What I will focus on today:

* 20% - Describe: Good summary of the dataset, repository well laid-out and organised. Reasonable commits to the repository.

Create a git repository and make it available online for the lecturer to clone. The repository should contain all your work for this assessment. Within the repository, create a jupyter [6] notebook that uses descriptive statistics and plots to describe the Boston House Prices [1] dataset. This part is worth 20% of your overall mark.

1. Good summary of the dataset

Step 1: I reviewed the Boston Standard Metropolitan Statistical Area - Boston house prices dataset. <https://www.kaggle.com/c/boston-housing>

According to the description this dataset details the *Housing Values in Suburbs of Boston. Data description; The Boston data frame has 506 rows and 14 columns.*

We are provided with a ‘key’ which describes what the information in each column represents;

***crim****per capita crime rate by town.*

***zn****proportion of residential land zoned for lots over 25,000 sq.ft.*

***indus****proportion of non-retail business acres per town.*

***chas****Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).*

***nox****nitrogen oxides concentration (parts per 10 million).*

***rm****average number of rooms per dwelling.*

***age*** *proportion of owner-occupied units built prior to 1940.*

***dis****weighted mean of distances to five Boston employment centres.*

***rad****index of accessibility to radial highways.*

***tax****full-value property-tax rate per $10,000.*

***ptratio****pupil-teacher ratio by town.*

***black****1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town.*

***lstat****lower status of the population (percent).*

***medv****median value of owner-occupied homes in $1000s.*

Source;

* Harrison, D. and Rubinfeld, D.L. (1978) Hedonic prices and the demand for clean air. J. Environ. Economics and Management 5, 81–102.
* Belsley D.A., Kuh, E. and Welsch, R.E. (1980) Regression Diagnostics. Identifying Influential Data and Sources of Collinearity. New York: Wiley.

I was immediately struck by one piece of information mentioned;

***black****1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town.*

This seemed very strange to me, and slightly offensive. I wondered if this would be explained a bit more when I became more familiar with the dataset.

When I click on ‘Data’ I expected to be presented with the dataset, but nothing appeared.

I revisited Moodle to see if Ian had mentioned that we need to create an account to access the dataset.

I noticed one comment from Ian in our Outlook group which is used for news, announcements and discussion. Ian’s comment was made in reply to a question posted by one of my fellow students;

*You'll find the common datasets in many different configurations - part of the assignment is to source the dataset in a form conducive to what you need to do with it.*

This could be important knowledge for me when completing this assignment. It also explains why I couldn’t immediately view the dataset in <https://www.kaggle.com/c/boston-housing>.

I discovered the below website which seemed to explain how to load the dataset in Jupyter Notebook;

<https://subscription.packtpub.com/book/programming/9781789804744/1/ch01lvl1sec11/our-first-analysis-the-boston-housing-dataset>

This website describes the dataset as follows;

*The dataset we'll look at in this section is the so-called Boston housing dataset. It contains US census data concerning houses in various areas around the city of Boston. Each sample corresponds to a unique area and has about a dozen measures. We should think of samples as rows and measures as columns. The data was fist published in 1978 and is quite small, containing only about 500 samples.*

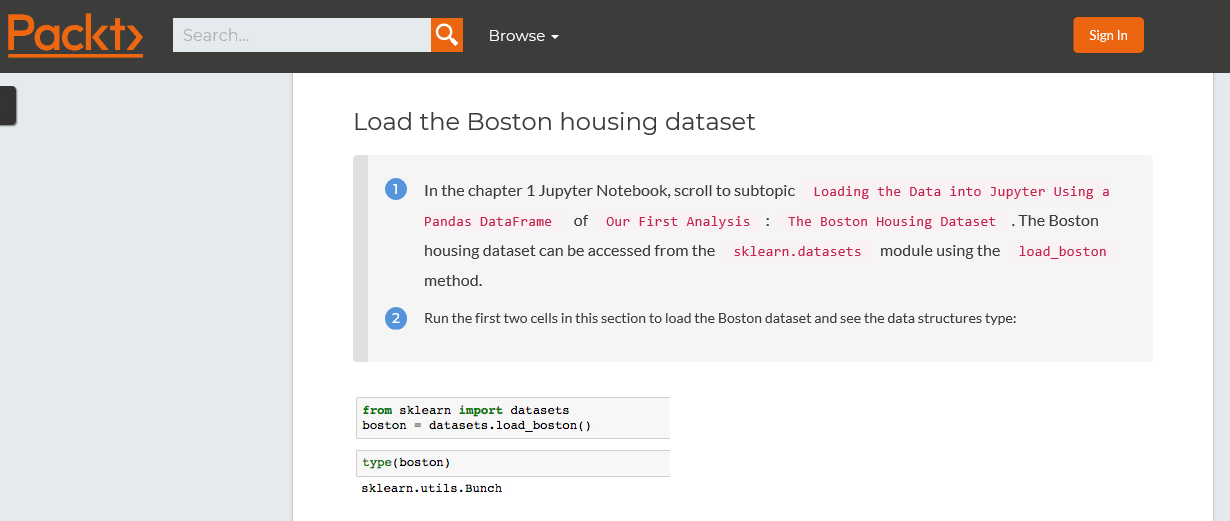
The website describes how to complete the following; “Load the data into Jupyter using a Pandas DataFrame”.

***Loading the Data into Jupyter Using a Pandas DataFrame***

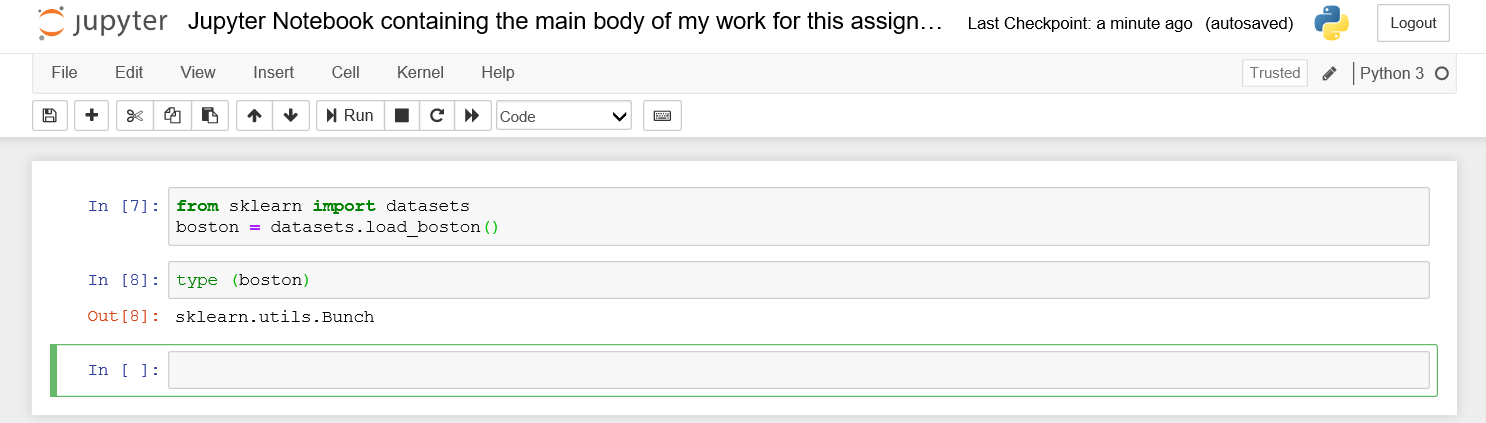
*Oftentimes, data is stored in tables, which means it can be saved as a comma-separated variable (CSV) file. This format, and many others, can be read into Python as a DataFrame object, using the Pandas library. Other common formats include tab-separated variable (TSV), SQL tables, and JSON data structures. Indeed, Pandas has support for all of these. In this example, however, we are not going to load the data this way because the dataset is available directly through scikit-learn. (*<https://subscription.packtpub.com/book/programming/9781789804744/1/ch01lvl1sec11/our-first-analysis-the-boston-housing-dataset>)

