Avocado Price Analysis

February 2, 2021

[82]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from scipy import stats
from scipy.stats import linregress
```

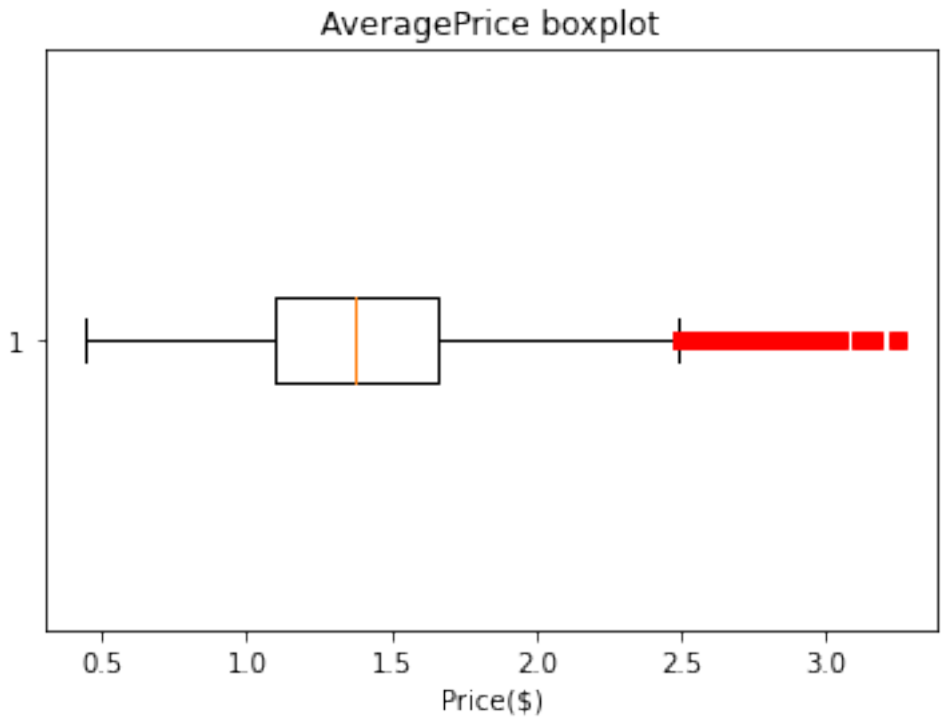
[2]:
```
data = pd.read_csv('avocado.csv',delimiter = ',')
data.drop(columns='Unnamed: 0',inplace=True)
```

[3]:
```
data.info()
```

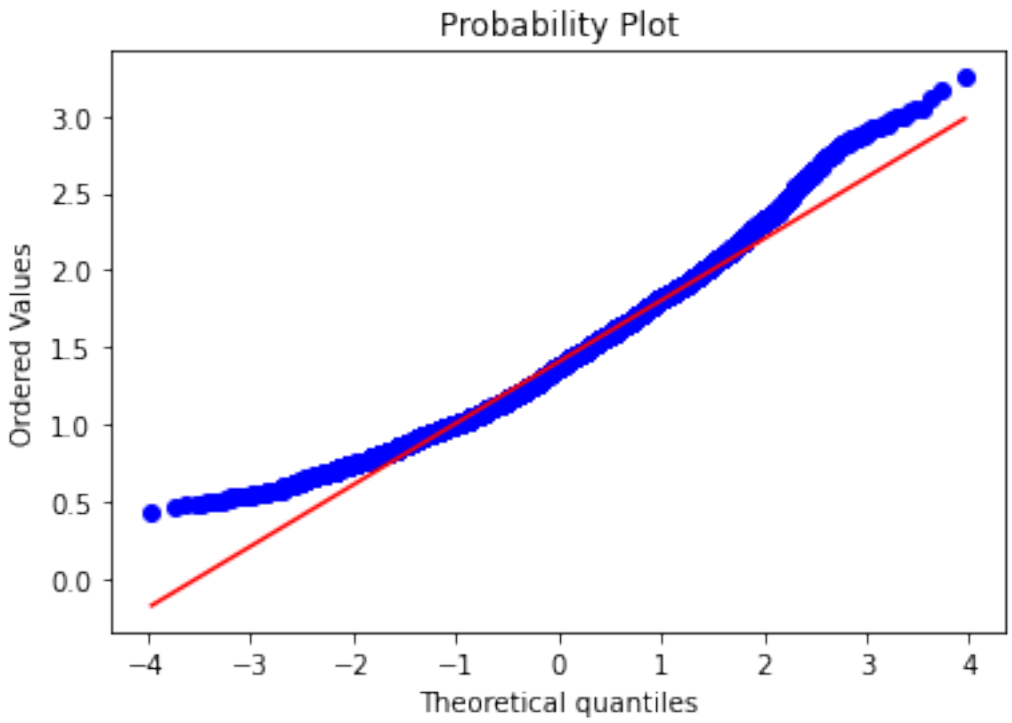
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18249 entries, 0 to 18248
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Date          18249 non-null  object 
 1   AveragePrice  18249 non-null  float64
 2   Total Volume  18249 non-null  float64
 3   4046          18249 non-null  float64
 4   4225          18249 non-null  float64
 5   4770          18249 non-null  float64
 6   Total Bags    18249 non-null  float64
 7   Small Bags    18249 non-null  float64
 8   Large Bags    18249 non-null  float64
 9   XLarge Bags   18249 non-null  float64
 10  type          18249 non-null  object 
 11  year          18249 non-null  int64  
 12  region        18249 non-null  object 
dtypes: float64(9), int64(1), object(3)
memory usage: 1.8+ MB
```

