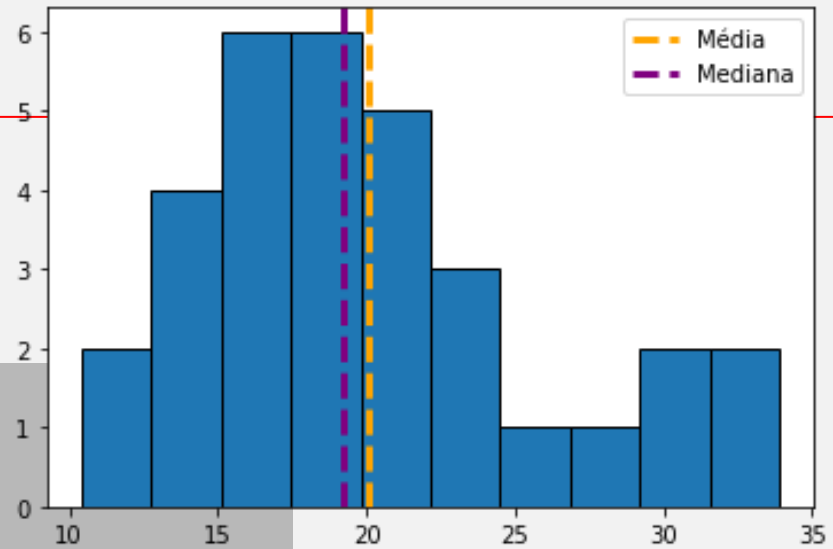


```
cores = ["orange","purple"]
legenda = ["Média","Mediana"]

plt.hist(df['mpg'], edgecolor="black");
media, mediana = df['mpg'].mean(), df['mpg'].median()

plt.axvline(media, color=cores[0], lw=3, ls="--")
plt.axvline(mediana, color=cores[1], lw=3, ls="--")

plt.legend(legenda, loc="upper right")
```

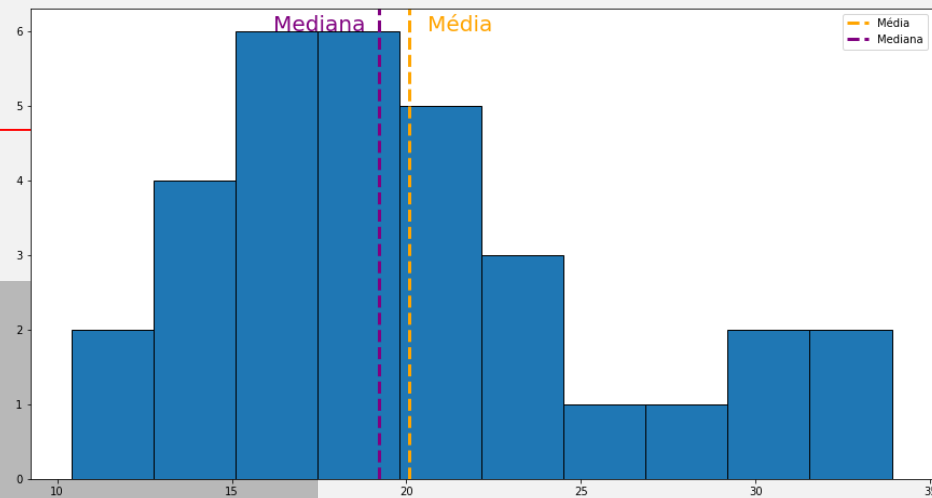


```
cores = ["orange","purple"]  
legenda = ["Média","Mediana"]
```

