SAIT

Advanced Concepts in Data Analytics

Lab: Machine Learning

This page was intentionally left blank.

Table of Contents

Machine Learning

Introduction

In the lab, you'll practice machine learning techniques to solve real-world problems. Using three different datasets, you'll perform classification, regression and clustering, respectively. You'll continue using pandas for data manipulation and seaborn for data visualization, and you'll also use scikit-learn, one of most widely used libraries in machine learning.

Equipment and Materials

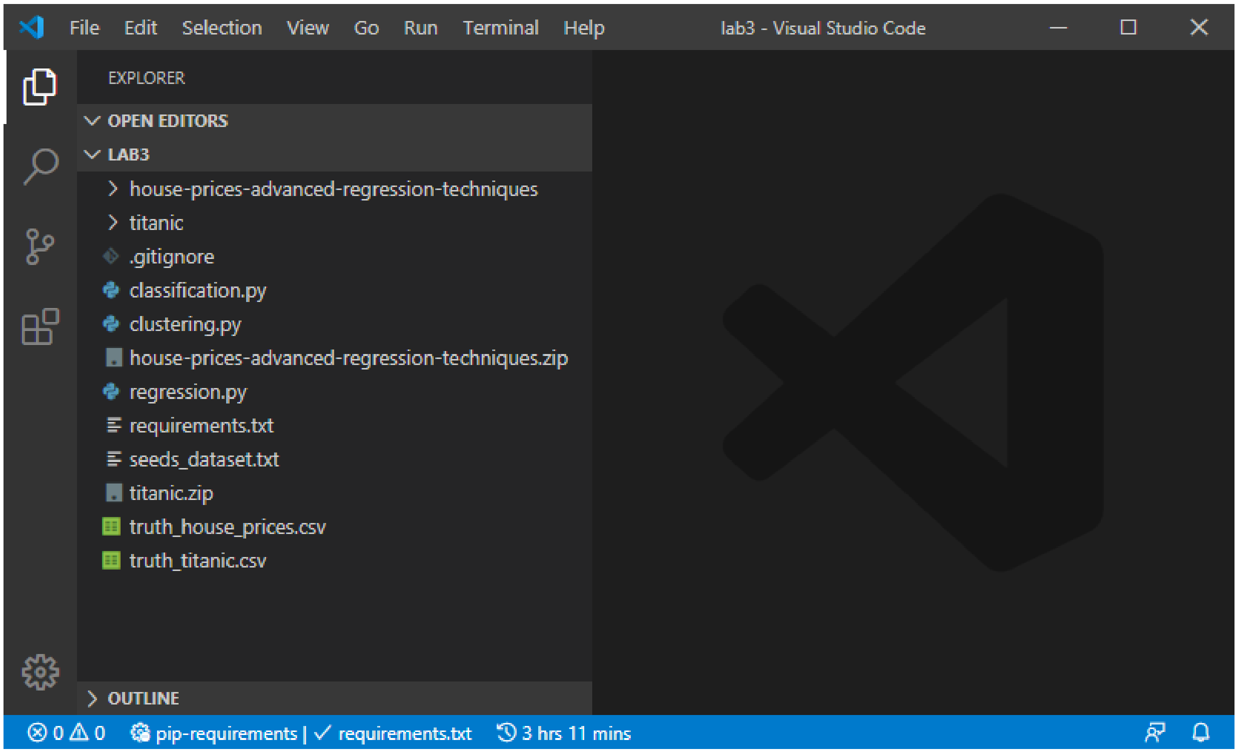
- BYOD laptop
- Python
- Visual Studio Code
- TXT file: requirements.txt
- Git file: gitignore
- Lab Activity 1 data and skeleton files:
    - titanic.zip
    - truth_titanic.csv
    - classficatioin.py
- Lab Activity 2 data and skeleton files:
    - house-prices-advanced-regression-techniques.zip
    - truth_house_prices.csv
    - regression.py
- Lab Activity 3 data and skeleton files:
    - Seeds_dataset.txt
    - clustering.py

Instructions

Lab Setup

1. Download all the data and skeleton files listed in the Equipment and Materials section above and create a folder for them on your computer (e.g., **lab3**).
2. Unzip each .zip file into your lab3 folder.
3. Open the folder in VS Code.

