Loading Important Libraries

The first step is to import our libraries (pandas, seaborn and matplotlib.pyplot to begin with) and read the CSV file containing our avocado pricing data into a pandas DataFrame.

In [ ]:
```python
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

Data Loading and Description

The Avocado dataset includes consumption of fruit in different regions of USA from 2015 till 2018 years of data.

We have two types of Avocado available Organic (Healthy) Conventional

Data Inspection Now we will inspect the data to see what it looks like in our dataframe. We use the .head() method in pandas to see the first 5 rows of the data.

In [7]:
```python
import pandas as pd
df=pd.read_csv("https://raw.githubusercontent.com/dsrscientist/Data-Science-ML-Capstone-Projects/master/avocado.csv")
df.head()
```

Out[7]:

	Unnamed: 0	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region
0	0.0	27-12-2015	1.33	64236.62	1036.74	54454.85	48.16	8696.87	8603.62	93.25	0.0	conventional	2015.0	Albany
1	1.0	20-12-2015	1.35	54876.98	674.28	44638.81	58.33	9505.56	9408.07	97.49	0.0	conventional	2015.0	Albany
2	2.0	13-12-2015	0.93	118220.22	794.70	109149.67	130.50	8145.35	8042.21	103.14	0.0	conventional	2015.0	Albany
3	3.0	06-12-2015	1.08	78992.15	1132.00	71976.41	72.58	5811.16	5677.40	133.76	0.0	conventional	2015.0	Albany
4	4.0	29-11-2015	1.28	51039.60	941.48	43838.39	75.78	6183.95	5986.26	197.69	0.0	conventional	2015.0	Albany

Feature Information of the DataSet

In [48]:
```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16468 entries, 0 to 16467
Data columns (total 14 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Unnamed: 0    1517 non-null   float64
 1   Date          1517 non-null   object
 2   AveragePrice  1517 non-null   float64
 3   Total Volume  1517 non-null   float64
 4   4046          1517 non-null   float64
 5   4225          1517 non-null   float64
 6   4770          1517 non-null   float64
 7   Total Bags    1517 non-null   float64
 8   Small Bags    1517 non-null   float64
 9   Large Bags    1517 non-null   float64
 10  XLarge Bags   1517 non-null   float64
 11  type          1517 non-null   object
 12  year          1517 non-null   float64
 13  region        1517 non-null   object
dtypes: float64(11), object(3)
memory usage: 1.8+ MB
```

According to the Infomation:

1)No-Null data 2)1 - Object Type 3)7 - Float Type 4)1 - Int Type

Feature Distirbution of data for Float and Int Data Type

In [86]:
```python
df.describe()
```

Out[86]:

	Unnamed: 0	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	year
count	1517.000000	1517.000000	1.517000e+03	1.517000e+03	1.517000e+03	1.517000e+03	1.517000e+03	1.517000e+03	1.517000e+03	1517.000000	1517.000000
mean	26.995386	1.074990	1.601879e+06	6.464387e+05	6.114375e+05	5.040550e+04	2.935974e+05	2.487736e+05	4.264205e+04	2181.771074	2015.162821
std	14.848287	0.188891	4.433143e+06	1.947614e+06	1.672906e+06	1.377812e+05	7.579765e+05	6.474765e+05	1.182157e+05	7455.712144	0.369324
min	0.000000	0.490000	3.875074e+04	4.677200e+02	1.783770e+03	0.000000e+00	3.311770e+03	3.311770e+03	0.000000e+00	0.000000	2015.000000
25%	14.000000	0.980000	1.474700e+05	2.040034e+04	4.147606e+04	9.112500e+02	3.620689e+04	2.972722e+04	5.407400e+02	0.000000	2015.000000
50%	29.000000	1.080000	4.027919e+05	8.175117e+04	1.186649e+05	7.688170e+03	7.397906e+04	6.237569e+04	5.044350e+03	0.000000	2015.000000