[4]:
```
plt.boxplot(data['AveragePrice'], 0, 'rs', 0)
plt.title('AveragePrice boxplot')
plt.xlabel('Price($)')
plt.plot()
```

[4]: []



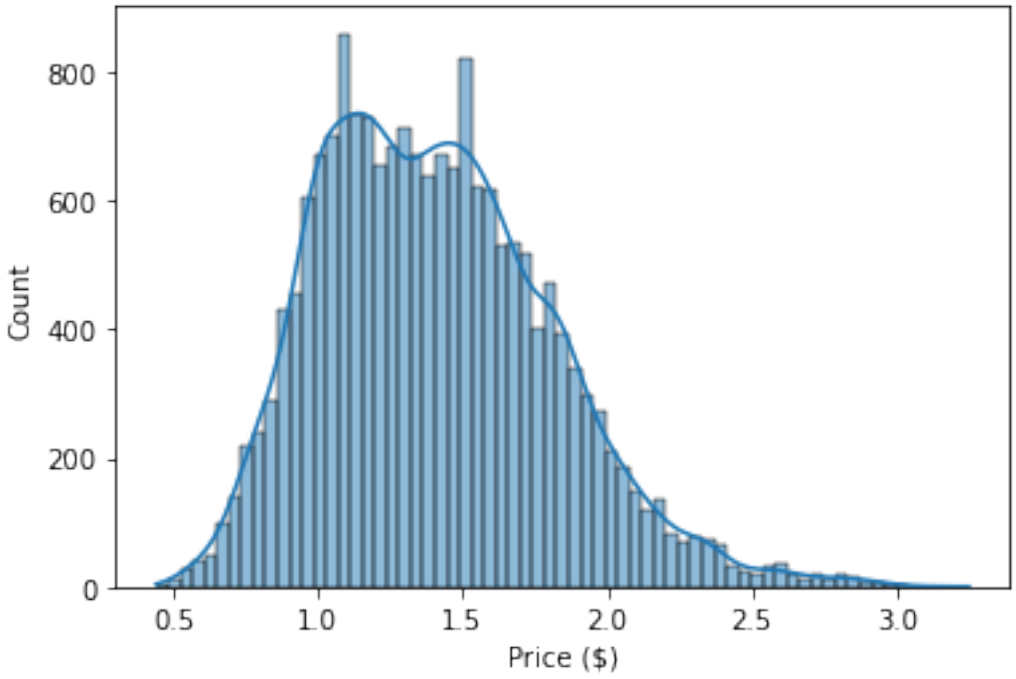
[5]:
```
ax4 = plt.subplot()
res = stats.probplot(data['AveragePrice'],dist ='norm',plot=plt)
```



Not a great fit for a normal distriibution.