   The initial structure should resemble the image below.



LAB3
- house-prices-advanced-regression-techniques
- titanic
- .gitignore
- classification.py
- clustering.py
- house-prices-advanced-regression-techniques.zip
- regression.py
- requirements.txt
- seeds_dataset.txt
- titanic.zip
- truth_house_prices.csv
- truth_titanic.csv

Figure 1: Opening the Lab Folder in VS Code
Used with permission from Microsoft.

4. Create a virtual environment named **venv**, configure VS Code to use the newly created virtual environment as the default Python interpreter, and install the required dependencies. See Lab 2, Activity 1: Steps 3 to 6 for reference.

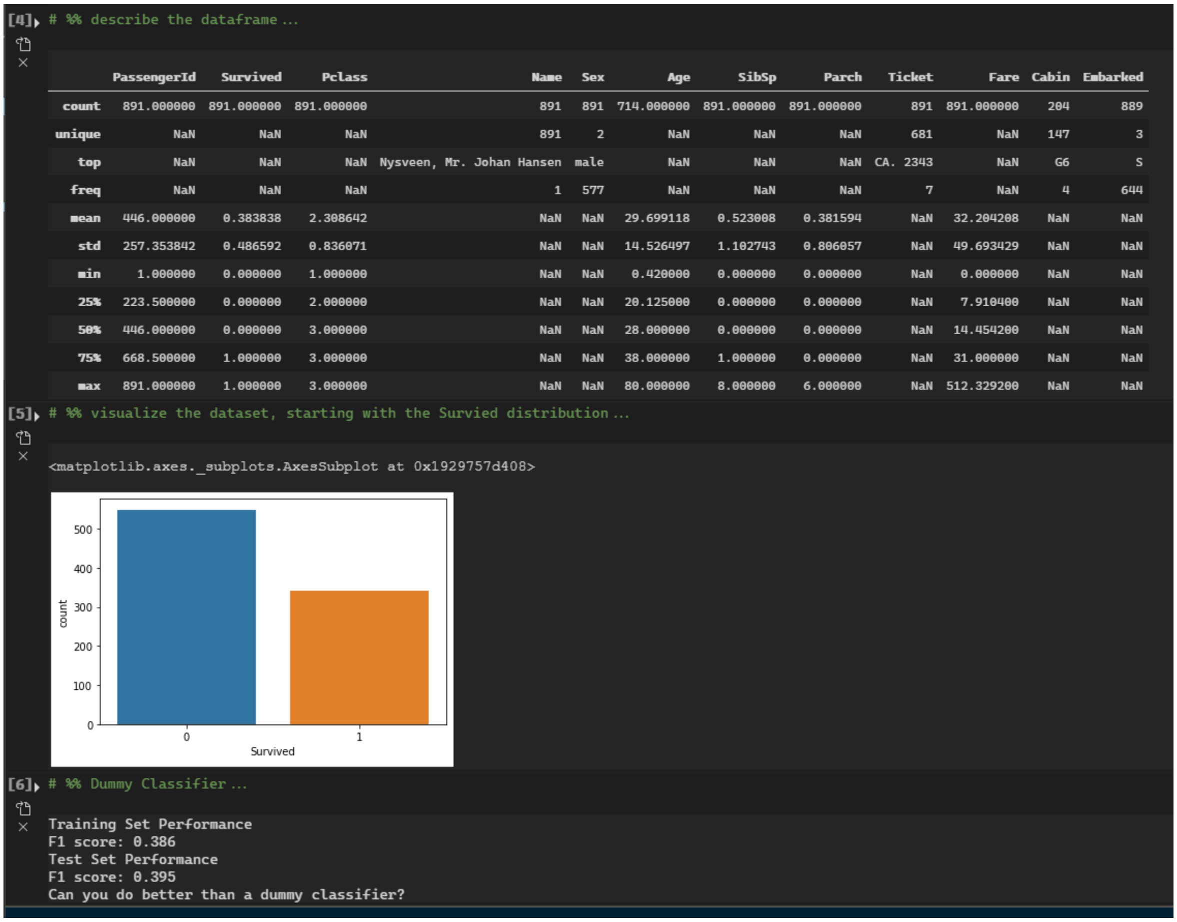
Lab Activity 1: Classifying Titanic Survival