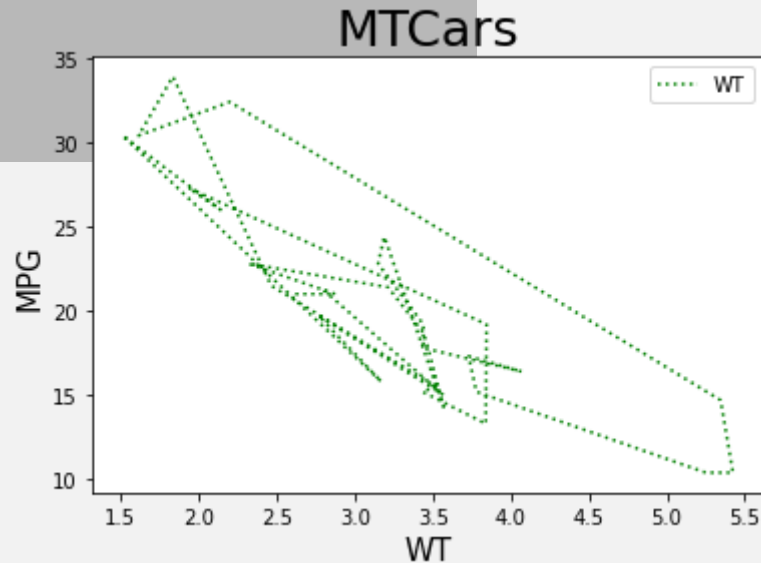
```
plt.hist(df['mpg'], edgecolor="black");  
media, mediana = df['mpg'].mean(), df['mpg'].median()  
plt.axvline(media, color=cores[0], lw=3, ls="--")  
plt.axvline(mediana, color=cores[1], lw=3, ls="--")
```

```
plt.legend(legenda, loc="upper right")
```

```
plt.annotate(legenda[0], xy=(media + 0.5, 6),  
            color=cores[0], fontsize=20)  
plt.annotate(legenda[1], xy=(mediana - 3, 6),  
            color=cores[1], fontsize=20)
```



```
plt.plot(df['wt'], df['mpg'], color="green", ls = "dotted")
plt.title("MTCars", fontsize=25)
plt.xlabel("WT", fontsize=15)
plt.ylabel("MPG", fontsize=15)
plt.legend(["WT"], loc="upper right")
```

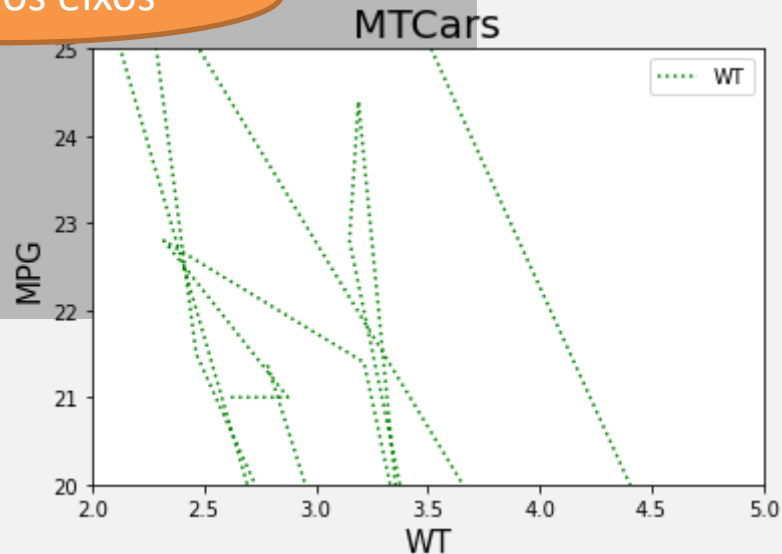


```
plt.plot(df['wt'], df['mpg'], color="green", ls = "dotted")
```