It also mentions; “*An important part after loading data for analysis is ensuring that it's clean. For example, we would generally need to deal with missing data and ensure that all columns have the correct datatypes. The dataset we use in this section has already been cleaned, so we will not need to worry about this.”*

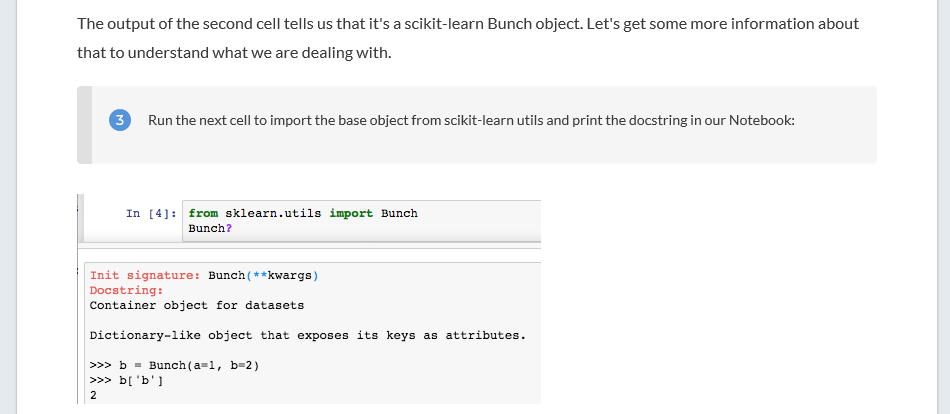
I followed the website’s recommended steps as follows;



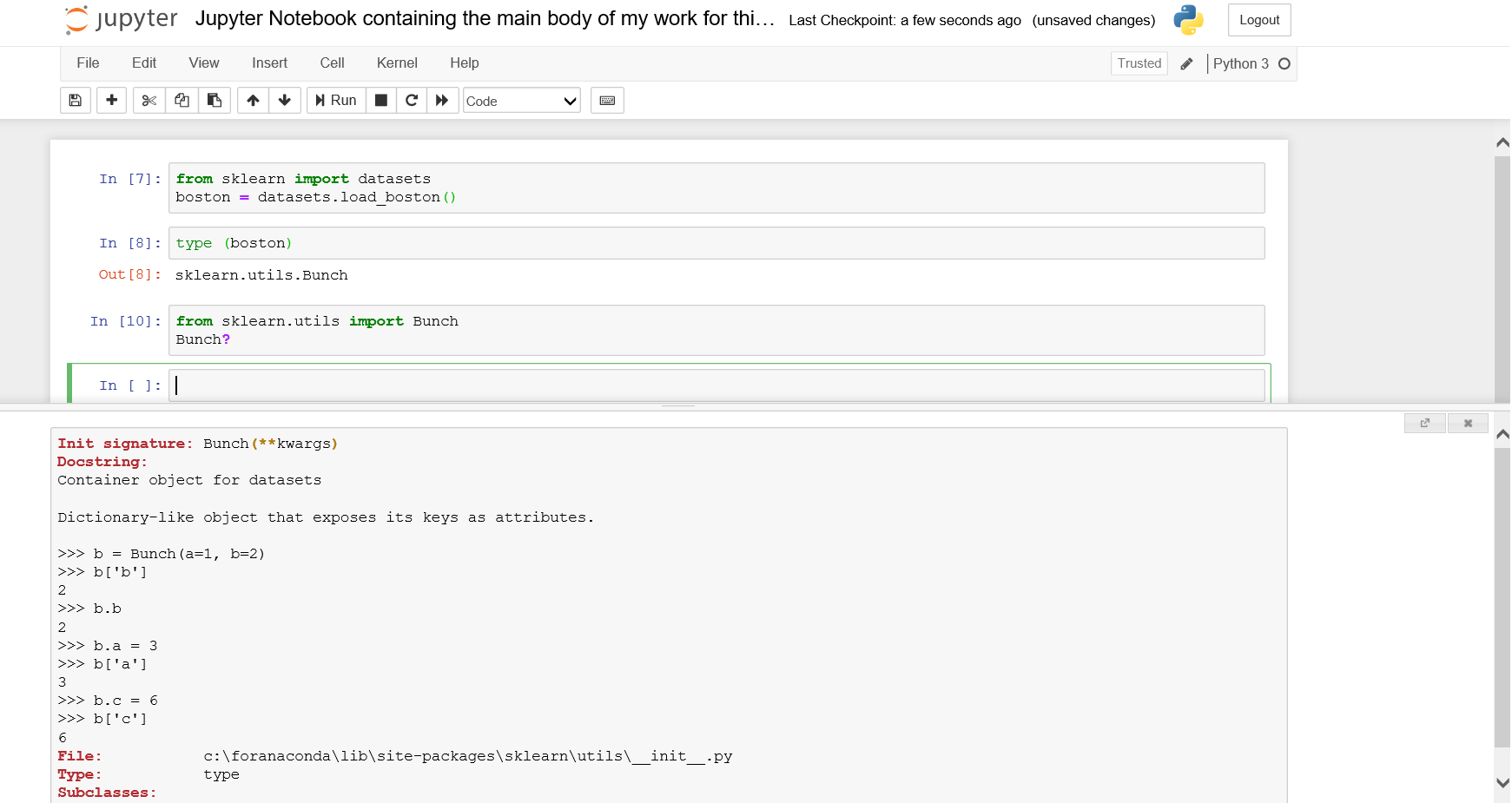
I followed their code as above in my Jupyter notebook and also got the same output;



