Visualization

Why and How

Before you begin your Data Science project

You must understand a few things:

1. What is the business problem I am trying to solve?
   - Problem Statement
2. What is the Outcome I am trying to achieve or predict?
   - Outcome variable
3. What kind of data do I have? Does it have all the information that I want captured to do a thorough analysis? If not, gather more data or wait.
   - Data Gathering
4. Once I have the data, what do I know about the business problem that:
   a) I can generate possible hypotheses that can predict the outcome
      - Hypothesis Generation
   b) I can explore possible features that will impact the outcome
      - Feature Engineering
   c) I can list one or more methods to use to solve the problem
      - Modeling
   d) I can evaluate whether the methods are appropriate and working
      - Validation

Problem Definition and Hypotheses Generation

Once I have the data, what do I know about the business problem that:

a) I can generate possible hypotheses that can predict the outcome
   - Hypothesis Generation example

   MRP of an item is an important factor when customers purchase an item. Similarly Location and Age of a Store determines its Sales.

b) I can explore possible features that will impact the outcome
   - Feature Engineering example

   Age of a Store and MRP may be factors in purchasing Items. However, Item ID and Outlet ID is not.

c) I can list one or more methods that I can use to solve the problem
   - Modeling

   Since the prediction is of the Numeric type, we can use a Random Forest Regressor

d) I can evaluate whether the methods are appropriate and working

   Use RMSE Score

Data Load & Preparation

Process to prepare data to be ready for Visualization using Pandas

Data Visualization

Powerful Graphing and Visualizing Capabilities of Matplotlib, Seaborn

Visualize Data Relationships

Powerful Python Libraries: Matplotlib and Seaborn

Python has powerful libraries for Visualization.
We are going to explore a couple:

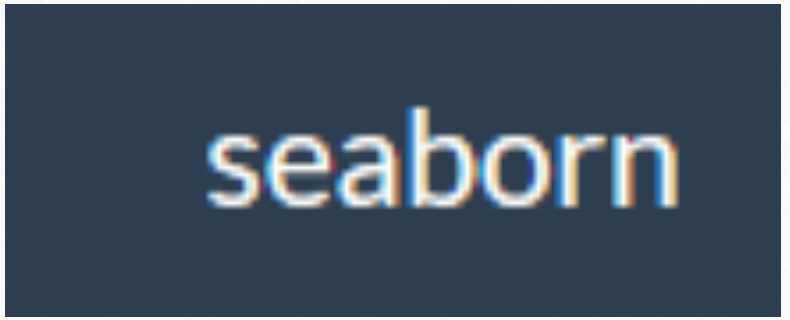
- Matplotlib and
- Seaborn

Matplotlib:

- Provides powerful publication quality 2D plots
- New tools added to augment Matplotlib:
  - Matplot3D, Basemap, Canopy

Seaborn:

- Developed at Stanford
- Statistical visualization built on Matplotlib
- High level interface provided to plot statistical measures
- https://web.stanford.edu/~mwaskom/software/seaborn/



Sources: Seaborn, Matplotlib

First Classify Variables into Categorical and Continuous

Then you can follow the simple rules below to quickly understand your Data Set

Rule #1

You must plot Continuous Variables through Histograms

- plt.figure()
- df[int_vars[4]].plot(kind='hist')

Rule #2

You must plot Categorical Variables by grouping and then summing or averaging over a Bar Chart

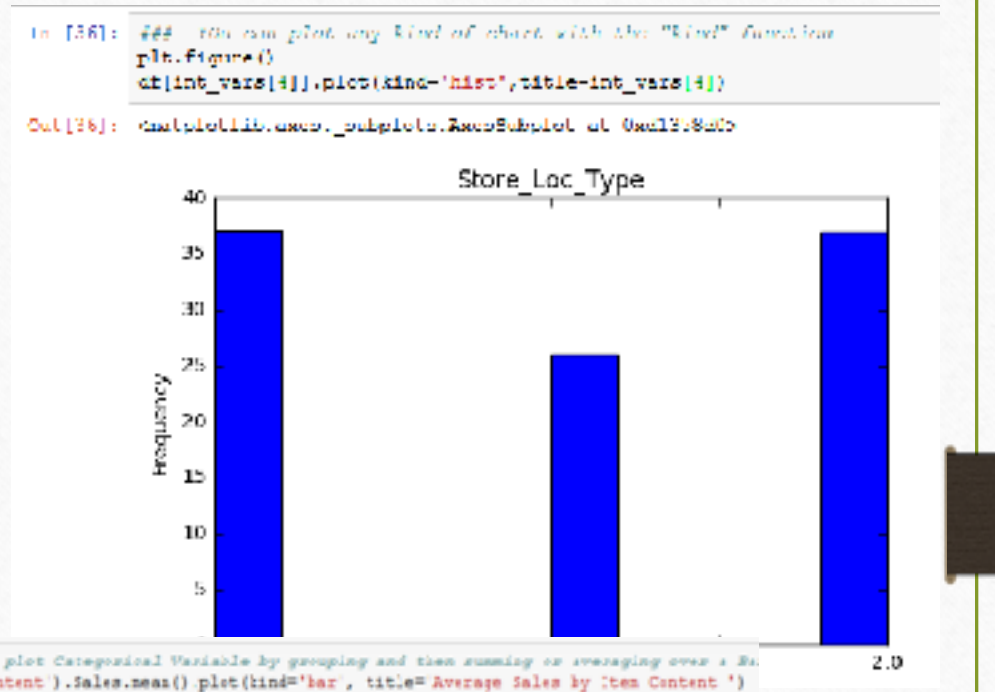
- df.groupby('Content').Sales.mean().plot(kind='bar', title='Average Sales by Item Content ')