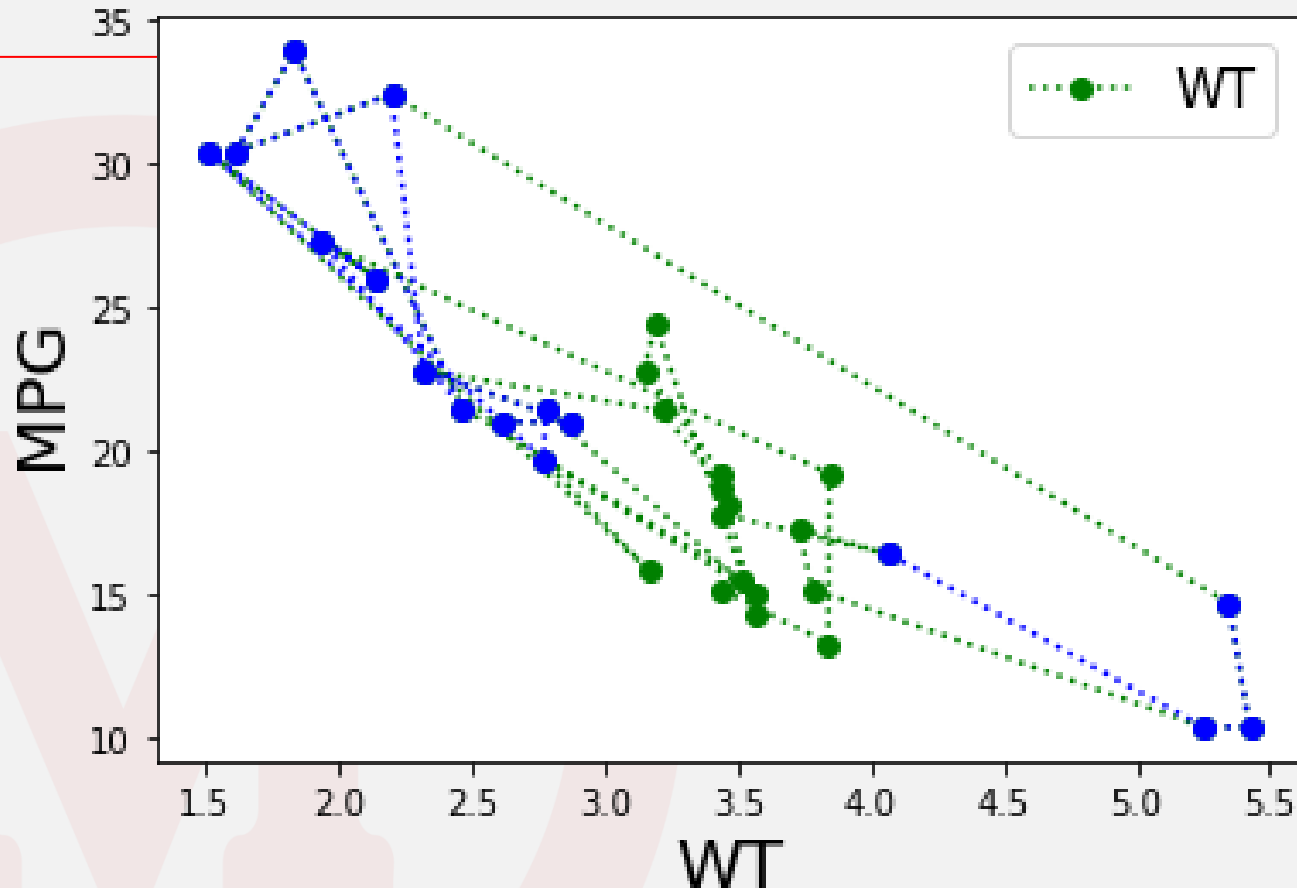
```
plt.xlim(2,5)  
plt.ylim(20,25)
```

Limite dos eixos

```
plt.title("MTCars", fontsize=20)  
plt.xlabel("WT", fontsize=15)  
plt.ylabel("MPG", fontsize=15)  
plt.legend(["WT"], loc="upper right")
```



