

## ✓ Continuous Variables - Boxplot - Exercise

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

```
from google.colab import drive
import os
```

```
drive.mount('/content/drive')
os.chdir('/content/drive/MyDrive/')
for item in os.listdir():
    print(item)
print("-----")
os.chdir('/content/drive/MyDrive/cloud/GitHub/AdvDataViz/Notebooks/')
for item in os.listdir():
    print(item)
print("-----")
notebooks = "/content/drive/MyDrive/cloud/GitHub/AdvDataViz/Notebooks"
print(os.listdir(notebooks))
print("-----")
```

```
file = "student_performance.csv"
file_path = os.path.join(notebooks, file)
with open(file_path, "r") as f:
    contents = f.read()
```

Mounted at /content/drive

learningStore

healthyCar

startup

cloud

Artificial Intelligence

-----

03 Matplotlib - Exercise.ipynb

02 Matplotlib.ipynb

01 Python\_Pandas.ipynb

04 Continuous Variables - Histogram .ipynb

05 Continuous Variables - Histogram - Exercise .ipynb

07 Continuous Variables - Boxplot - Exercise .ipynb

03 Matplotlib - Exercise Solutions.ipynb

05 Continuous Variables - Histogram - Exercise Solutions.ipynb

06 Continuous Variables - Boxplot.ipynb

08 Continuous Variables - Scatterplot.ipynb

07 Continuous Variables - Boxplot - Exercise Solutions.ipynb

09 Continuous Variables - Scatterplot - Exercise Solutions.ipynb

09 Continuous Variables - Scatterplot - Exercise .ipynb

10 Categorical Variables - Bar\_Pie.ipynb

12 Seaborn.ipynb

11 Pandas Data Visualization.ipynb

13 Seaborn - Exercise .ipynb

Top 50 US Tech Companies.csv

13 Seaborn - Exercise Solution.ipynb

15 Custom Modules.ipynb

14 Functions.ipynb

churn.csv

student\_performance.csv

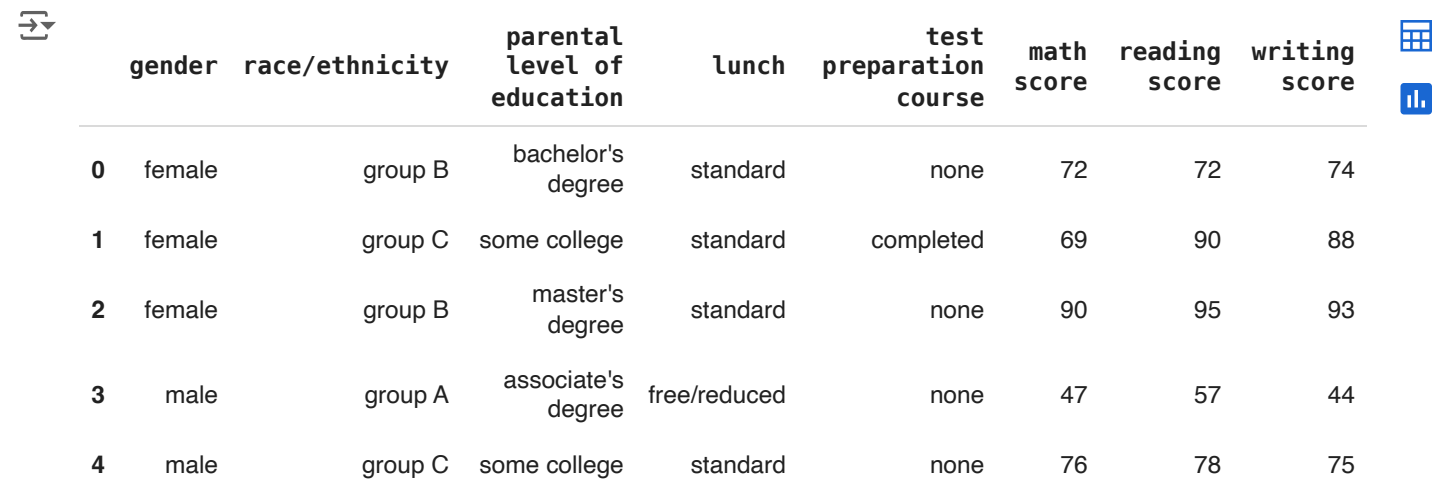
```
myplotlib.py
employee_attrition_.csv
heart-disease.csv
```

```
['03 Matplotlib - Exercise.ipynb', '02 Matplotlib.ipynb', '01 Python_Pandas.ipynb', '04 Cont
```