In this section, you'll investigate one of the most infamous disasters: the sinking of the Titanic. Although over 1,500 lives were lost, some people survived. You'll use a dataset containing a list of passengers and their characteristics (survival, sex, age, ticket class, cabin number, port of embarkation, etc.) and build a predictive model to classify whether a passenger would have survived based on their passenger data.

Note: Although you have already downloaded this dataset, it is also publicly available from Titanic: Machine Learning from Disaster (https://www.kaggle.com/c/titanic).

1. Use the skeleton file **classification.py** and run through the nonempty code blocks.

   Your result should resemble the screen below.



```
[4] # %% describe the dataframe ...
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
count	891.000000	891.000000	891.000000	891	891	714.000000	891.000000	891.000000	891	891.000000	204	889
unique	NaN	NaN	NaN	891	2	NaN	NaN	NaN	681	NaN	147	3
top	NaN	NaN	NaN	Nysveen, Mr. Johan Hansen	male	NaN	NaN	NaN	CA. 2343	NaN	G6	S
freq	NaN	NaN	NaN	1	577	NaN	NaN	NaN	7	NaN	4	644
mean	446.000000	0.383838	2.308642	NaN	NaN	29.699118	0.523008	0.381594	NaN	32.204208	NaN	NaN
std	257.353842	0.486592	0.836071	NaN	NaN	14.526497	1.102743	0.806057	NaN	49.693429	NaN	NaN
min	1.000000	0.000000	1.000000	NaN	NaN	0.420000	0.000000	0.000000	NaN	0.000000	NaN	NaN
25%	223.500000	0.000000	2.000000	NaN	NaN	20.125000	0.000000	0.000000	NaN	7.910400	NaN	NaN
50%	446.000000	0.000000	3.000000	NaN	NaN	28.000000	0.000000	0.000000	NaN	14.454200	NaN	NaN
75%	668.500000	1.000000	3.000000	NaN	NaN	38.000000	1.000000	0.000000	NaN	31.000000	NaN	NaN
max	891.000000	1.000000	3.000000	NaN	NaN	80.000000	8.000000	6.000000	NaN	512.329200	NaN	NaN

```
[5] # %% visualize the dataset, starting with the Survied distribution ...

<matplotlib.axes._subplots.AxesSubplot at 0x1929757d408>
```

```
[6] # %% Dummy Classifier ...