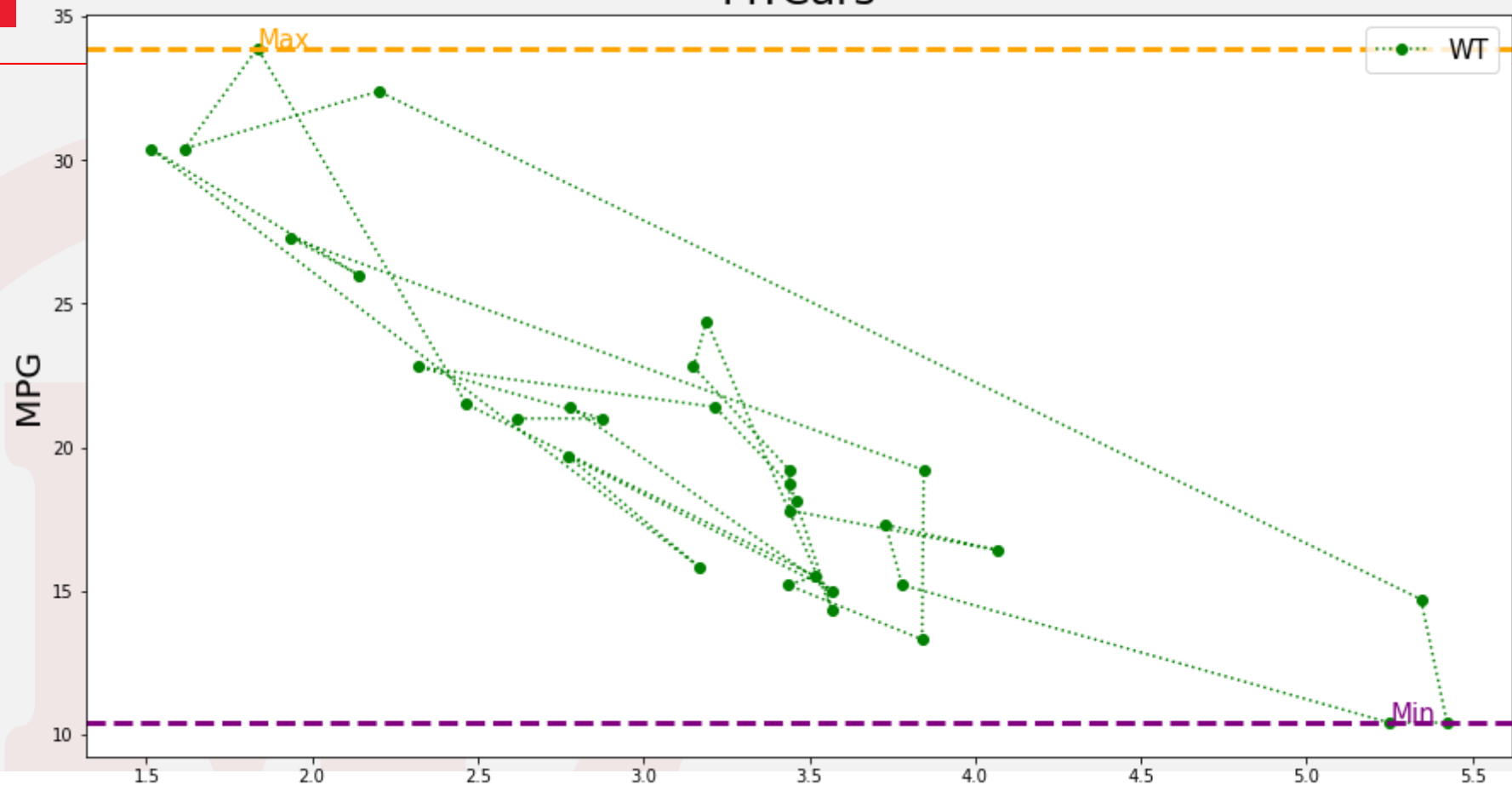
MTCars



```
menor_3 = df["wt"] < 3
maior_4 = df["wt"] > 4

plt.plot(df['wt'], df['mpg'], color="green", ls = "dotted",
         , marker = "o")
plt.plot(df[menor_3]['wt'],df[menor_3]['mpg'],color="blue",
         , ls = "dotted", marker = "o")
plt.plot(df[maior_4]['wt'],df[maior_4]['mpg'],color="blue",
         , ls = "dotted", marker = "o")
plt.title("MTCars", fontsize=25)
plt.xlabel("WT", fontsize=20)
plt.ylabel("MPG", fontsize=20)
plt.legend(["WT"], loc="upper right", fontsize=15)
```