I continued to follow the instructions from the website;

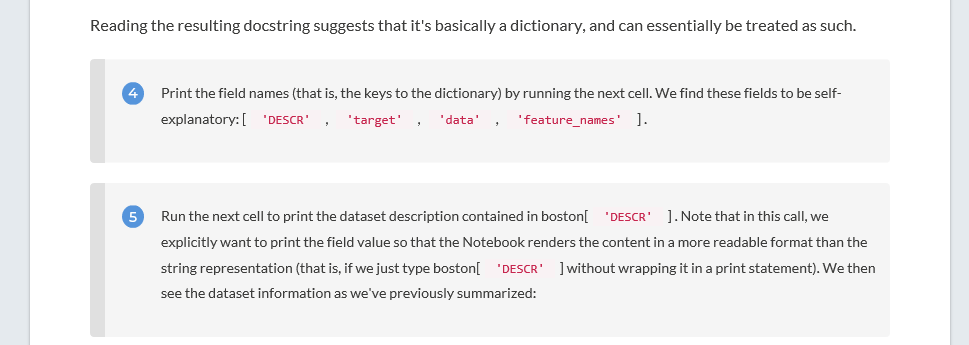


I encounter the same results as displayed on the website in my Jupyter Notebook;

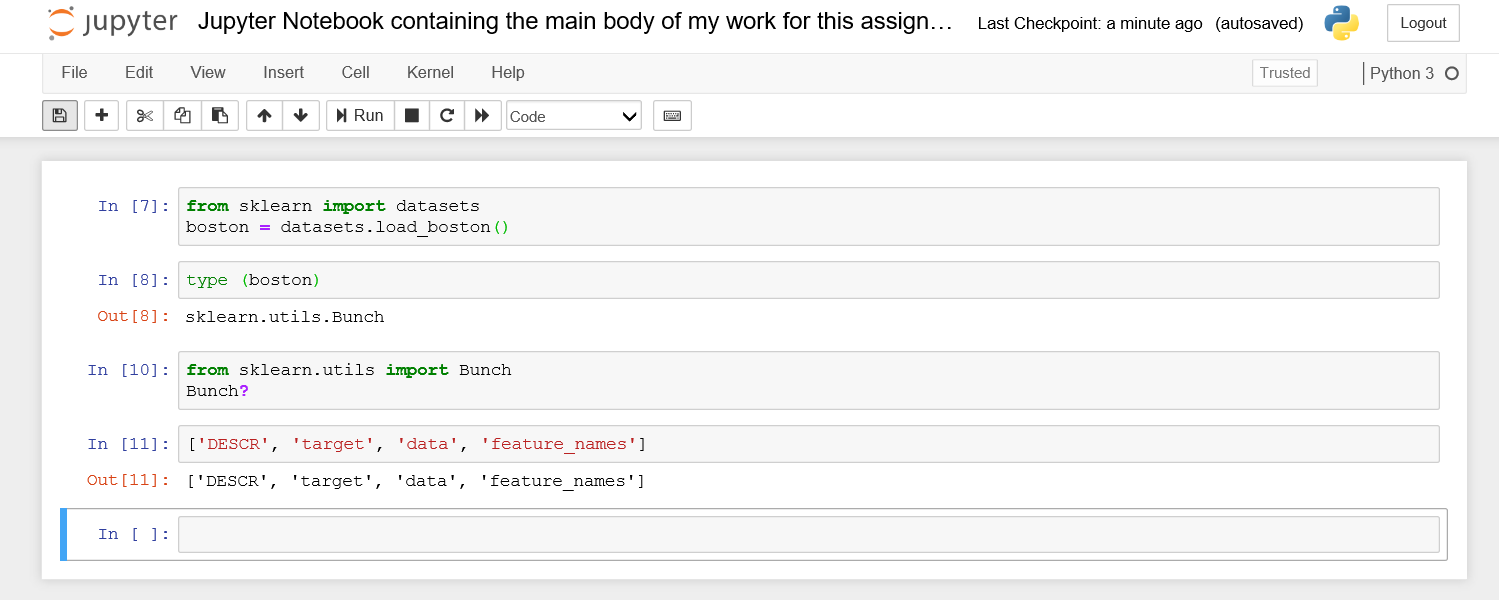


The website explains that ‘Reading the resulting docstring suggests that it's basically a dictionary, and can essentially be treated as such.’ (<https://subscription.packtpub.com/book/programming/9781789804744/1/ch01lvl1sec11/our-first-analysis-the-boston-housing-dataset>)

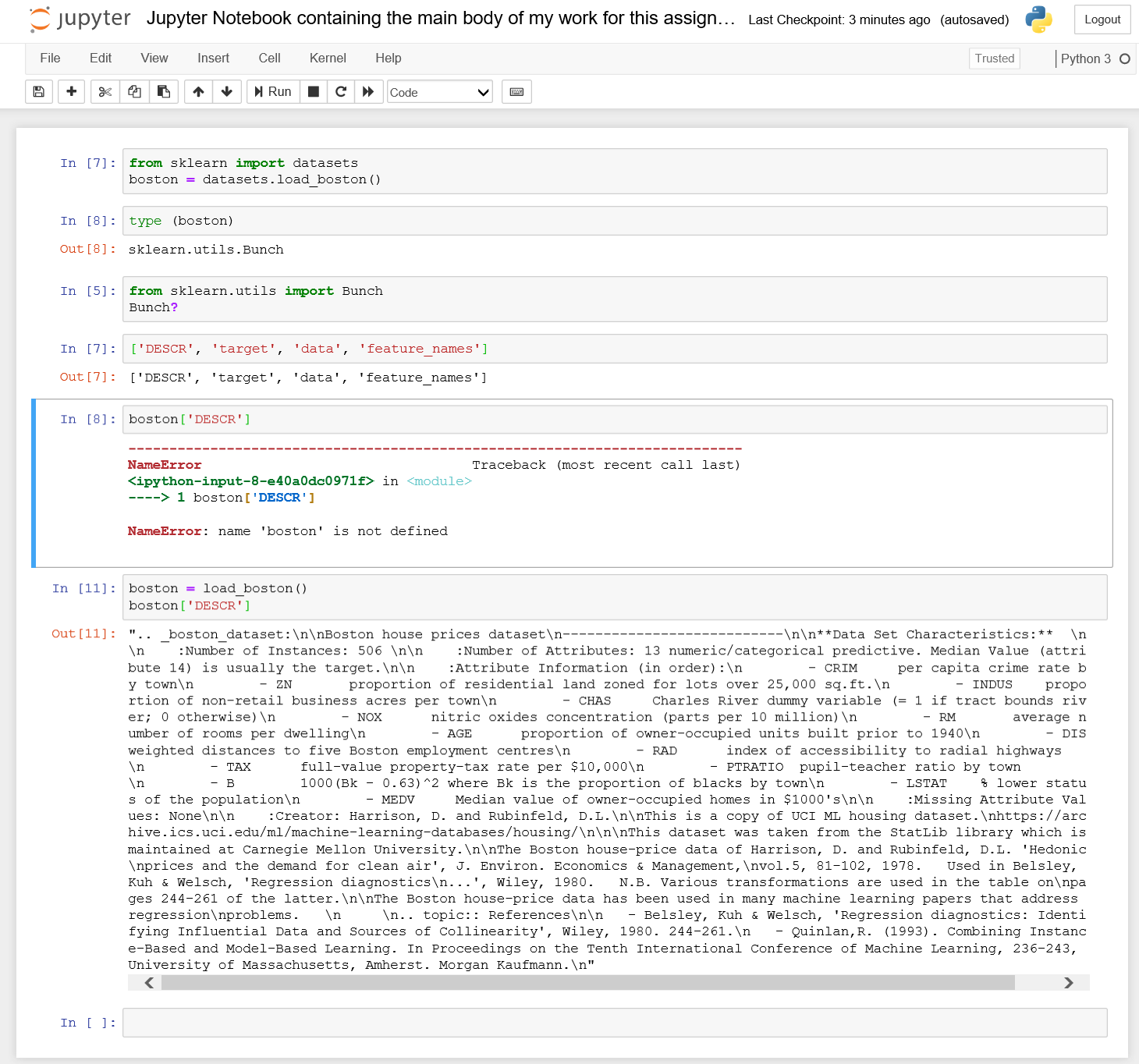
The website guides me on to the next steps;