75%	39.000000	1.190000	9.819751e+05	3.775785e+05	4.851503e+05	2.916730e+04	1.576097e+05	1.461994e+05	2.926767e+04	401.480000	2015.000000
max	51.000000	1.680000	4.465546e+07	1.893304e+07	1.895648e+07	1.381516e+06	6.736304e+06	5.893642e+06	1.121076e+06	108072.790000	2016.000000

Above statistics data show that their multiple outliers mostly in XLargeBags There is also difference between mean and 50% value in some of the columns which used to get fix for better prediction

In [52]:
```python
df.type.unique()
```

Out[52]:
```
array(['conventional', nan], dtype=object)
```

Exploratory Data Analysis

In [54]:
```python
df.year.unique()
```

Out[54]:
```
array([2015., 2016.,   nan])
```

Type of Avocado vs Average Price

Predicting Average Price of Avocado

In [19]:
```python
columns_to_drop = ['Unnamed: 0', '4046', '4225', '4770', 'Total Bags', 'Small Bags', 'Large Bags', 'XLarge Bags', 'type']
avo_df = df.drop(columns_to_drop, axis=1)
display(avo_df.head())
```

	Date	AveragePrice	Total Volume	year	region
0	27-12-2015	1.33	64236.62	2015.0	Albany
1	20-12-2015	1.35	54876.98	2015.0	Albany
2	13-12-2015	0.93	118220.22	2015.0	Albany
3	06-12-2015	1.08	78992.15	2015.0	Albany
4	29-11-2015	1.28	51039.60	2015.0	Albany

In [31]:
```python
regions = avo_df.region.unique()

print(regions)
```

```
['Albany' 'Atlanta' 'BaltimoreWashington' 'Boise' 'Boston'
 'BuffaloRochester' 'California' 'Charlotte' 'Chicago' 'Columbus'
 'DallasFtWorth' 'Denver' 'Detroit' 'GrandRapids' 'GreatLakes'
 'HarrisburgScranton' 'HartfordSpringfield' 'Houston' 'Indianapolis'
 'Jacksonville' 'LasVegas' 'LosAngeles' 'Louisville' 'MiamiFtLauderdale'
 'Midsouth' 'Nashville' 'NewYork' 'Northeast' 'NorthernNewEngland'
 'Orlando' 'Philadelphia' 'PhoenixTucson' 'Pittsburgh' 'Plains' 'Portland'
 'RaleighGreensboro' 'RichmondNorfolk' 'Roanoke' 'SanDiego' 'SanFrancisco'
 'Seattle' 'SouthCarolina' 'SouthCentral' 'Southeast' 'Spokane' 'StLouis'
 'Syracuse' 'Tampa' 'TotalUS' 'West' 'WestTexNewMexico' nan]
```

Statistical EDA

Now would be a good time to carry out a little statistical EDA, to get an initial idea of the shape of our data. Let's take a look at the maximum, mean, median, minimum and standard deviation values, just to get a flavour of the spread of prices in our data.

In [32]:
```python
print('Maximum = ' + str(avo_df.AveragePrice.max()))
print('Mean = ' + str(avo_df.AveragePrice.mean()))
print('Median = ' + str(avo_df.AveragePrice.median()))
print('Minimum = ' + str(avo_df.AveragePrice.min()))
print('Standard Deviation = ' +str(avo_df.AveragePrice.std()))
```

```
Maximum = 1.68
Mean = 1.0749901120632825
Median = 1.08
Minimum = 0.49
Standard Deviation = 0.18889123235190147
```

Just for interest, and to get to know our dataset better, let's find out the regions with the cheapest and most expensive avocados, according to our dataset.

In [33]:
```python
display(avo_df[avo_df.AveragePrice == avo_df.AveragePrice.min()])
display(avo_df[avo_df.AveragePrice == avo_df.AveragePrice.max()])
```

	Date	AveragePrice	Total Volume	year	region
760	27-12-2015	0.49	1137707.43	2015.0	PhoenixTucson

	Date	AveragePrice	Total Volume	year	region
1457	06-11-2016	1.68	3395058.42	2016.0	California
1458	30-10-2016	1.68	3139833.50	2016.0	California

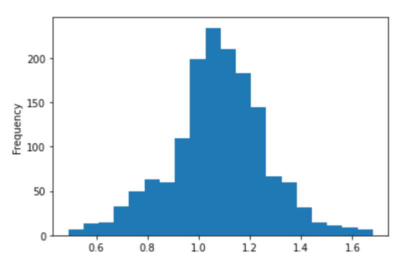
Visual EDA

Now we can move on to performing some visual EDA. This is always an important step in getting a feel for a dataset.