Training Set Performance
F1 score: 0.386
Test Set Performance
F1 score: 0.395
Can you do better than a dummy classifier?
```

Figure 2: Executing the Skeleton File

Used with permission from Microsoft.

This reads the data files into dataframes, shows some simple statistics, and visualizes the survival distribution. It also provides an evaluation function and a dummy classifier.

However, the dummy classifier only achieves a score of approximately 40% F1 on the test data.

2. Use your data mining skills to explore and understand the dataset. Some sample questions are provided in the skeleton file.

   For example, if you choose to answer **Survived w.r.t Age distribution**, your result should resemble the screen below.

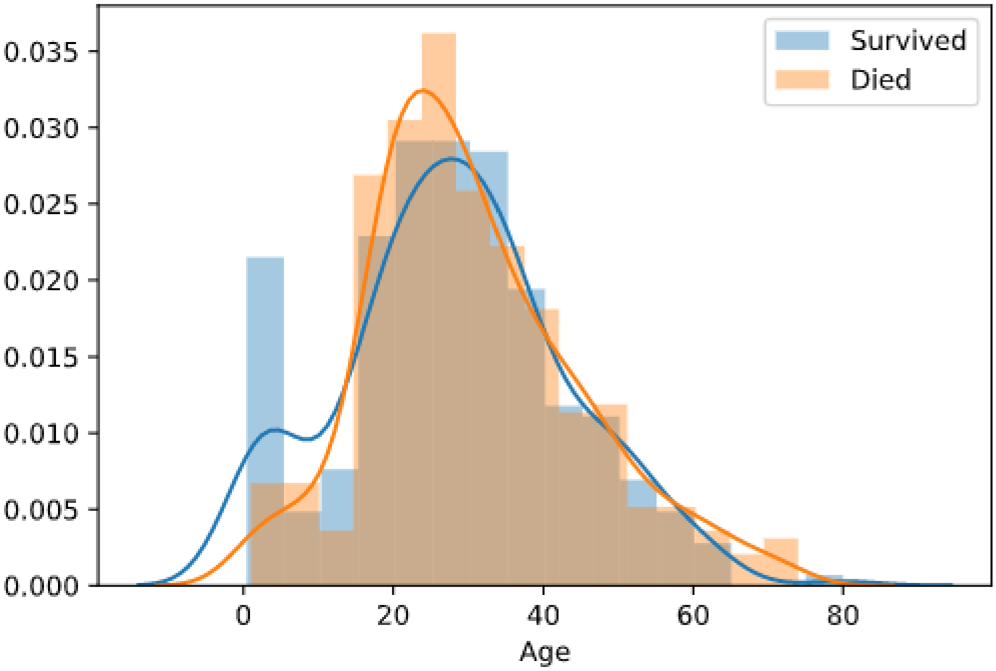


Figure 3: Age Distribution Survival Rate
© 2020, Southern Alberta Institute of Technology