[6]:
```
sns.histplot(data['AveragePrice'],kde=True)
plt.xlabel('Price ($)')
plt.plot()
```

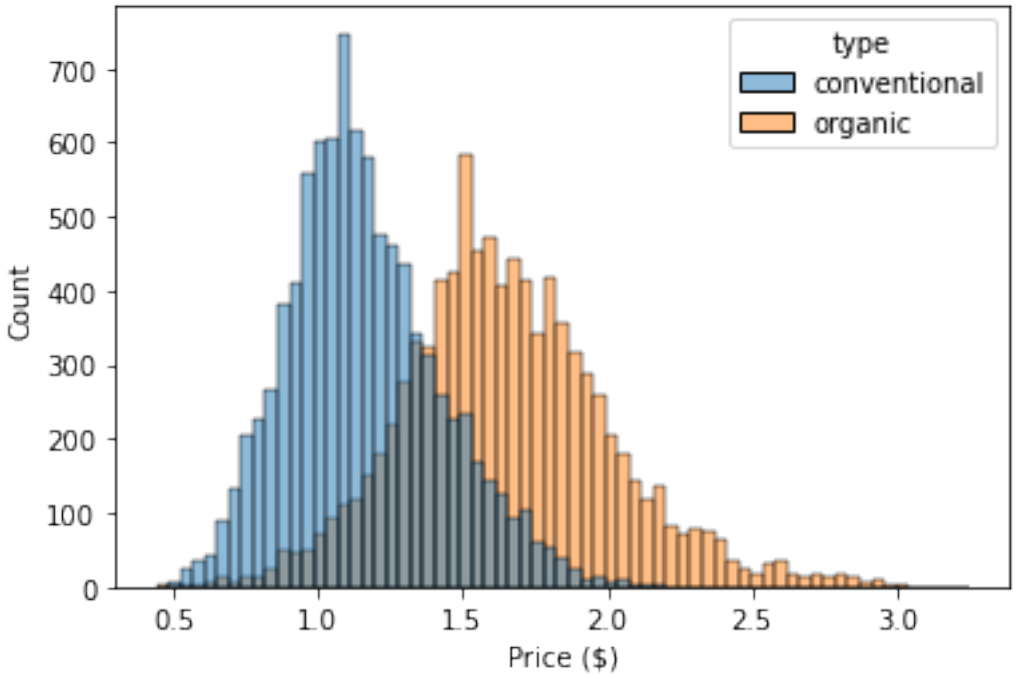
[6]: []



