**Project 14**

**Load Dataset-**

 Use pandas to explore the data both with descriptive statistics and data visualization. that we are specifying the names of each column when loading the data.



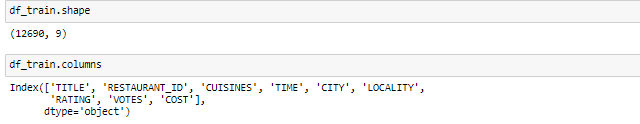
## Summarize the Dataset-

## Steps are using to take a look at the data a few different ways:

1. Dimensions of the dataset.
2. Peek at the data itself.
3. Statistical summary of all attributes.
4. Breakdown of the data by the class variable.

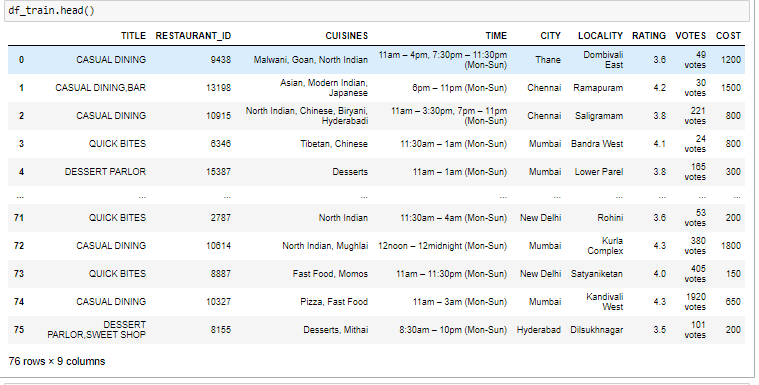
### Dimensions of Dataset-

From this we have to check how many data rows and columns actually present in datasets.



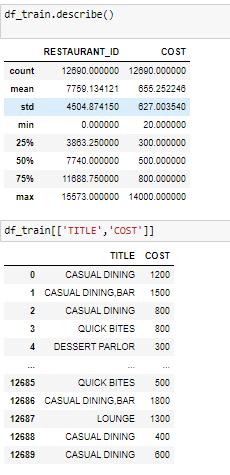
### Peek at the Data-

Checking the overview of the data with print (df) or df.head()

.

### Statistical Summary-

After checking overview of the data from df.head we have to go with Stats part from we have to find summary of each attribute. This includes the count, mean, the min and max values as well as some percentiles.

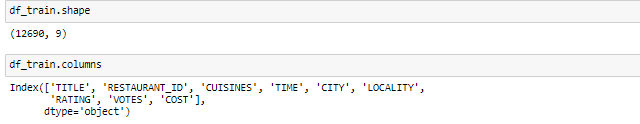


## Data Visualization

In this part we have to visualize data from different form .

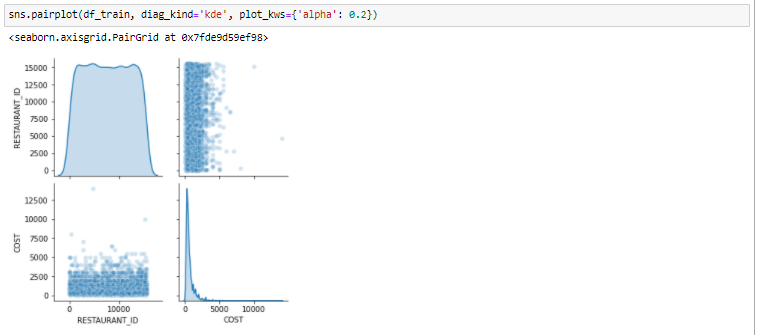
1. Univariate plots to better understand each attribute.
2. Multivariate plots to better understand the relationships between attributes.

### Univariate Plots-



### Multivariate Plots

Now we can look at the interactions between the variables. First, let’s look at scatterplots of all pairs of attributes. This can be helpful to spot structured relationships between input variables.



## Evaluating Some Algorithms

create some models of the data and estimate their accuracy on unseen data.

1. Separate out a validation dataset.
2. Set-up the test harness to use 10-fold cross validation.
3. Build multiple different models to predict species from flower measurements
4. Select the best model.

### Create a Validation Dataset-

 we will use statistical methods to estimate the accuracy of the models that we create on unseen data. We also want a more concrete estimate of the accuracy of the best model on unseen data by evaluating it on actual unseen data.

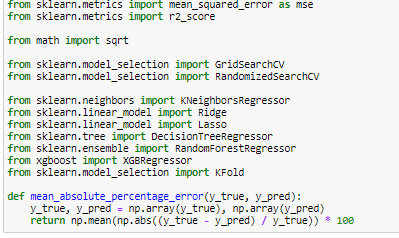
That is, we are going to hold back some data that the algorithms will not get to see and we will use this data to get a second and independent idea of how accurate the best model might actually be. We will split the loaded dataset into two, 80% of which we will use to train, evaluate and select among our models, and 20% that we will hold back as a validation dataset.



You now have training data in the *X\_train* and *Y\_train* for preparing models and a *X\_validation* and *Y\_validation* sets that we can use later.

Notice that we used a python slice to select the columns in the NumPy array.

### Build Models



Algorithms would be good on this problem or what configurations to use.We get an idea from the plots that some of the classes are partially linearly separable in some dimensions, so we are expecting generally good results.

* KNeighborsRegressor
* RandomForestRegressor
* XGBRegressor

### Select Best Model

We have models and accuracy estimations for each. We need to compare the models to each other and select the most accurate. Running the example above, we get the following raw results.