3. In the last code block, provide your solution to the classification problem.

   Your goal is to provide a better solution than the dummy classifier. In other words, your F1 score on the test set should be higher than 0.395.

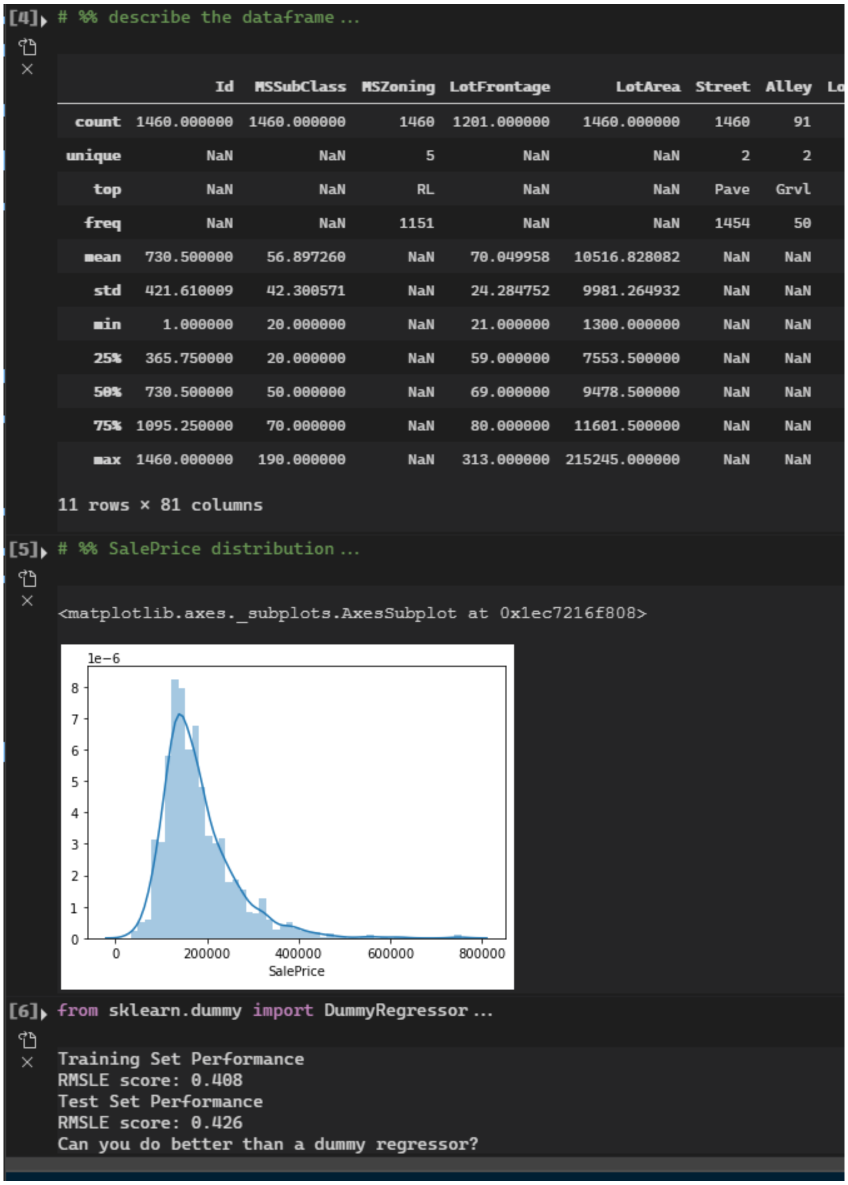
Lab Activity 2: Predicting House Prices

In this section, you will work with the Ames Housing dataset. You'll use the characteristics of a house and build a predictive model to forecast its sale price.

Note: Although you have already downloaded this dataset, it is also publicly available from House Prices: Advanced Regression Techniques (https://www.kaggle.com/c/house-prices-advanced-regression-techniques).

1. Use the skeleton file **regression.py** and run through the nonempty code blocks.

   Your result should resemble the screen below.