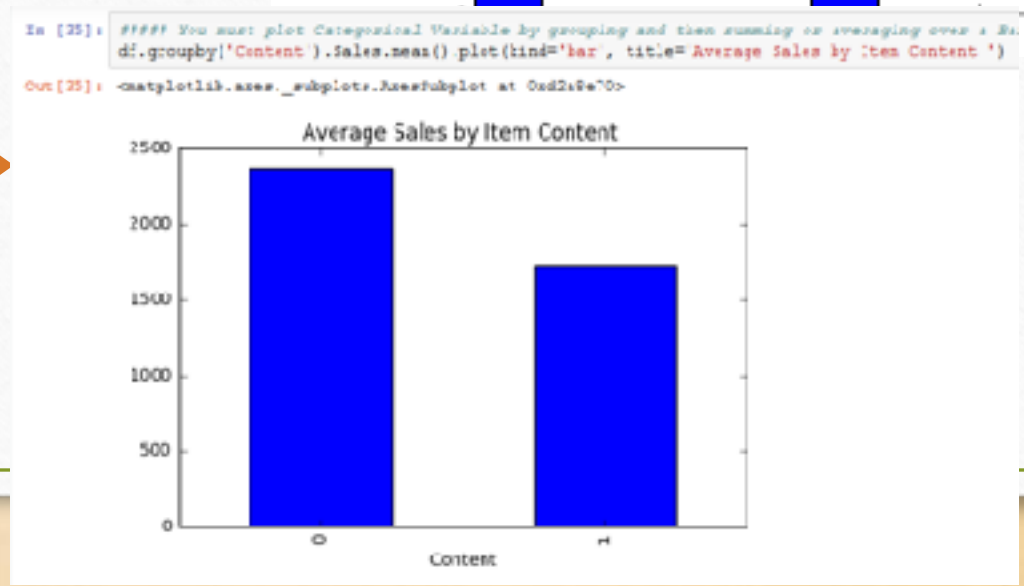
Rule 3

Your Y Variable must always be a number (numeric)

Your X variable can be any of the below

```
int_vars = [x for x in list(df) if df[x].dtype=='int64']
cont_vars= [x for x in list(df) if df[x].dtype=='float64']
cat_vars = [x for x in list(df) if df[x].dtype=='object']
print(int_vars,'\n', cont_vars,'\n', cat_vars)
```





Ram's Simple Rules to get Started on Visualization

Simple Rules to follow based on my experience

First Classify your variables into these three kinds of variables:
    # Nominal (or Categorical), Numeric and Ordinal (or Ordering)

Second, use these Data Visualization Rules as Rules of Thumb:
    # 1. if Categorical is on X and Categorical on Y, use a SidebySide Bar plot or Stacked Bar Plot or CrossTab plot
    # 2. If categorical X and numeric Y, you use a Bar Plot
    # 3. If numeric X and numeric Y , use a scatter plot
    # 4. If X is categorical and univariate, if levels less than 5, use Pie Chart
    # 5. If X is categorical and univariate, if levels greater than 5, use Bar Chart
    # 6. If numeric, univariate use a Histogram (min and max values taken into account)
    # 7. Use a HeatMap when you have to show degrees of something, such as Correlations among multiple indices
    # 8. For a mix of numeric and ordinal variables, use a Correlation Matrix (HeatMap)
    # -- Use Pearson Coeff for Correlation when your data is Continuous and Linear
    # -- Use Spearman Coeff for Correlation when your data is Ordinal and perhaps NL