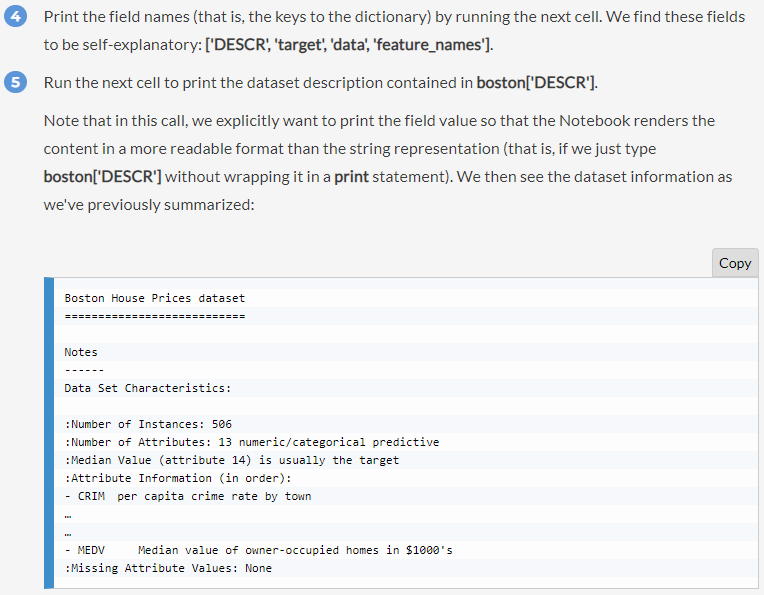
I stopped at this point as I became confused;



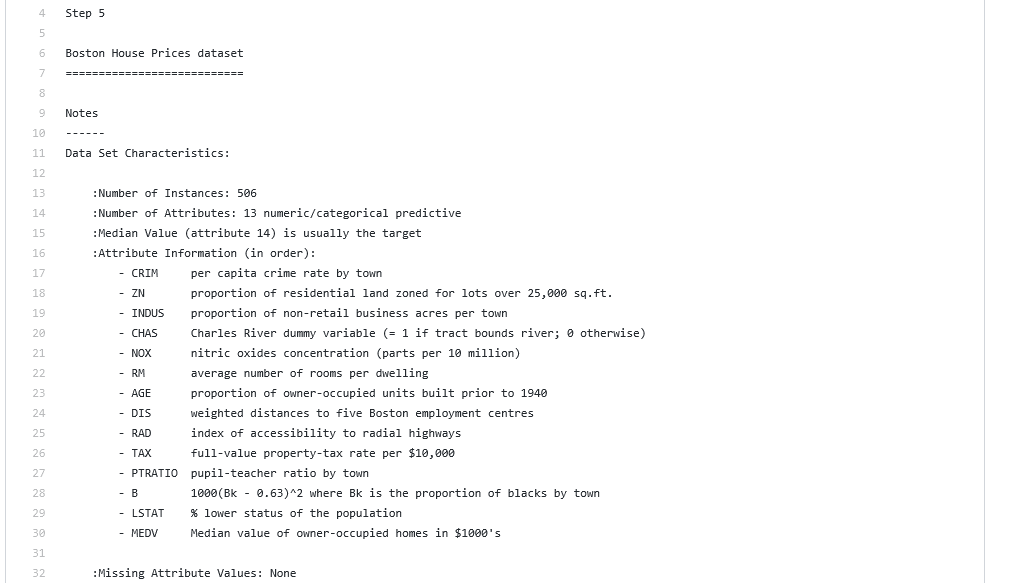
I picked up at this point again the following day and had to play around with some different code before I managed to achieve the below;



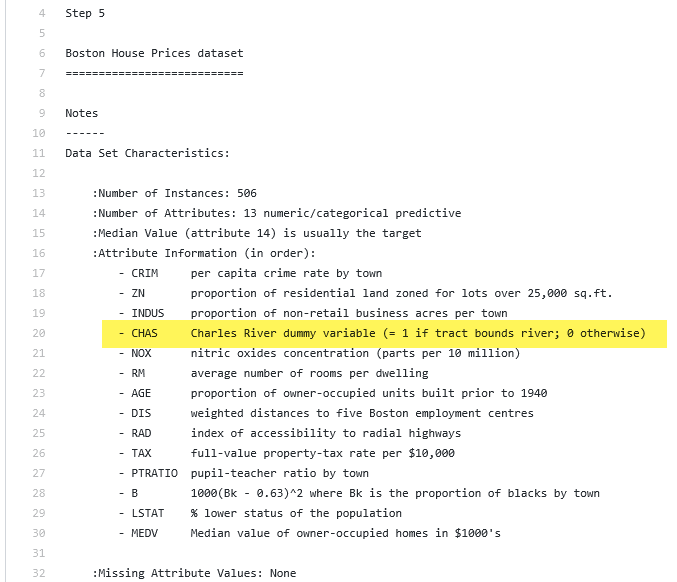
I managed to load the dataset into my Jupyter notebook. This resembles the outcome of Step 5 as described on <https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781789958171/1/ch01lvl1sec04/our-first-analysis-the-boston-housing-dataset>;