```
[4] # %% describe the dataframe ...
```

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	Lo
count	1460.000000	1460.000000	1460	1201.000000	1460.000000	1460	91	
unique	NaN	NaN	5	NaN	NaN	2	2	
top	NaN	NaN	RL	NaN	NaN	Pave	Grvl	
freq	NaN	NaN	1151	NaN	NaN	1454	50	
mean	730.500000	56.897260	NaN	70.049958	10516.828082	NaN	NaN	
std	421.610009	42.300571	NaN	24.284752	9981.264932	NaN	NaN	
min	1.000000	20.000000	NaN	21.000000	1300.000000	NaN	NaN	
25%	365.750000	20.000000	NaN	59.000000	7553.500000	NaN	NaN	
50%	730.500000	50.000000	NaN	69.000000	9478.500000	NaN	NaN	
75%	1095.250000	70.000000	NaN	80.000000	11601.500000	NaN	NaN	
max	1460.000000	190.000000	NaN	313.000000	215245.000000	NaN	NaN	

11 rows × 81 columns

```
[5] # %% SalePrice distribution ...

<matplotlib.axes._subplots.AxesSubplot at 0x1ec7216f808>
```

```
[6] from sklearn.dummy import DummyRegressor ...

Training Set Performance
RMSLE score: 0.408
Test Set Performance
RMSLE score: 0.426
Can you do better than a dummy regressor?
```

Figure 4: Executing the Skeleton File

Used with permission from Microsoft.

2. Use your data mining skills to explore and understand the dataset. Some sample questions are provided in the skeleton file.

   For example, if you choose to answer **SalePrice distribution w.r.t YearBuilt**, your result should resemble the screen below.

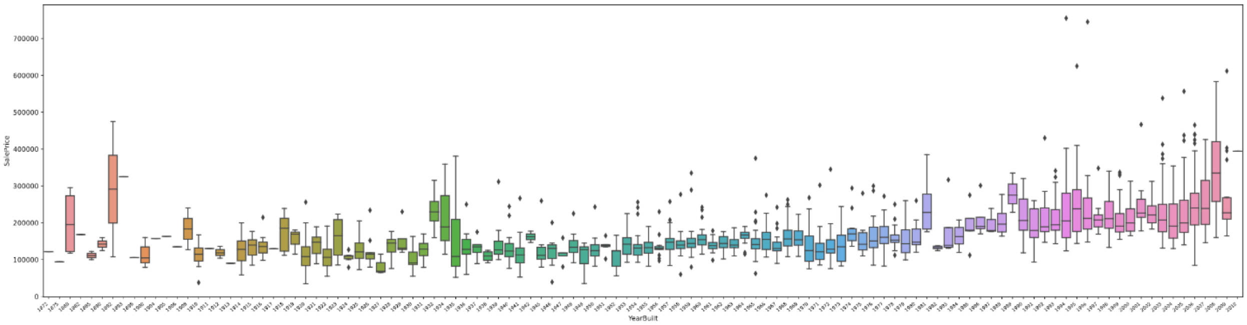


Figure 5: SalePrice distribution w.r.t YearBuilt

© 2020, Southern Alberta Institute of Technology

3. Plot as many figures as possible to visually comprehend the relationship between variables and sale prices.

4. In the last code block, provide your solution to the regression problem. You goal is to provide a better solution than the dummy regressor. In other words, your RMSLE score on the test set should be lower than **0.426**.

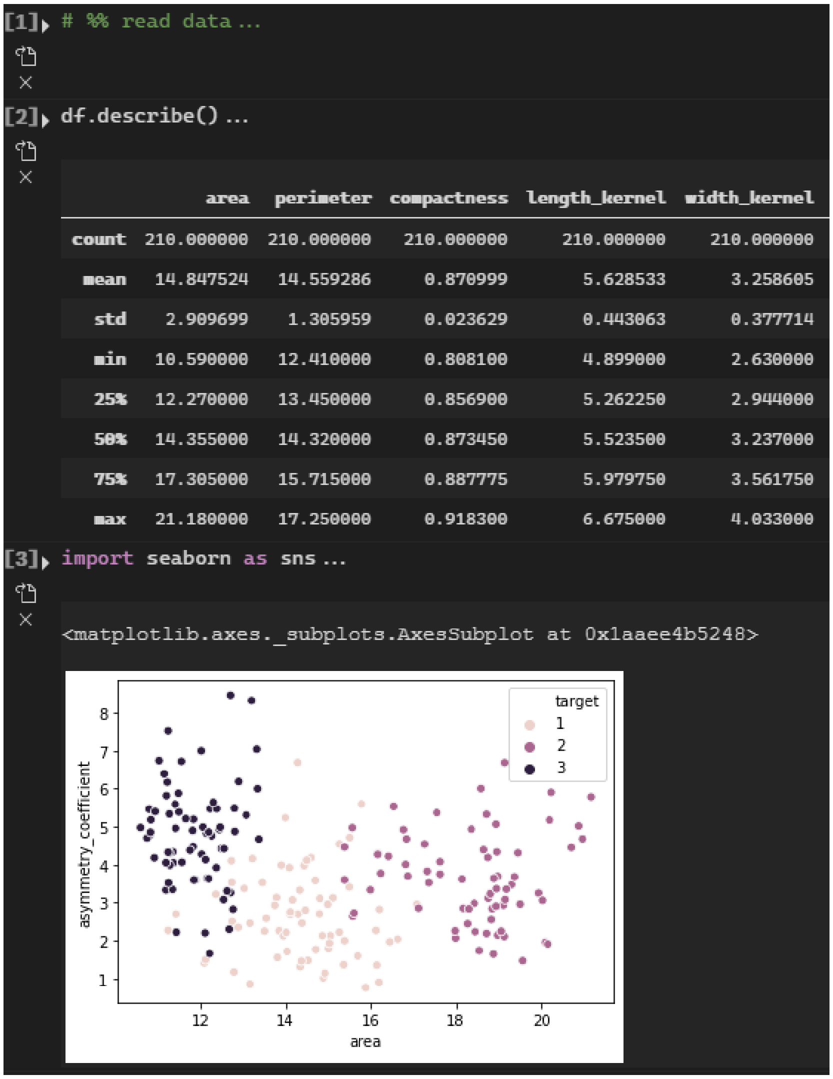
Lab Activity 3: Clustering Seeds

In this section, you will work with a wheat seed dataset with three varieties of wheat. You'll use the characteristics of the seed's internal kernel structure to build a predictive model to group the seeds into clusters.

Note: Although you have already downloaded this dataset, it is also publicly available from UCI Machine Learning Repository: Seeds Data Set (https://archive.ics.uci.edu/ml/datasets/seeds).

1. Use the skeleton file **clustering.py** and run through the first three code blocks.

   Your result should resemble the image below.