First of all, let's take a look at the distribution of prices using a histogram

In [34]:
```python
avo_df['AveragePrice'].plot(kind='hist', bins=20)
```

Out[34]:
```
<AxesSubplot:ylabel='Frequency'>
```



In [35]:
```python
avo_df['region'].value_counts().head()
```

Out[35]:
```
California             76
Albany                 67
BaltimoreWashington    65
Boise                  65
Boston                 62
Name: region, dtype: int64
```

Create Average Price DataFrame

Create New DataFrame

In [36]:
```python
group_by_region = avo_df.groupby(by=['region'])
avo_df_avg = group_by_region.mean()
avo_df_avg = avo_df_avg.drop(['year'], axis=1)
display(avo_df_avg.head())
```

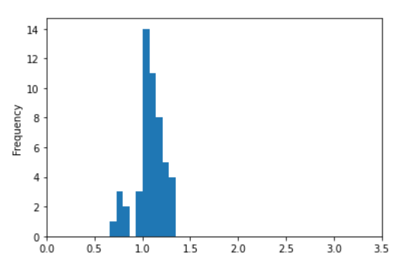
	AveragePrice	Total Volume
region		
Albany	1.238657	76290.195373
Atlanta	1.012037	467637.160926
BaltimoreWashington	1.160923	807644.197385
Boise	0.974923	81046.168769
Boston	1.205484	553458.590000

Visual EDA on the new DataFrame

Let's take a look at the distribution of prices for our new smaller dataset to see if the distribution looks roughly the same

In [37]:
```python
avo_df_avg['AveragePrice'].plot(kind='hist', xlim=(0,3.5), bins=10)
```

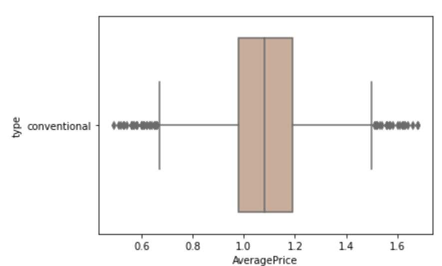
Out[37]:
```
<AxesSubplot:ylabel='Frequency'>
```



Type of Avocado vs Average Price

In [65]:
```python
import seaborn as sns
sns.boxplot(y="type", x="AveragePrice", data=df, palette = 'pink')
```

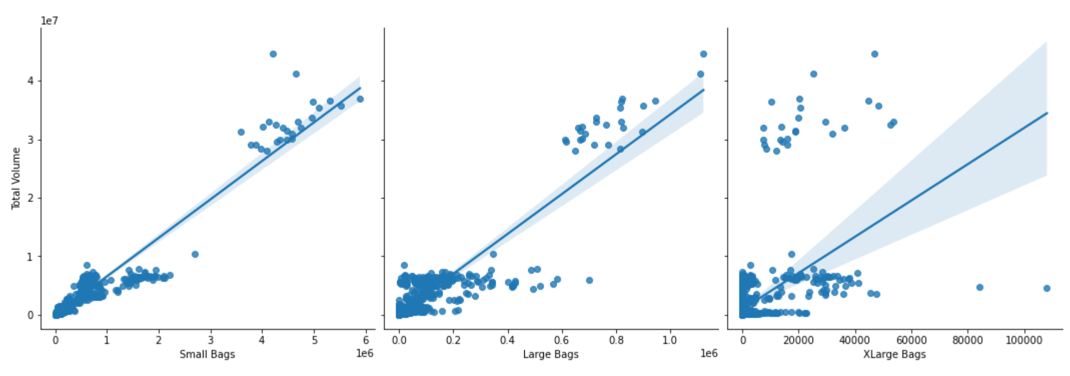
Out[65]:
```
<AxesSubplot:xlabel='AveragePrice', ylabel='type'>
```



Total Volume vs Small, Large and XLarge

In [76]:
```python
import warnings
warnings.filterwarnings('ignore')
import seaborn as sns
sns.pairplot(df, x_vars=['Small Bags', 'Large Bags', 'XLarge Bags'], y_vars='Total Volume', size=5, aspect=1, kind='reg')
```