MTCars



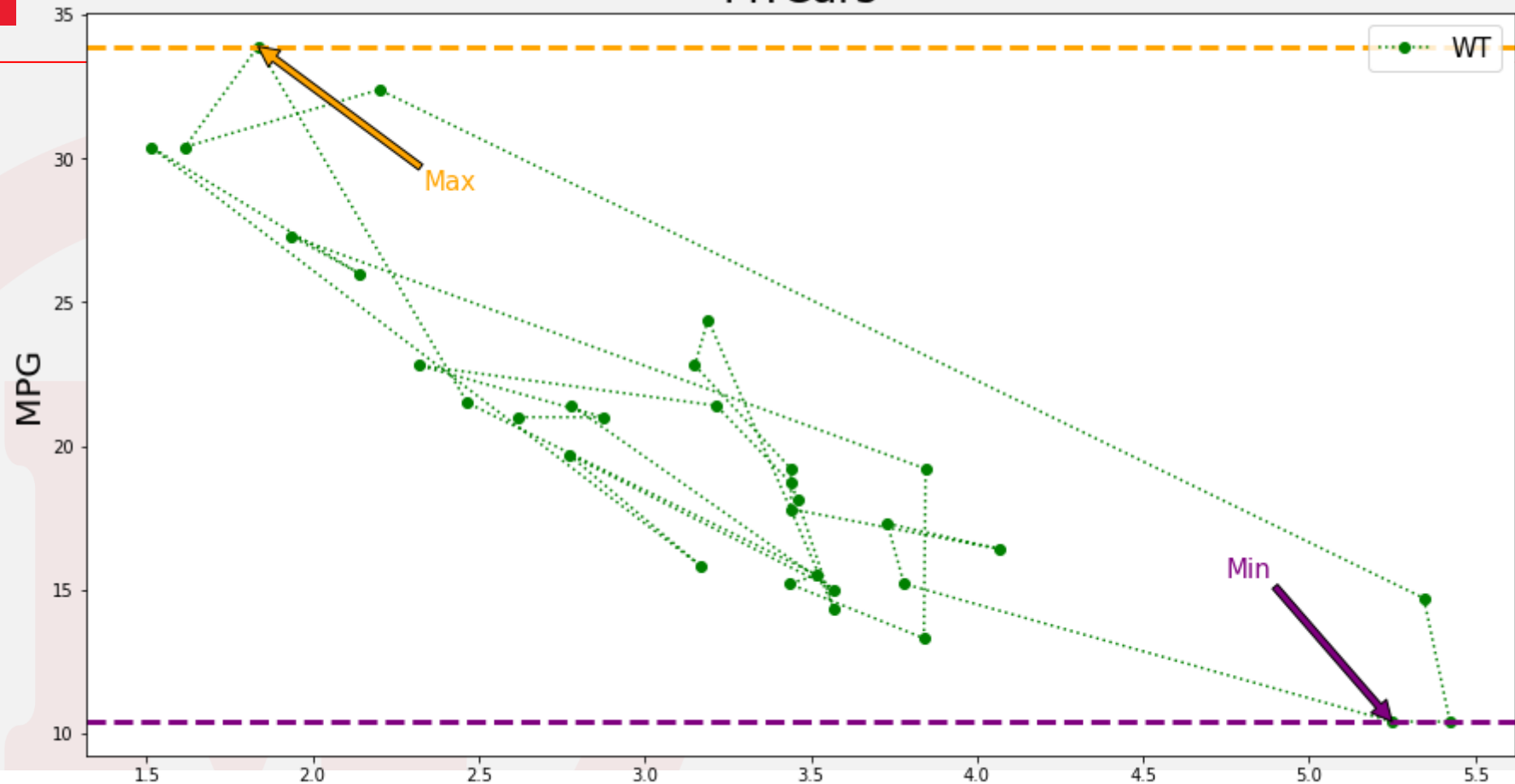
```
plt.plot(df['wt'], df['mpg'], color="green", ls = "dotted", marker = "o")
plt.title("MTCars", fontsize=25)
plt.xlabel("WT", fontsize=20)
plt.ylabel("MPG", fontsize=20)
plt.legend(["WT"], loc="upper right", fontsize=15)

plt.axhline(max(df['mpg']), color="orange", lw=3, ls="--")
plt.axhline(min(df['mpg']), color="purple", lw=3, ls='--')

x_max = df['wt'][df['mpg'].idxmax()]
y_max = max(df['mpg'])
plt.annotate("Max", xy=(x_max,y_max), color="orange", fontsize=15)

x_min = df['wt'][df['mpg'].idxmin()]
y_min = min(df['mpg'])
plt.annotate("Min", xy=(x_min,y_min), color="purple", fontsize=15)
```

