

# Pyplot vs OO Matplotlib

which one is better?

- when plotting something quickly, use pyplot
- when plotting something more advanced, use matplotlib

```
In [1]: %matplotlib inline
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

In [2]: heart_disease = pd.read_csv("13.1 heart-disease.csv")
heart_disease.head()

Out[2]:
```

	age	sex	cp	trestbps	chol	fb	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

```
In [3]: over_50 = heart_disease[heart_disease["age"] > 50]
over_50.head()

Out[3]:
```

	age	sex	cp	trestbps	chol	fb	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
5	57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
6	56	0	1	140	294	0	0	153	0	1.3	1	0	2	1

```
In [4]: over_50.max()

Out[4]:
```

age	77.0
sex	1.0
cp	3.0
trestbps	200.0
chol	564.0
fb	1.0
restecg	2.0
thalach	195.0
exang	1.0
oldpeak	6.2
slope	2.0
ca	4.0
thal	3.0
target	1.0
dtype:	float64

## Pyplot

```
In [5]: over_50.plot(kind = "scatter",
x = "age",
y = "chol",
c = "target");
```

## Matplotlib OO (object oriented)

```
In [6]: fig, ax = plt.subplots(figsize = (10, 5))
over_50.plot(kind = "scatter",
x = "age",
y = "chol",
c = "target",
ax = ax);

# ax.set_xlim([45, 100])
```

## OO method from scratch

```
In [7]: # create plot
fig, ax = plt.subplots(figsize = (10,7))

# plot the data
scatter = ax.scatter(x = over_50.age,
y = over_50.chol,
c = over_50.target,
marker = '+')

# customize plot
ax.set(title = "Heart Disease",
xlabel = "Age",
ylabel = "Cholesterol level")

# add legend to identify each plot on the data
ax.legend("scatter.legend_elements()", title = "target");

# add horizontal line across the axis
ax.axhline(over_50.chol.mean(),
linestyle = '--');
```

```
In [8]: over_50.head()

Out[8]:
```

	age	sex	cp	trestbps	chol	fb	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
5	57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
6	56	0	1	140	294	0	0	153	0	1.3	1	0	2	1