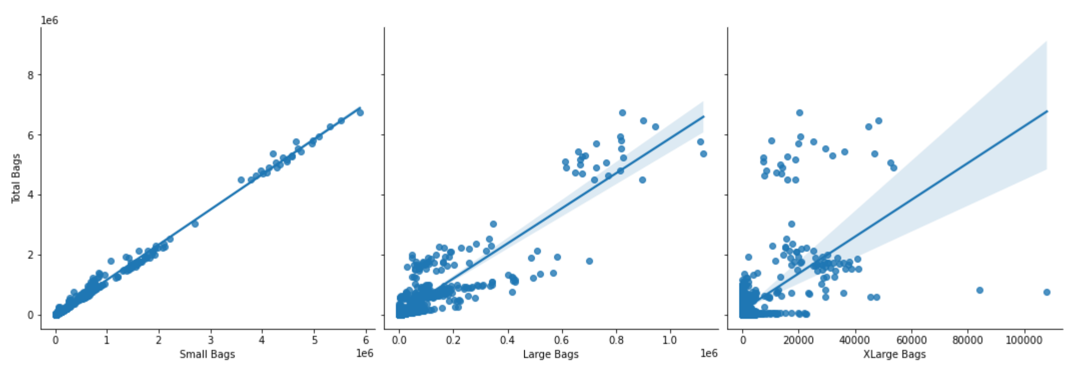
Out[76]:
```
<seaborn.axisgrid.PairGrid at 0x174bbe18a30>
```



Total Bags vs Small Bags, Large Bags and XLarge Bags

In [83]:
```python
import warnings
warnings.filterwarnings('ignore')
import seaborn as sns
sns.pairplot(df, x_vars=['Small Bags', 'Large Bags', 'XLarge Bags'], y_vars='Total Bags', size=5, aspect=1, kind='reg')
```

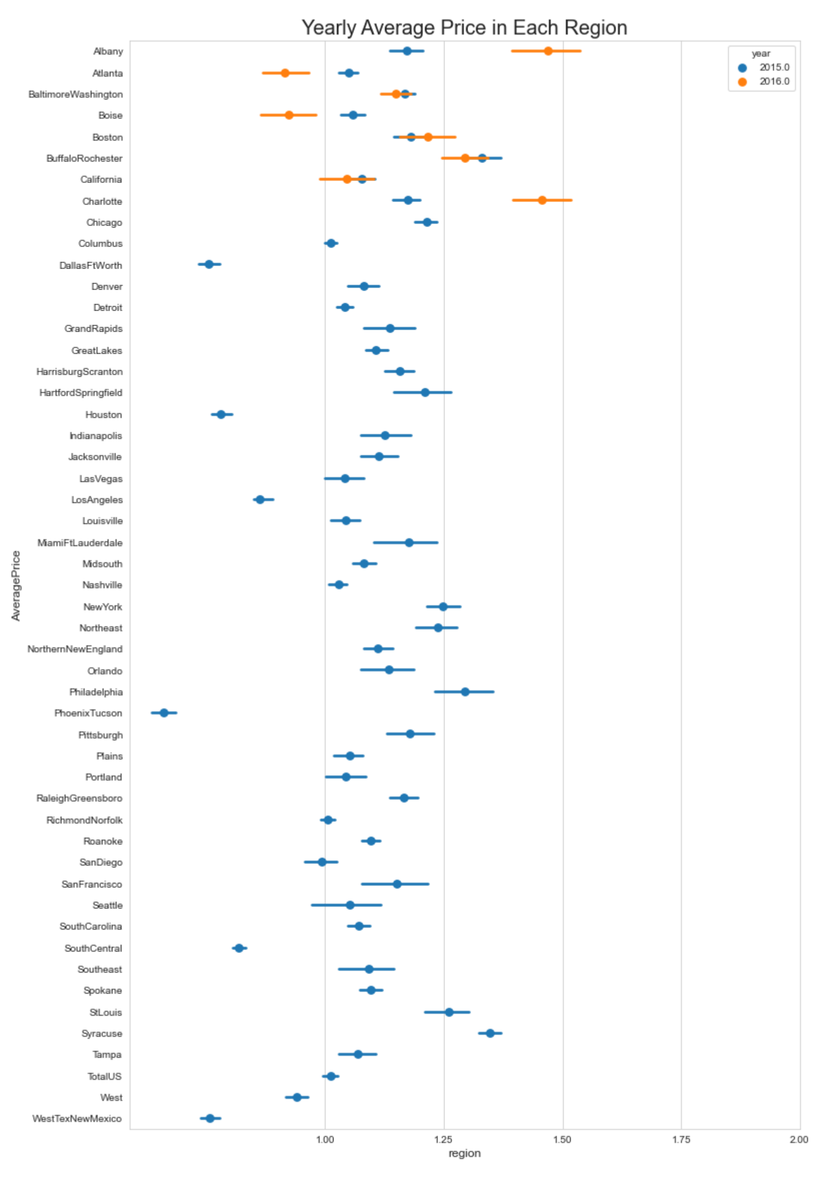
Out[83]:
```
<seaborn.axisgrid.PairGrid at 0x174bc700850>
```



Region Vs Year distribution

In [85]:
```python
plt.figure(figsize=(12,20))
sns.set_style('whitegrid')
sns.pointplot(x='AveragePrice',y='region',data=df, hue='year',join=False)
plt.xticks(np.linspace(1,2,5))
plt.xlabel('region',{'fontsize':'large'})
plt.ylabel('AveragePrice',{'fontsize':'large'})
plt.title("Yearly Average Price in Each Region",{'fontsize':20})
```

Out[85]:
```
Text(0.5, 1.0, 'Yearly Average Price in Each Region')
```



Conclusion

With the help of notebook I learnt how EDA can be carried out using Pandas and other plotting libraries.

Also I have seen making use of packages like matplotlib, plotly and seaborn to develop better insights about the data. I have also seen how preproceesing helps in dealing with missing values and irregualities present in the data. I also learnt how to create new features which will in turn help us to better predict the survival. I also make use of pandas profiling feature to generate an html report containing all the information of the various features present in the dataset. I have seen the impact of columns like type, year/date on the Average price increase/decrease rate. The most important inference drawn from all this analysis is, I get to know what are the features on which price is highly positively and negatively coorelated with.

THANK YOU