MTCars



```
plt.plot(df['wt'], df['mpg'], color="green", ls = "dotted", marker = "o")
plt.title("MTCars", fontsize=25)
plt.xlabel("WT", fontsize=20)
plt.ylabel("MPG", fontsize=20)
plt.legend(["WT"], loc="upper right", fontsize=15)

plt.axhline(max(df['mpg']), color="orange", lw=3, ls="--")
plt.axhline(min(df['mpg']), color="purple", lw=3, ls='--')

x_max = df['wt'][df['mpg'].idxmax()]
y_max = max(df['mpg'])

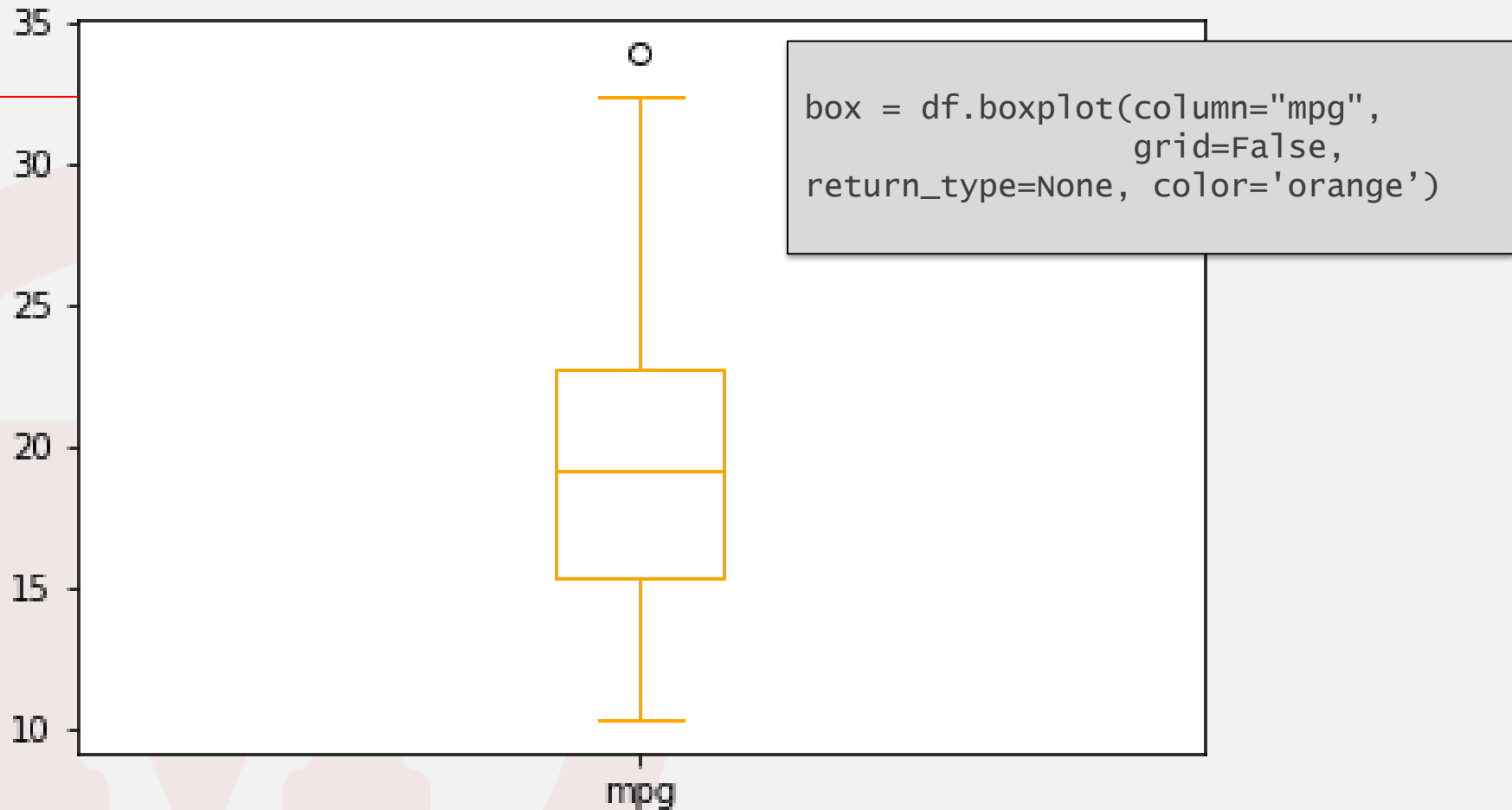
xt_max = df['wt'][df['mpg'].idxmax()] + 0.5
yt_max = max(df['mpg']) - 5

plt.annotate("Max", xy=(x_max,y_max), color="orange",