We can see that the Boston House Prices dataset in my Jupyter notebook is not as nicely laid out as described on the website. I need to re-examine my code to see how I can achieve the more structured look as shown on the website (https://github.com/TrainingByPackt/Applied-Data-Science-with-Python-and-Jupyter/blob/4532893e826e627f3009f572ad10b6b950141970/Lesson%201/Lesson%201.txt#L6-L33);

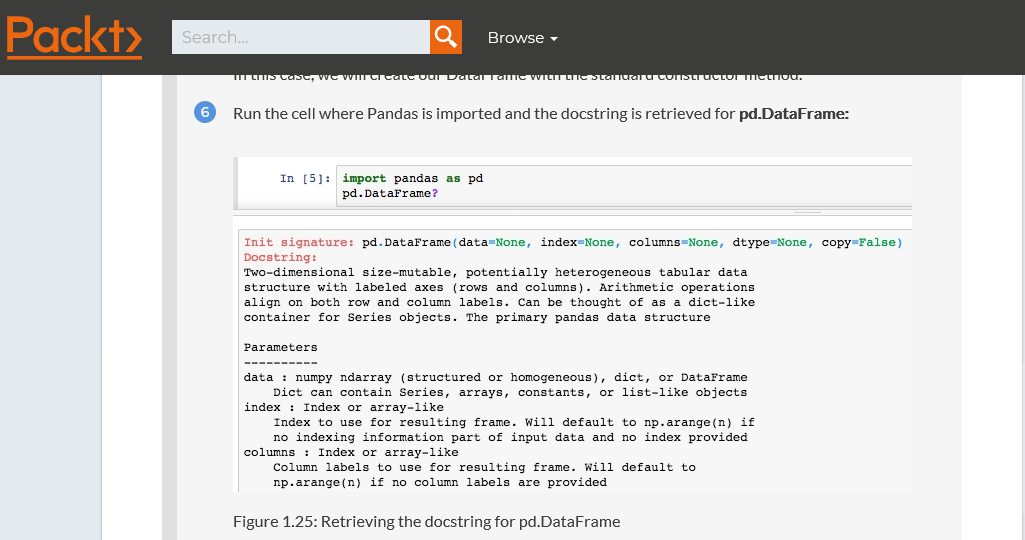


We can see a reference to the ‘Charles River’ which I will need to address in the second part of this assignment.

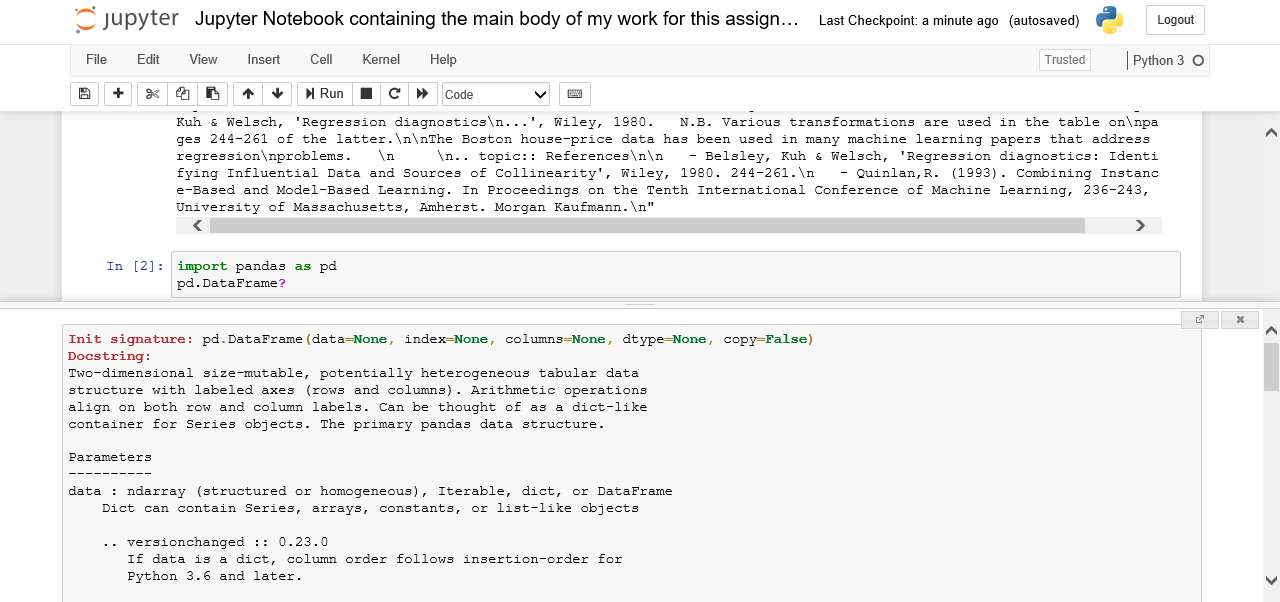


I have managed to load the dataset into my Jupyter notebook – now I need to figure out how to ‘use descriptive statistics and plots to describe the Boston House Prices dataset’. I should watch Ian’s videos for instruction on how to complete this.

I continued on to step 6;