## Dataset: Student Performance

```
#df = pd.read_csv("student_performance.csv")
df = pd.read_csv(file_path)
```

```
df.head()
```



	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

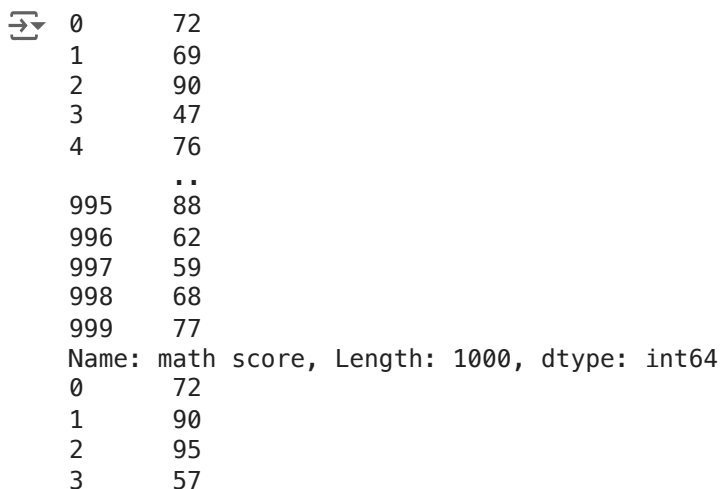
Next steps:

[Generate code with df](#)
[View recommended plots](#)
[New interactive sheet](#)

## Use math and reading scores for exercises

```
math_scores = df["math score"]
reading_scores = df["reading score"]
```

```
print(math_scores)
print(reading_scores)
```



```
0      72
1      69
2      90
3      47
4      76
..
995    88
996    62
997    59
998    68
999    77
Name: math score, Length: 1000, dtype: int64
0      72
1      90
2      95
3      57
```

```

4      78
      ..
995    99
996    55
997    71
998    78
999    86
Name: reading score, Length: 1000, dtype: int64

```

## ✓ Use simulated machine learning scores for exercises

```

# Simulated machine Learning cross-validation performance scores
# Randomly generated values between 0 and 1
LR = np.random.rand(10) # Logistic Regression
SVM = np.random.rand(10) # Support Vector Machine
RF = np.random.rand(10) # Random Forest
KNN = np.random.rand(10) # K-Nearest Neighbor

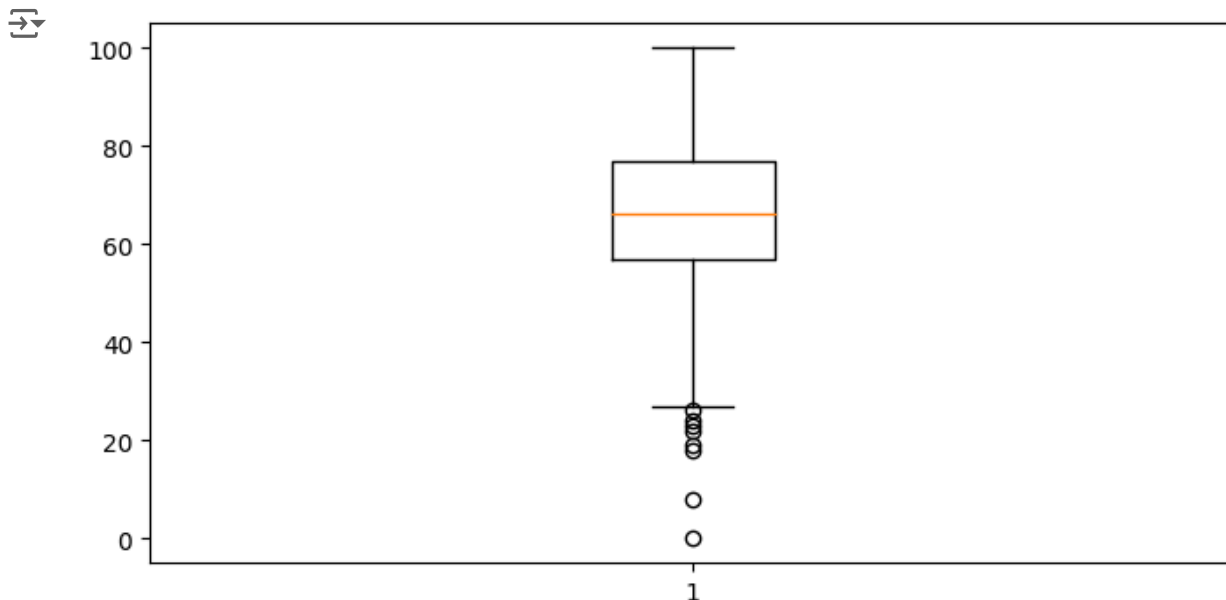
```