```
[1] # %% read data ...
[2] df.describe() ...
```

```
            area   perimeter  compactness  length_kernel  width_kernel
count  210.000000  210.000000   210.000000     210.000000    210.000000
mean    14.847524   14.559286     0.870999       5.628533      3.258605
std      2.909699    1.305959     0.023629       0.443063      0.377714
min     10.590000   12.410000     0.808100       4.899000      2.630000
25%     12.270000   13.450000     0.856900       5.262250      2.944000
50%     14.355000   14.320000     0.873450       5.523500      3.237000
75%     17.305000   15.715000     0.887775       5.979750      3.561750
max     21.180000   17.250000     0.918300       6.675000      4.033000
```

```
[3] import seaborn as sns ...
<matplotlib.axes._subplots.AxesSubplot at 0x1aaee4b5248>
```

Figure 6: Executing Skeleton File
Used with permission from Microsoft.

2. Use your data mining skills to explore and understand the dataset. Plot as many figures as need to visually comprehend the relationship between different variables.

   For example, the relationship between **perimeter** and **compactness** can be visualized as shown in the image below.

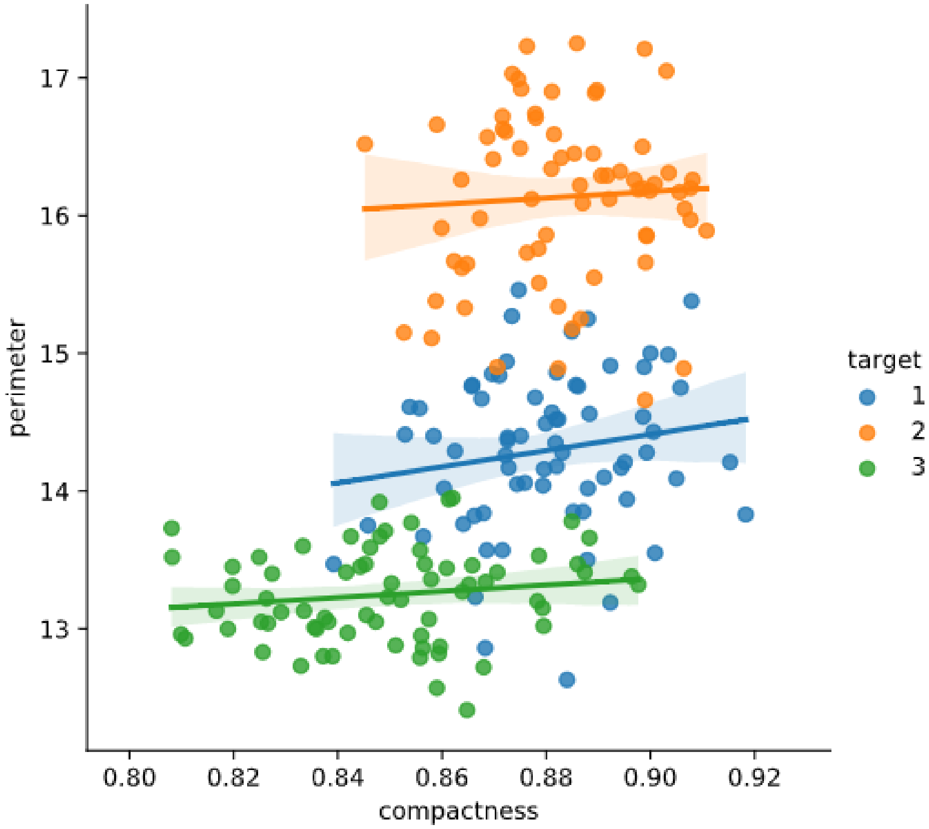


Figure 7. Exploring Perimeter vs. Compactness
© 2020, Southern Alberta Institute of Technology