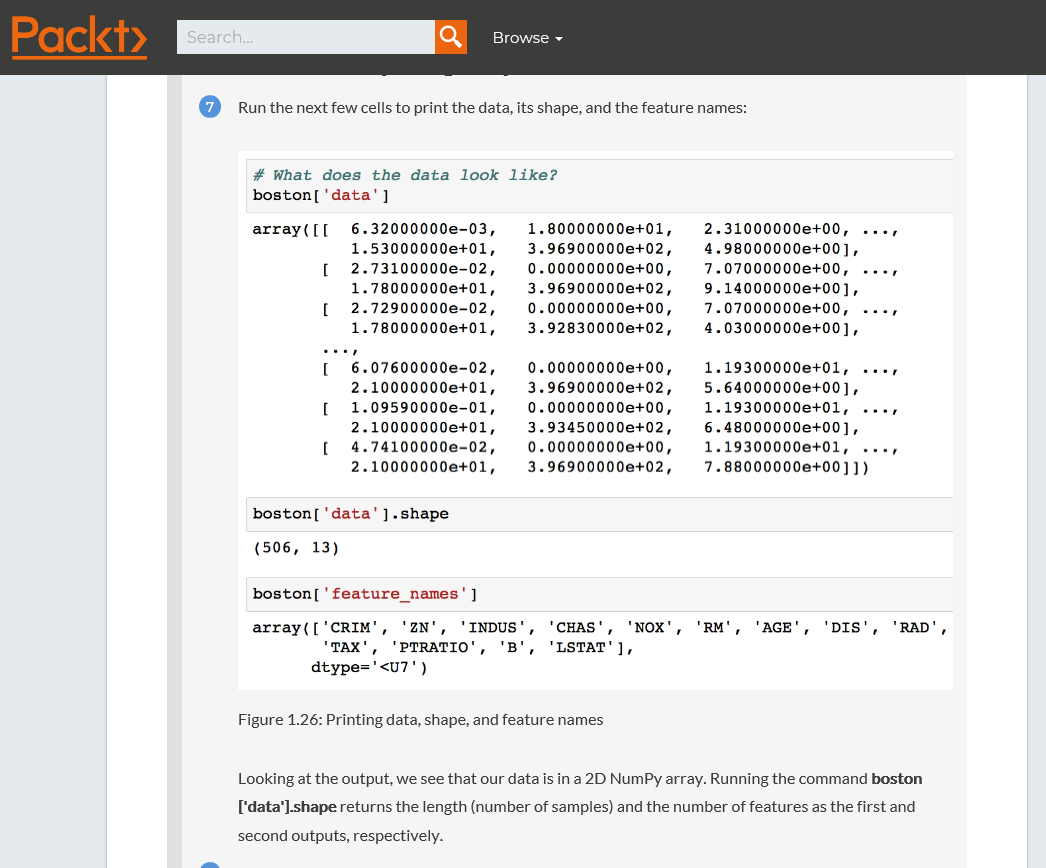


I achieved the same result in my Jupyter Notebook;

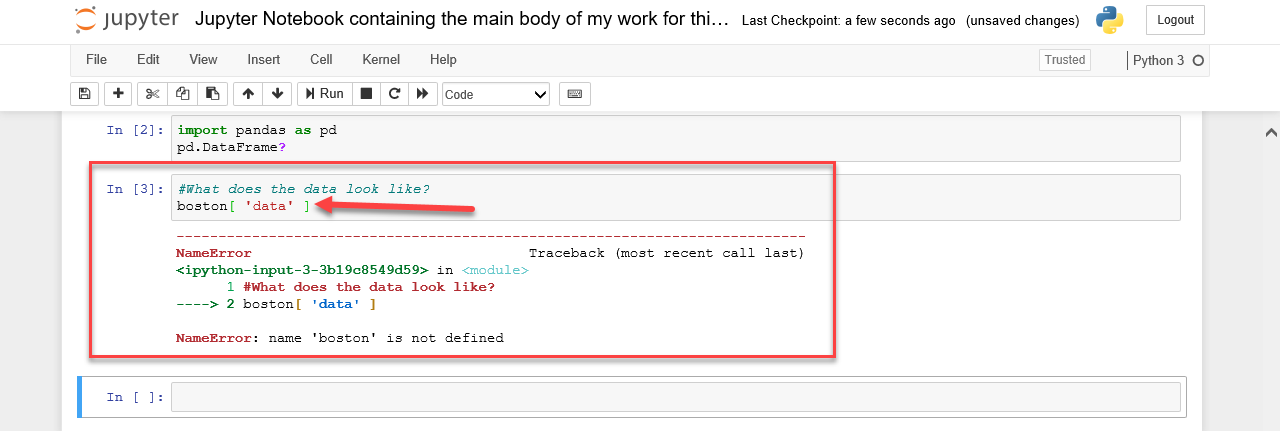


The website informs me that the ‘The docstring reveals the DataFrame input parameters. We want to feed in boston['data'] for the data and use boston['feature\_names'] for the headers,’ (<https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781789958171/1/ch01lvl1sec04/our-first-analysis-the-boston-housing-dataset>).

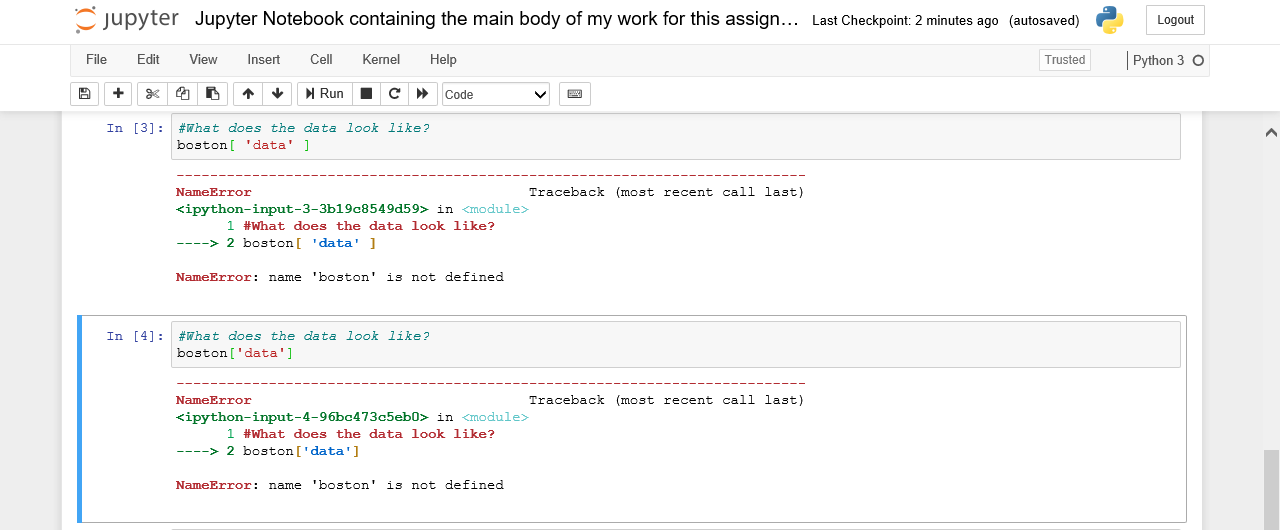
Progressing on to Step 7;



I tried to copy the above, and initially made a mistake as I encountered the below error;



I realised I had inserted 2 spaces before & after the ‘data’ which I should not have done – I amended this and tried again;



I got the same error again ‘NameError: name 'boston' is not defined’.

I establish that there must be something wrong with a previous step in my Jupyter Notebook, and I suspect it is that I have done something wrong at step 4.

