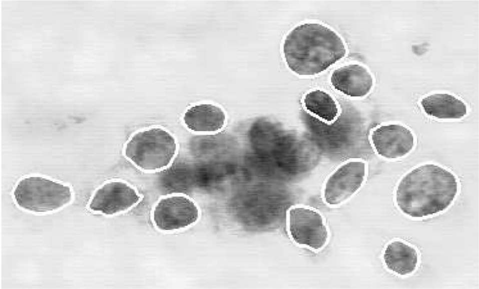
This notebook is an exercise in the Data Visualization course. You can reference the tutorial at this link.

In this exercise, you will use your new knowledge to propose a solution to a real-world scenario. To succeed, you will need to import data into Python, answer questions using the data, and generate **histograms** and **density plots** to understand patterns in the data.

Scenario

You'll work with a real-world dataset containing information collected from microscopic images of breast cancer tumors, similar to the image below.



Each tumor has been labeled as either **benign** (*noncancerous*) or **malignant** (*cancerous*).

To learn more about how this kind of data is used to create intelligent algorithms to classify tumors in medical settings, **watch the short video at this link**!

Setup

Run the next cell to import and configure the Python libraries that you need to complete the exercise.

In [ ]:

```
import pandas as pd
pd.plotting.register_matplotlib_converters()
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
print("Setup Complete")
```

The questions below will give you feedback on your work. Run the following cell to set up our feedback system.

In [ ]:

```
# Set up code checking
import os
if not os.path.exists("../input/cancer_b.csv"):
    os.symlink("../input/data-for-datavis/cancer_b.csv", "../input/cancer_b.csv")
    os.symlink("../input/data-for-datavis/cancer_m.csv", "../input/cancer_m.csv")
from learntools.core import binder
binder.bind(globals())
from learntools.data_viz_to_coder.ex5 import *
print("Setup Complete")
```

Step 1: Load the data

In this step, you will load two data files.

- Load the data file corresponding to **benign** tumors into a DataFrame called `cancer_b_data`. The corresponding filepath is `cancer_b_filepath`. Use the `"Id"` column to label the rows.
- Load the data file corresponding to **malignant** tumors into a DataFrame called `cancer_m_data`. The corresponding filepath is `cancer_m_filepath`. Use the `"Id"` column to label the rows.

In [ ]:

```
# Paths of the files to read
# Paths of the files to read
cancer_b_filepath = "../input/cancer_b.csv"
cancer_m_filepath = "../input/cancer_m.csv"

# Fill in the line below to read the (benign) file into a variable cancer_b_data
cancer_b_data = pd.read_csv(cancer_b_filepath,index_col='Id')

# Fill in the line below to read the (malignant) file into a variable cancer_m_data
cancer_m_data = pd.read_csv(cancer_m_filepath,index_col='Id')

# Run the line below with no changes to check that you've loaded the data correctly
step_1.check()
```

In [ ]:

```
# Lines below will give you a hint or solution code
#step_1.hint()
#step_1.solution()
```

Step 2: Review the data

Use a Python command to print the first 5 rows of the data for benign tumors.

In [ ]:

```
# Print the first five rows of the (benign) data
____ # Your code here
cancer_b_data.head()
```

Use a Python command to print the first 5 rows of the data for malignant tumors.

In [ ]:

```
# Print the first five rows of the (malignant) data
____ # Your code here
cancer_m_data.head()
```

In the datasets, each row corresponds to a different image. Each dataset has 31 different columns, corresponding to:

- 1 column (`'Diagnosis'`) that classifies tumors as either benign (which appears in the dataset as **`B`**) or malignant (**`M`**), and
- 30 columns containing different measurements collected from the images.

Use the first 5 rows of the data (for benign and malignant tumors) to answer the questions below.

```
In [ ]: # Fill in the line below: In the first five rows of the data for benign tumors, what is the

        # largest value for 'Perimeter (mean)'?
        max_perim = 87.46

        # Fill in the line below: What is the value for 'Radius (mean)' for the tumor with Id 842517?
        mean_radius = 20.57

        # Check your answers
        step_2.check()
```

```
In [ ]: # Lines below will give you a hint or solution code
        #step_2.hint()
        #step_2.solution()
```

Step 3: Investigating differences

Part A

Use the code cell below to create two histograms that show the distribution in values for `'Area (mean)'` for both benign and malignant tumors. (*To permit easy comparison, create a single figure containing both histograms in the code cell below.*)

In [ ]:

```
# Histograms for benign and maligant tumors
sns.distplot(a=cancer_b_data['Area (mean)'], label="Benign", kde=False)
sns.distplot(a=cancer_m_data['Area (mean)'], label="Malignant", kde=False)
plt.legend()
# Check your answer
step_3.a.check()
```

In [ ]:

```
# Lines below will give you a hint or solution code
#step_3.a.hint()
#step_3.a.solution_plot()
```

Part B

A researcher approaches you for help with identifying how the `'Area (mean)'` column can be used to understand the difference between benign and malignant tumors. Based on the histograms above,

- Do malignant tumors have higher or lower values for `'Area (mean)'` (relative to benign tumors), on average?
- Which tumor type seems to have a larger range of potential values?

In [ ]:

```
#step_3.b.hint()
```

In [ ]:

```
# Check your answer (Run this code cell to receive credit!)
step_3.b.solution()
```

Step 4: A very useful column

Part A

Use the code cell below to create two KDE plots that show the distribution in values for `'Radius (worst)'` for both benign and malignant tumors. (*To permit easy comparison,* *create a single figure containing both KDE plots in the code cell below.*)

In [ ]:

```
# KDE plots for benign and malignant tumors
____ # Your code here (benign tumors)
____ # Your code here (malignant tumors)
sns.kdeplot(data=cancer_b_data['Radius (worst)'], shade=True, label="Benign")
sns.kdeplot(data=cancer_m_data['Radius (worst)'], shade=True, label="Malignant")
plt.legend()
# Check your answer
step_4.a.check()
```

In [ ]:

```
# Lines below will give you a hint or solution code
#step_4.a.hint()
#step_4.a.solution_plot()
```

Part B

A hospital has recently started using an algorithm that can diagnose tumors with high accuracy. Given a tumor with a value for `'Radius (worst)'` of 25, do you think the algorithm is more likely to classify the tumor as benign or malignant?

In [ ]:

```
#step_4.b.hint()
```

In [ ]:

```
# Check your answer (Run this code cell to receive credit!)
step_4.b.solution()
```

Keep going

Review all that you've learned and explore how to further customize your plots in the **next tutorial**!

Have questions or comments? Visit the Learn Discussion forum to chat with other Learners.