## ✓ 1.) Create a box plot of "math\_score". Set the figsize to (8,4).

```

fig, ax = plt.subplots(figsize=(8,4))
ax.boxplot(math_scores);

```



- 2.) Create two (paired) box plots in the same axis. Set the figsize to (15, 5). Create one box plot for "math\_score" and the other one for "reading\_score". Set the x label to "Tests" and the ylabel to "Score".

 **Generate**

create a dataframe with 2 columns and 10 rows

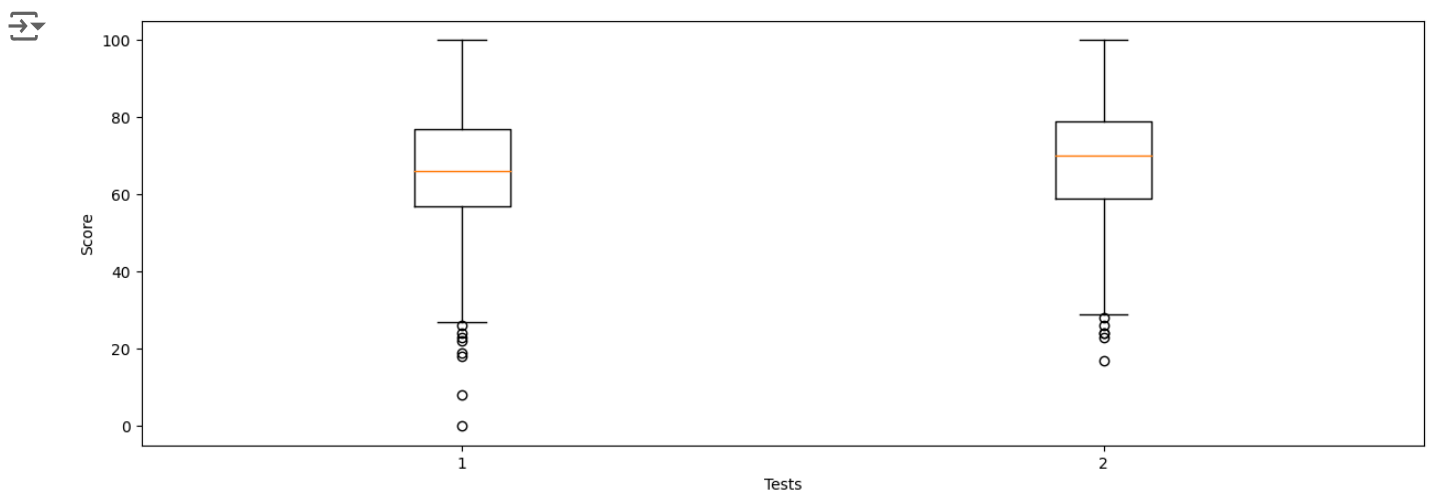


Close

```
fig, ax = plt.subplots(figsize=(15,5))

ax.boxplot([math_scores, reading_scores])

ax.set_xlabel('Tests')
ax.set_ylabel('Score');
```



- 3.) Create box plots within the same axis for each of the four machine learning models (i.e., "LR", "SVM", "RF", "KNN"). Set the figsize to (15, 5). Set the x label to "Classification Models", the ylabel to "Accuracy", and the title to "Machine Learning". Set the xtick labels to each of the models' respective names. Set the fontsize to 16 for all of the labels.