## Subplots

- create subplot using age, chol and thalach

```
In [9]: # create a plot
fig, (ax0, ax1) = plt.subplots(nrows = 2,
ncols = 1,
figsize = (10,10),
sharex = True)

# axis 0
# plot the data
scatter = ax0.scatter(x = over_50.age,
y = over_50.chol,
c = over_50.target,
marker = 'o')

# customize the data labels
ax0.set(title = "Heart Disease and Cholesterol Level",
xlabel = "Age",
ylabel = "Cholesterol")

# add legends
ax0.legend("scatter.legend_elements()", title = "target")

# add a meanline
ax0.axhline(over_50.chol.mean(), linestyle = '--');

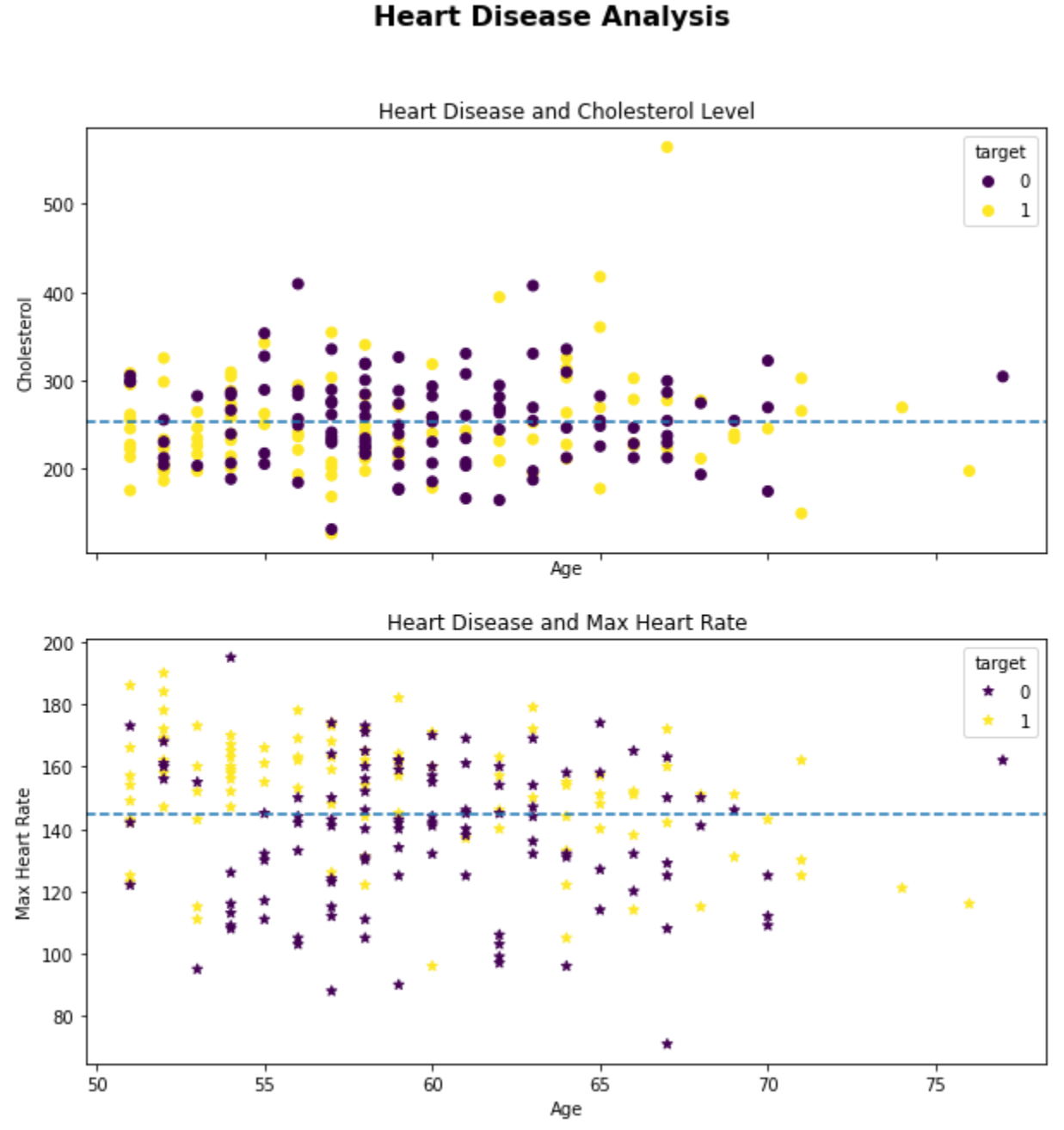
# axis1
# plot the data
scatter1 = ax1.scatter(x = over_50.age,
y = over_50.thalach,
c = over_50.target,
marker = '+')

# customize the labels
ax1.set(title = "Heart Disease and Max Heart Rate",
xlabel = "Age",