             fontsize=15,xytext=(xt_max,yt_max), arrowprops=dict(facecolor="orange") )

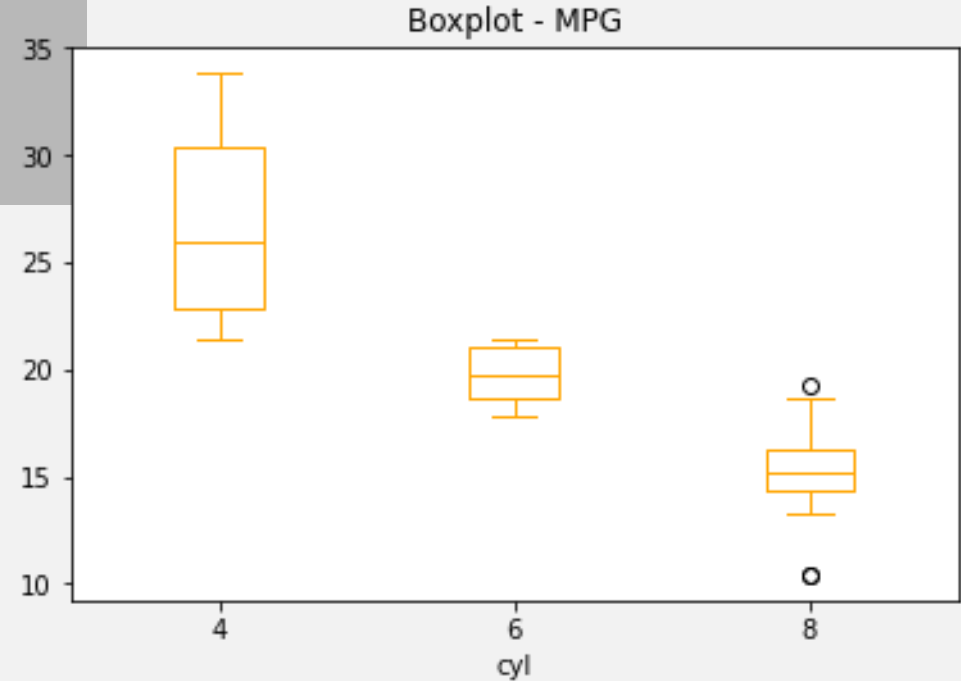
x_min = df['wt'][df['mpg'].idxmin()]
y_min = min(df['mpg'])

xt_min = df['wt'][df['mpg'].idxmin()] - 0.5
yt_min = min(df['mpg']) + 5

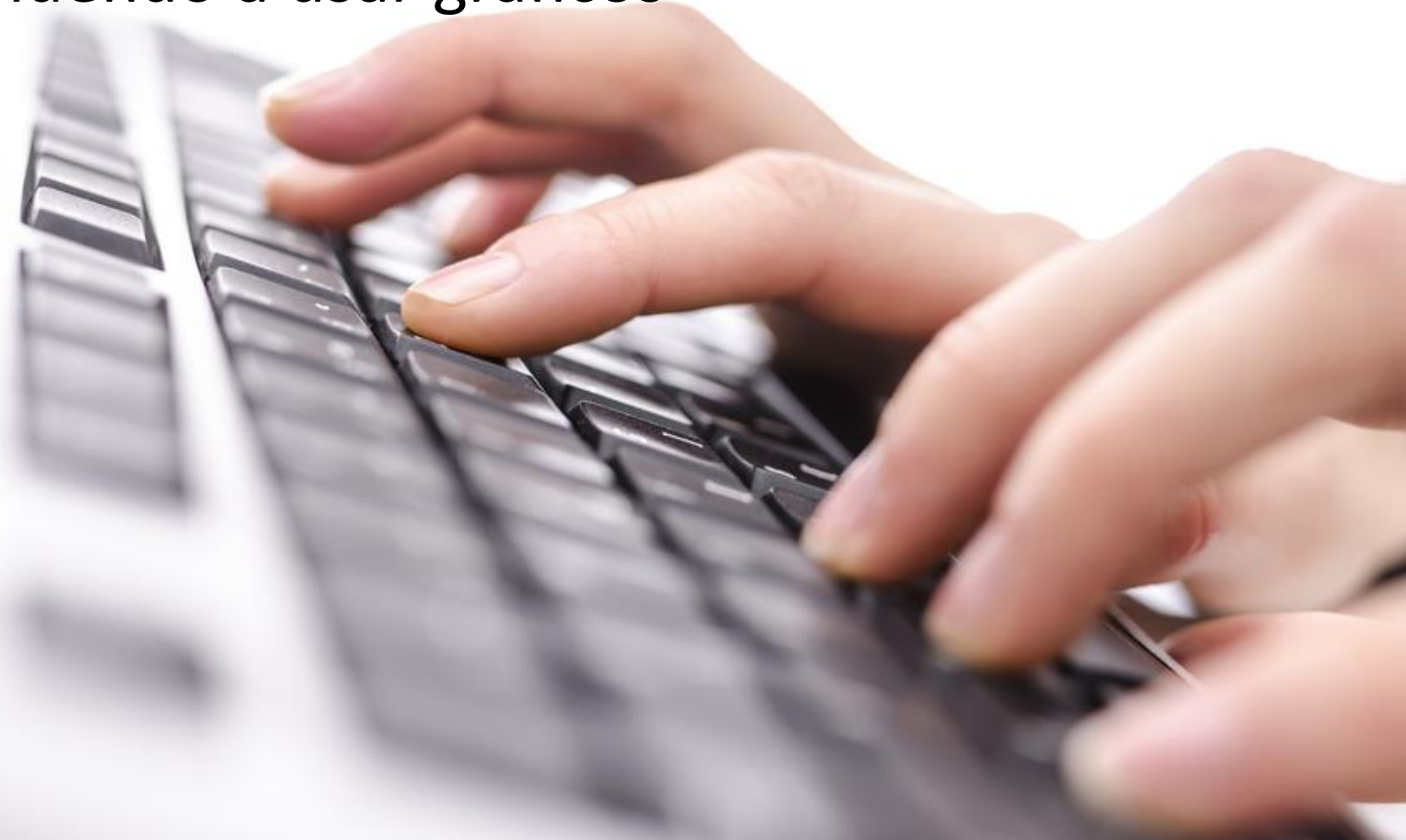
plt.annotate("Min", xy=(x_min,y_min), color="purple", fontsize=15,
             xytext=(xt_min,yt_min), arrowprops=dict(facecolor="purple"))
```



```
box = df.boxplot(column="mpg", by="cyl",  
                  grid=False,  
                  return_type=None, color='orange')  
plt.suptitle("")  
plt.title("Boxplot - MPG")
```



Aprendendo a usar graficos



Referências - Materiais

Estilo de linha: https://matplotlib.org/3.2.1/api/_as_gen/matplotlib.lines.Line2D.html

BoxPlot no Matplotlib: https://matplotlib.org/api/_as_gen/matplotlib.pyplot.boxplot.html

Cores no Matplotlib: https://matplotlib.org/gallery/color/named_colors.html

Marcador no Matplotlib: https://matplotlib.org/api/markers_api.html?highlight=marker

BoxPlot no Pandas:

<https://pandas.pydata.org/pandas-docs/version/0.23.4/generated/pandas.DataFrame.boxplot.html>





Faculdade de Computação e Informática
Mackenzie