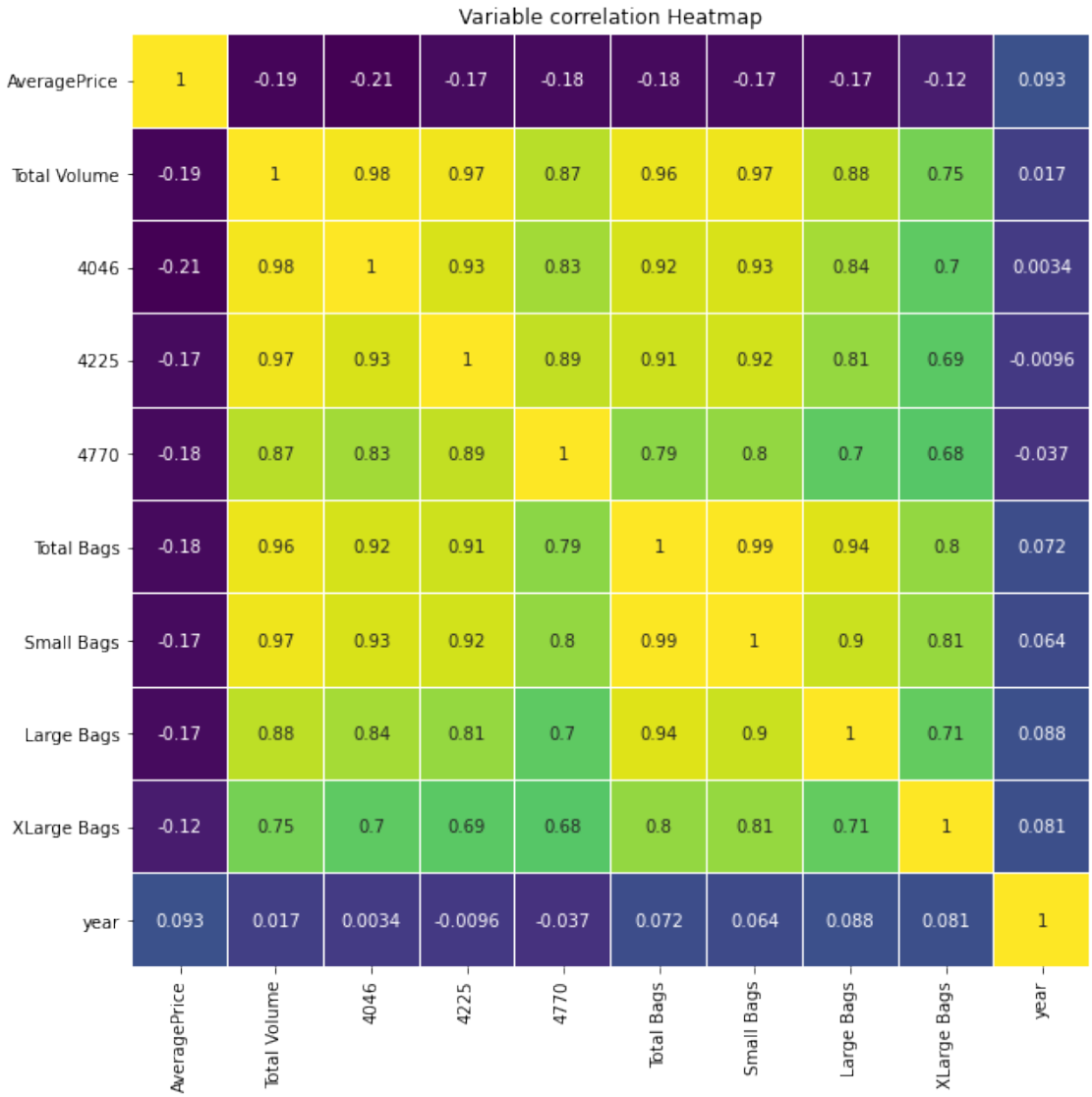
It appears to be a bimodal distribution, which is strange for a price of the same item you would expect it to be unimodal.

later in the analysis, I come to compare the variable 'type' which categorizes the avocados between organic and conventional. this explains the bimodal distribution, as the skews for organic and conv are different

[7]:
```
sns.histplot(data,x='AveragePrice',hue='type')
plt.xlabel('Price ($)')
plt.show()
```



[8]:
```
d_corr = data.corr()
fig,ax = plt.subplots(figsize=(10,10))
ax = sns.heatmap(d_corr,annot=True,linewidth=0.01,cbar=False,cmap='viridis')
ax.set_title('Variable correlation Heatmap')
plt.show()
```