I searched online for guidance and found some information pertaining to this error message;

***NameError: global name '---' is not defined***

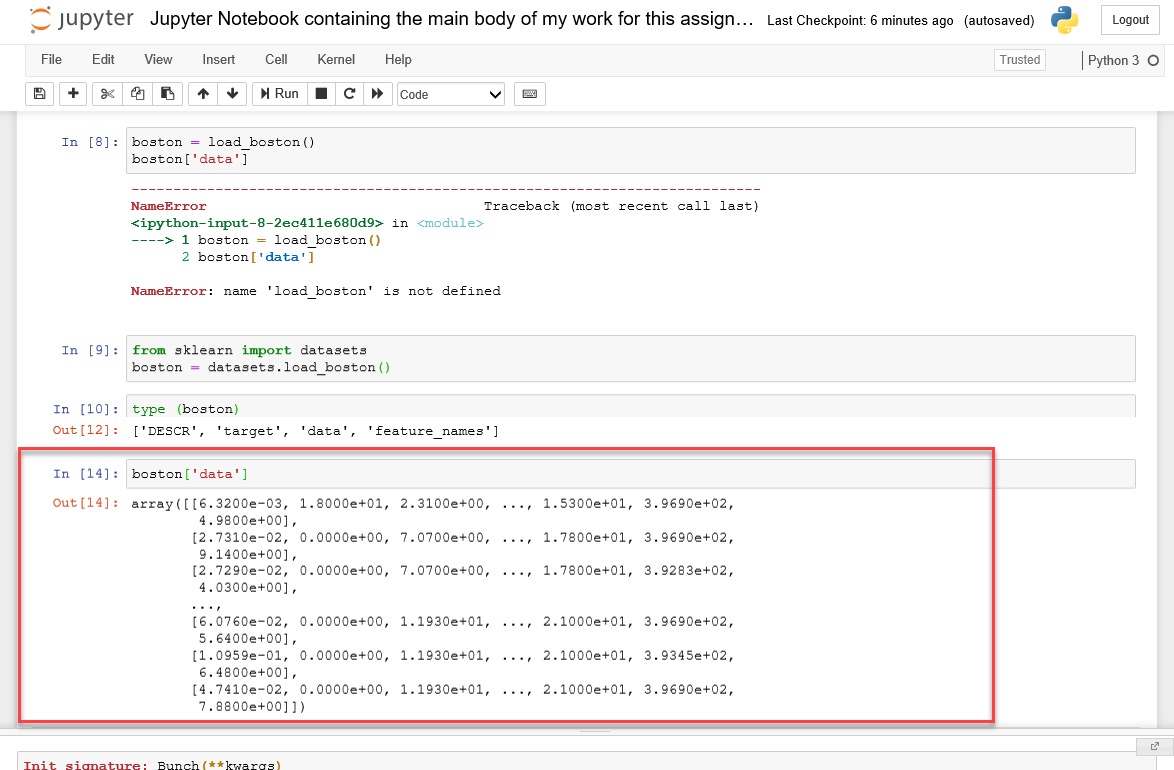
*Python knows the purposes of certain names (such as names of built-in functions like print). Other names are defined within the program (such as variables). If Python encounters a name that it doesn't recognize, you'll probably get this error.*

*Some common causes of this error include:*

* *Forgetting to give a variable a value before using it in another statement*
* *Misspelling the name of a built-in function (e.g., typing "inpit" instead of "input")*

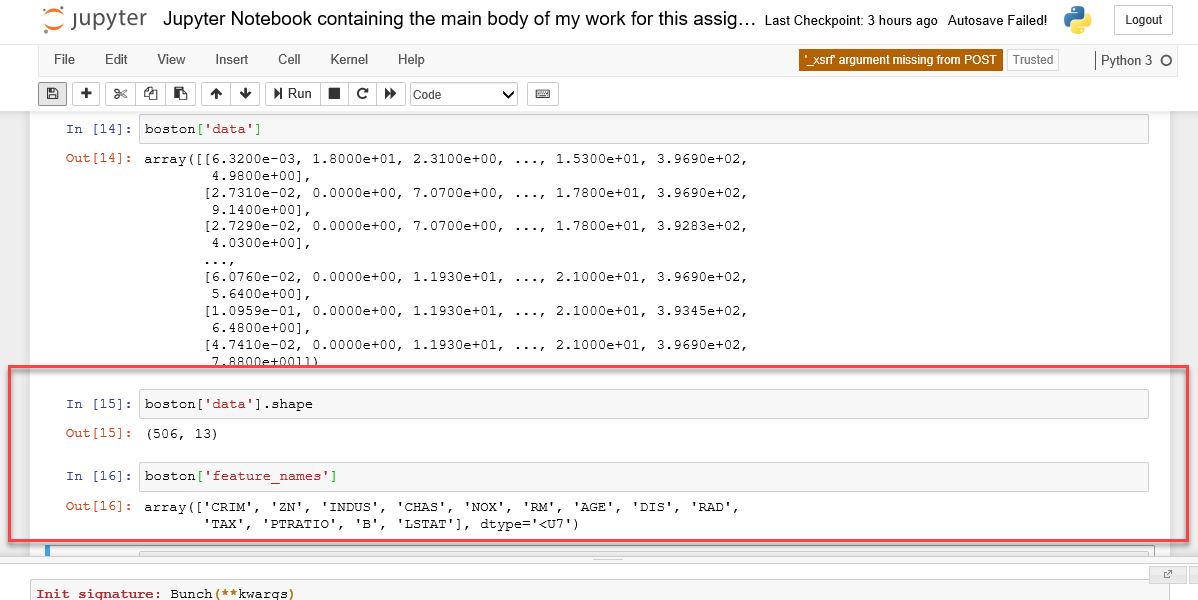
(Source: <https://www2.cs.arizona.edu/people/mccann/errors-python#Four>)

I played around with my Jupyter Notebook, changing the code in the cells and eventually realised that if I ran all steps again from 1 to 7 it worked correctly for me;



What a relief!

Now my results sync up once again with the results displayed on the website.



As per the website;

*Looking at the output, we see that our data is in a 2D NumPy array. Running the command* ***boston['data'].shape*** *returns the length (number of samples) and the number of features as the first and second outputs, respectively.*

At this point I compared the results in Out[16] above with and noticed 2 discrepancies;

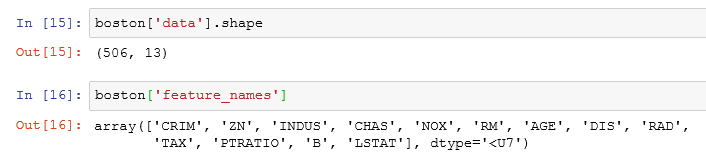
1. ***medv –*** This is included in the Boston Standard Metropolitan Statistical Area - Boston house prices dataset as per <https://www.kaggle.com/c/boston-housing>

According to this description, this dataset details the *Housing Values in Suburbs of Boston. Data description; The Boston data frame has 506 rows and 14 columns.*

We are provided with a ‘key’ which describes what the information in each column represents, and ***medv (median value of owner-occupied homes in $1000s)***

is one of the 14 keys.