3. Suppose you didn't know the exact number of wheat varieties. Determine the best number of clusters by iterating a list of candidate numbers and finding the "elbow" point for the change in inertia and/or homogeneity.
4. Complete the second-last code block, and then run the final code block.
    a. Since you know that the actual number of clusters is 3, use a list from 1 to 10 as the candidate.
    b. In each iteration, fit the data to KMeans with a different candidate number of clusters.
    c. Get the inertia score and homogeneity score and store them in a dictionary.

Your results should be similar to the image below.

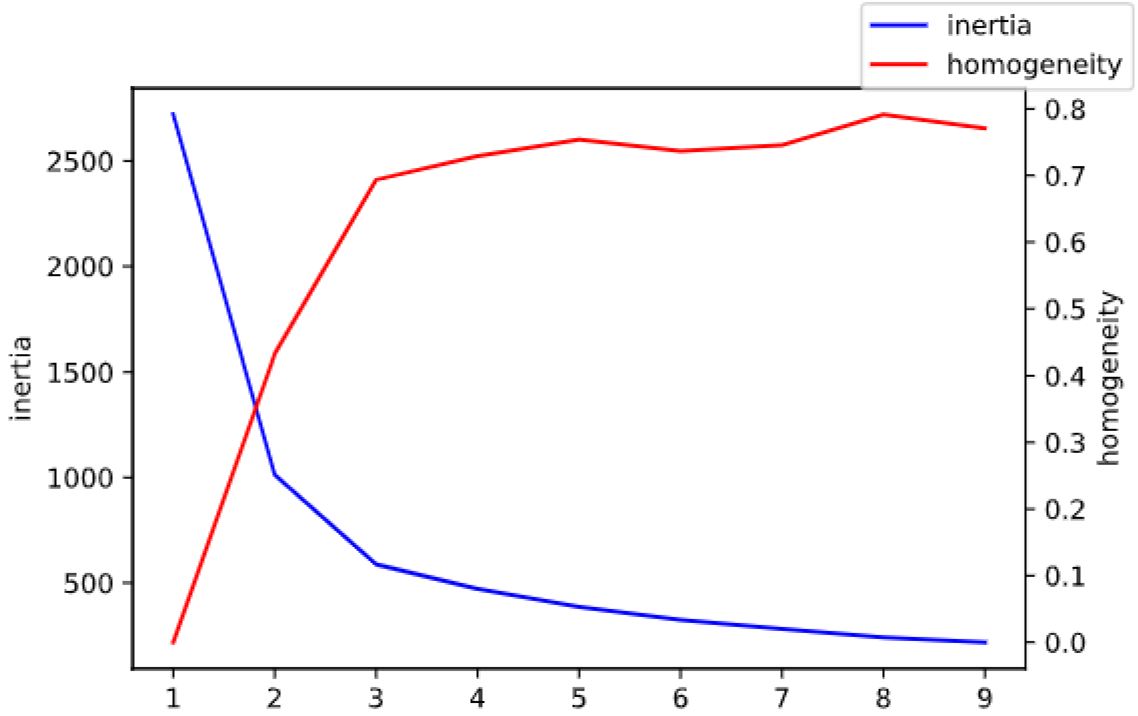


Figure 8. Determining Best Number of Clusters
© 2020, Southern Alberta Institute of Technology

References

Charytanowicz, M., Niewczas, J., Kulczycki, P., Kowalski, P. A., Lukasik, S., & Zak, S. (2010). A complete gradient clustering algorithm for features analysis of X-ray images. In Pietka, E. and Kawa, J. (Eds.), *Information technologies in biomedicine*, (pp. 15–24). Berlin: Springer-Verlag.

De Cock, D. (2011). Ames, Iowa: Alternative to the Boston housing data as an end of semester regression project. *Journal of Statistics Education, 19*(3). doi: 10.1080/10691898.2011.11889627