ylabel = "Max Heart Rate")

# add legend
ax1.legend("scatter1.legend_elements()", title = "target")

# add a meanline
ax1.axhline(over_50.thalach.mean(), linestyle = '--');

# add a title to the figure
fig.suptitle("Heart Disease Analysis", fontsize = 16, fontweight = 'bold');
```



```
In [10]: # change plot styles
plt.style.use('grayscale')
```

```
In [11]: # create a plot
fig, (ax0, ax1) = plt.subplots(nrows = 2,
ncols = 1,
figsize = (10,10),
sharex = True)

# axis 0
# plot the data
scatter = ax0.scatter(x = over_50.age,
y = over_50.chol,
c = over_50.target,
cmap = 'winter',
marker = '+')

# customize the data labels
ax0.set(title = "Heart Disease and Cholesterol Level",
ylabel = "Cholesterol")

# set data limit
ax0.set_xlim([50,80])
ax0.set_ylim([100, 600])

# add legends
ax0.legend("scatter.legend_elements()", title = "target")

# add a meanline
ax0.axhline(over_50.chol.mean(), linestyle = '--');

# axis1
# plot the data
scatter1 = ax1.scatter(x = over_50.age,
y = over_50.thalach,
c = over_50.target,
cmap = 'plasma',
marker = '^')

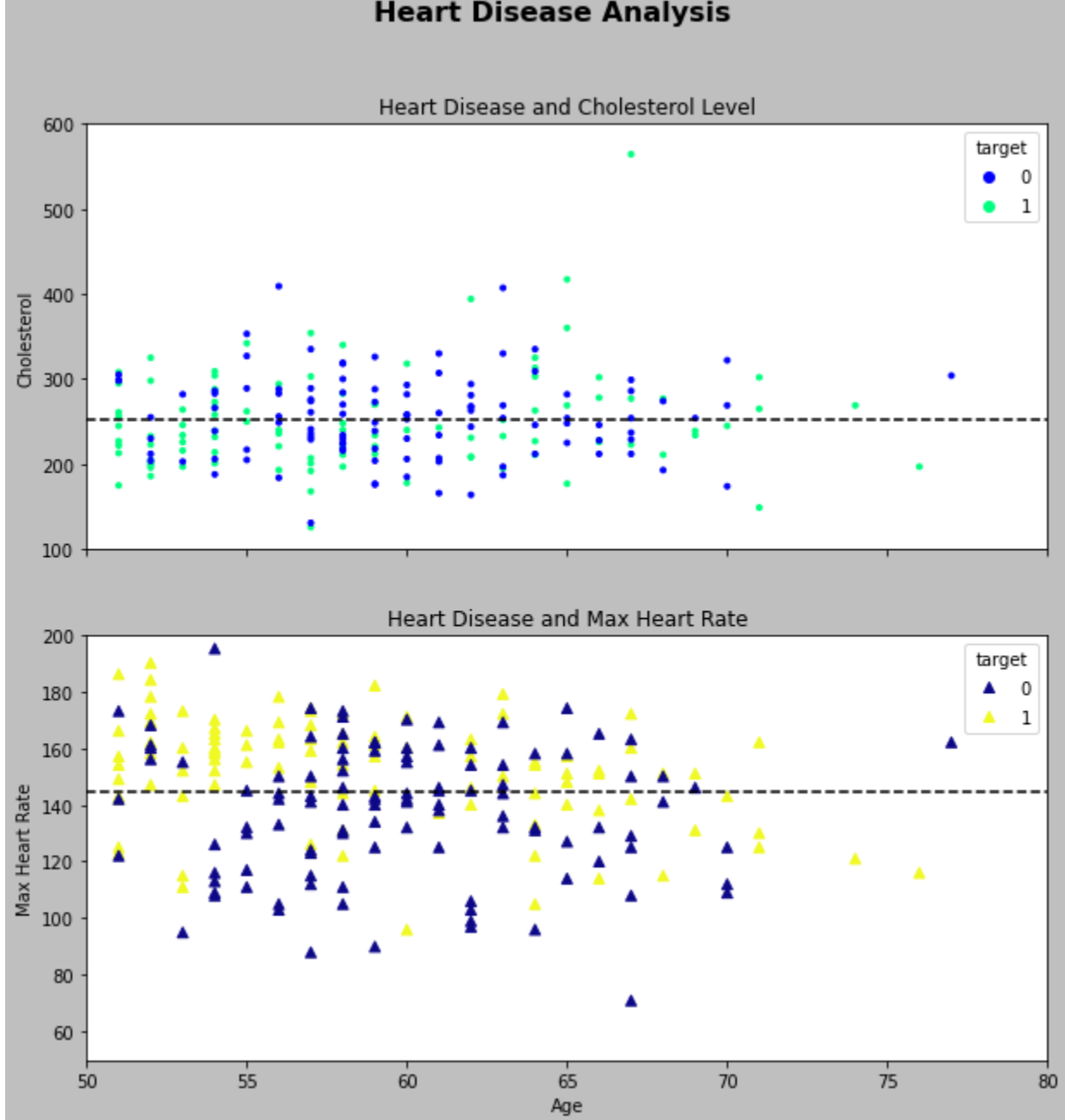
# customize the labels
ax1.set(title = "Heart Disease and Max Heart Rate",
xlabel = "Age",
ylabel = "Max Heart Rate")

# set data limit
ax1.set_xlim([50, 80])
ax1.set_ylim([50, 200])

# add legend
ax1.legend("scatter1.legend_elements()", title = "target")

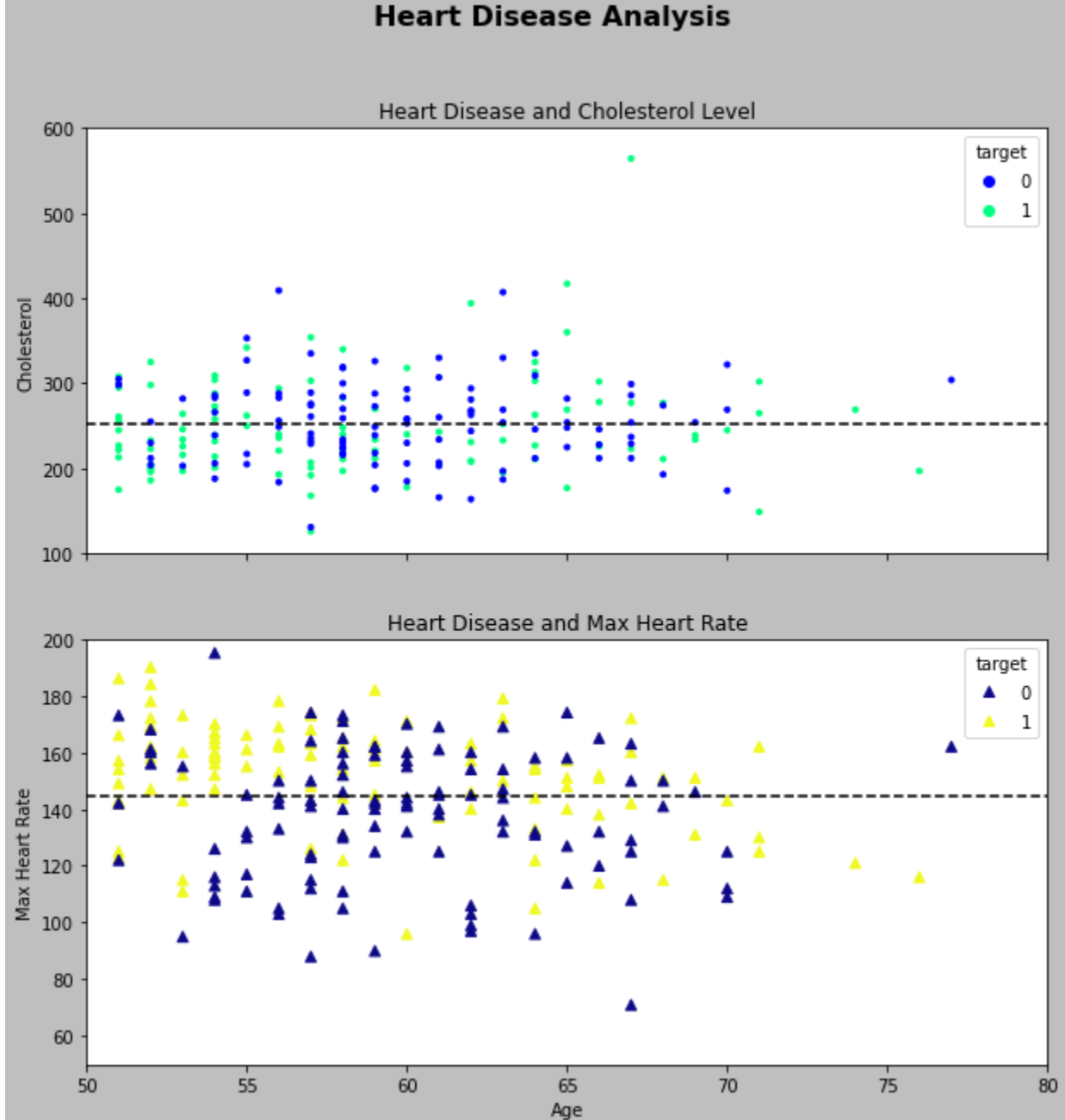
# add a meanline
ax1.axhline(over_50.thalach.mean(), linestyle = '--');

# add a title to the figure
fig.suptitle("Heart Disease Analysis", fontsize = 16, fontweight = 'bold');
```



- call out the picture(fig)

```
In [12]: fig
```



```
In [ ]:
```