However, ***medv*** is not included in the output in my Jupyter Notebook or on the <https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781789958171/1/ch01lvl1sec04/our-first-analysis-the-boston-housing-dataset> website. According to Jupyter Notebook and the website, there are only 13 keys;

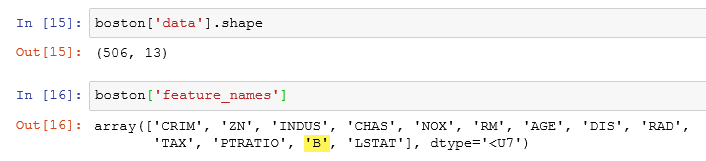


1. ***black -*** This is included in the Boston Standard Metropolitan Statistical Area - Boston house prices dataset as per <https://www.kaggle.com/c/boston-housing>

According to this description, this dataset details the *Housing Values in Suburbs of Boston. Data description; The Boston data frame has 506 rows and 14 columns.*

We are provided with a ‘key’ which describes what the information in each column represents, and ***black (1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town)*** is one of the 14 keys.

However, this is represented by ‘B’ in my Jupyter Notebook and on the <https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781789958171/1/ch01lvl1sec04/our-first-analysis-the-boston-housing-dataset> website;



I continued on to Step 8 as per the <https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781789958171/1/ch01lvl1sec04/our-first-analysis-the-boston-housing-dataset> website;

8. Load the data into a Pandas DataFrame df by running the following:

df = pd.DataFrame(data=boston['data'],

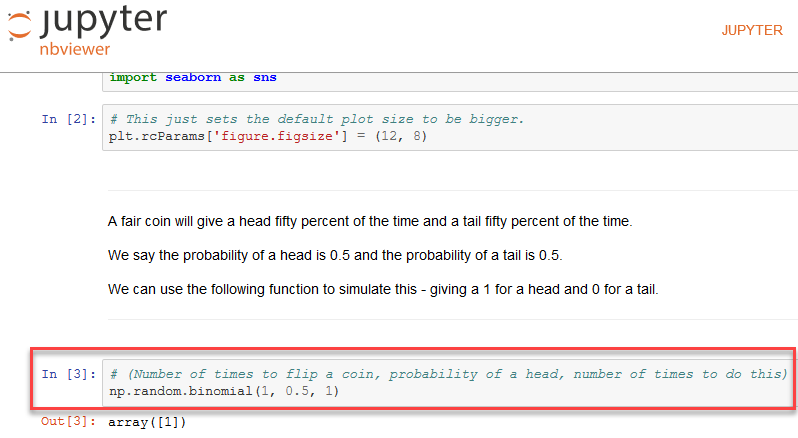
columns=boston['feature\_names'])

Note: I stopped at step 13. Continue from here tomorrow. Continue on with Exercise 5 & adapt it for the Charles river thing. Also Exercise 6 & maybe 7.

Having arrived at this stage, I decided to review some of Ian’s videos on Moodle again to see if I can figure out how to use descriptive statistics and plots to describe the Boston House Prices dataset. I noticed in *Coin flipping in Python* <https://web.microsoftstream.com/video/2cd606c0-3269-4a2d-975d-f089514b8bfc?referrer=https:%2F%2Flearnonline.gmit.ie%2Fcourse%2Fview.php%3Fid%3D136> Ian describes how to replicate the flipping of a coin in his Jupyter Notebook at 4 mins;

# (Number of times to flip a coin, probability of a head, number of times to do this)

np.random.binomial(1, 0.5, 1)

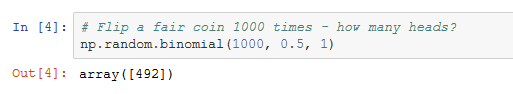


Maybe this will be useful to me at a later stage.

This technique is designed to give an uncertain output for the same input. ‘Stochastic’ – relying on some type of a probability distribution for the outcome – as described by Ian. 8mins. Ian supplied some code in a Jupyter Notebook (<https://nbviewer.jupyter.org/github/ianmcloughlin/jupyter-teaching-notebooks/blob/master/coin-flip.ipynb>) describing this;

# Flip a fair coin 1000 times - how many heads?

np.random.binomial(1000, 0.5, 1)



14 mins – seaborn – gives an idea of how many times something has happened in the form of a plot.

# We'll use numpy and scipy.stats to analyse flipping a coin.

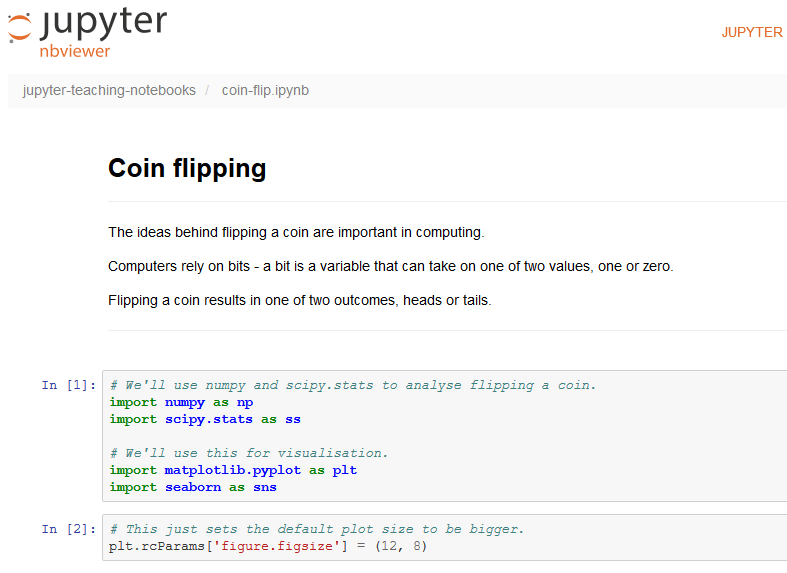
import numpy as np

import scipy.stats as ss

# We'll use this for visualisation.

import matplotlib.pyplot as plt

import seaborn as sns



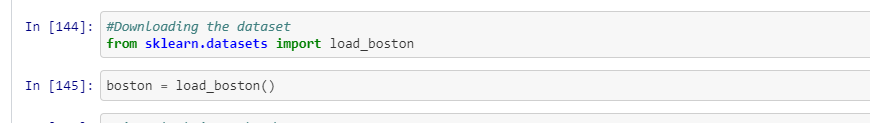
Maybe I could apply this to one element of the dataset (e.g. ***indus****proportion of non-retail business acres per town, or*

***chas****Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).)*

Website mentioned by Ian <https://seaborn.pydata.org/generated/seaborn.distplot.html>

~~I decided at this point to move on to a different set of videos set up on Moodle by Dr. Ian McLoughlin to review the Dog Dataset I saw mentioned in our class Outlook group (Training neural networks). I thought I would investigate to see if any descriptive statistics and plots were employed for the Dog Dataset which I could use in the same manner to describe the Boston House Prices dataset.~~

From M:



**My Jupyter Notebook should use descriptive statistics and plots to describe the Boston House Prices dataset.**

[~~https://www.ritchieng.com/machine-learning-project-boston-home-prices/~~](https://www.ritchieng.com/machine-learning-project-boston-home-prices/)