 **Generate**

print hello world using rot13

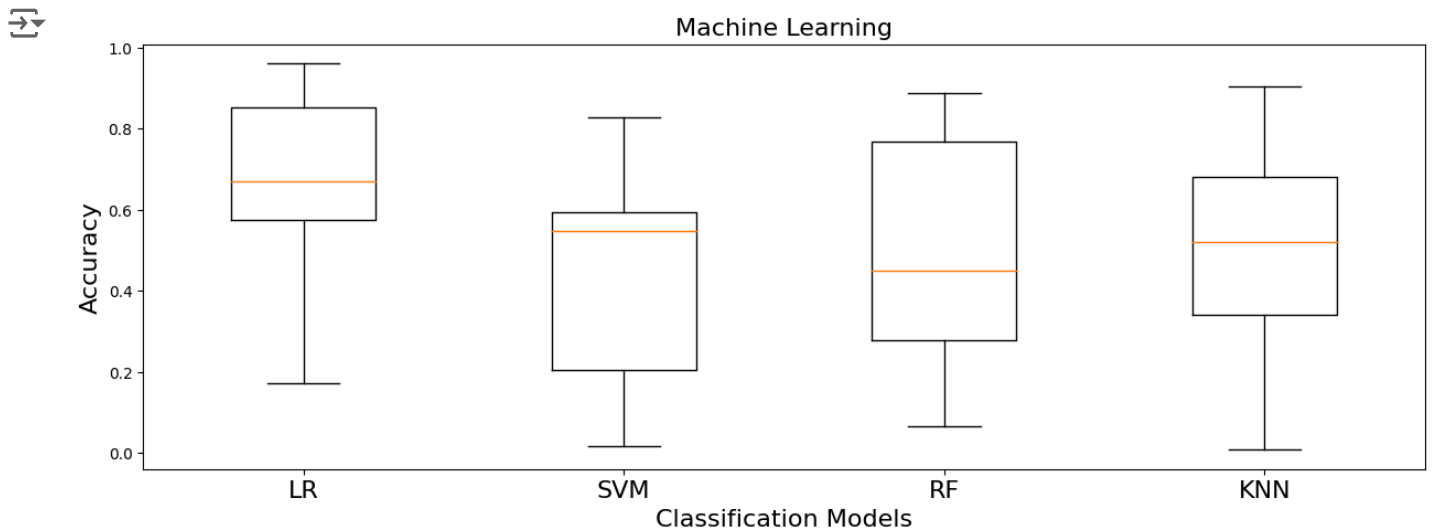


Close

```
fig, ax = plt.subplots(figsize = (15, 5))

bplot = ax.boxplot([LR, SVM, RF, KNN])

ax.set_xlabel('Classification Models', fontsize=16)
ax.set_ylabel('Accuracy', fontsize=16)
ax.set_title('Machine Learning', fontsize=16)
ax.set_xticklabels(["LR", "SVM", "RF", "KNN"], fontsize=16);
```



- 4) Refine the plot above by filling the boxes with color. Use the colors "cornflowerblue", "crimson", "gold", and "mediumseagreen". Also, set the median color to white with a linewidth of 4.

```
fig, ax = plt.subplots(figsize = (15, 5))

bplot = ax.boxplot([LR, SVM, RF, KNN], patch_artist=True, medianprops={"color":"white", "linewidth

ax.set_xlabel('Classification Models', fontsize=16)
ax.set_ylabel('Accuracy', fontsize=16)
ax.set_title('Machine Learning', fontsize=16)
ax.set_xticklabels(["LR", "SVM", "RF", "KNN"], fontsize=16)

# colors to use
colors = ["cornflowerblue", "crimson", "gold", "mediumseagreen"]

for patch, color in zip(bplot['boxes'], colors):
    patch.set_facecolor(color);
```