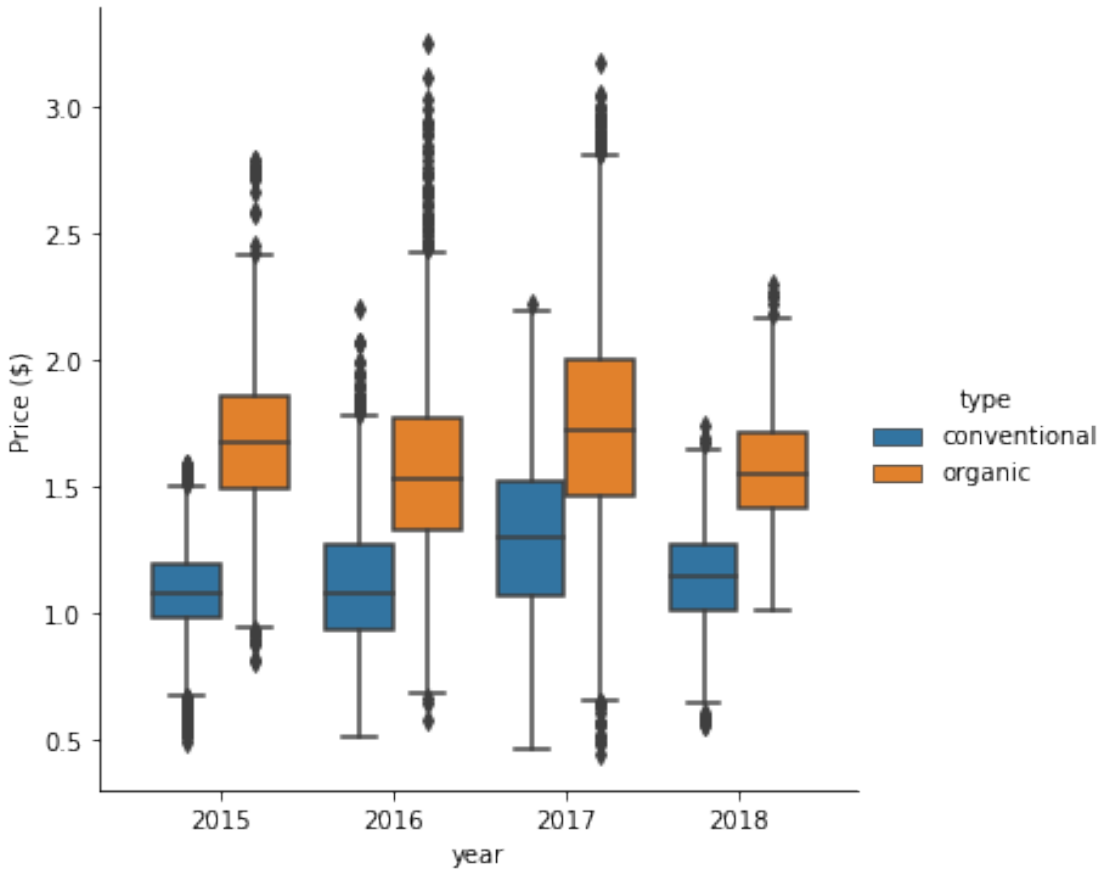


0.1 Categorical

0.1.1 Type organic / conventional

[9]:
```
sns.catplot(x ='year',y='AveragePrice', data=data,hue='type',kind='box')
plt.ylabel('Price ($)')
plt.plot()
```

[9]: []

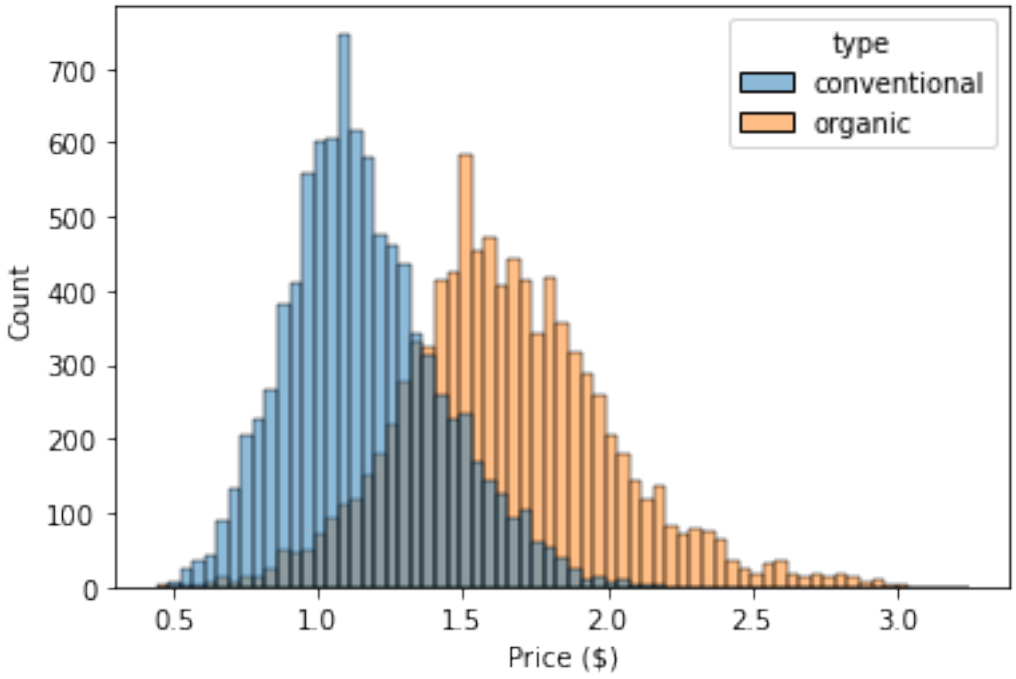


- Organic avocados tend to be more expensive that conventional avocados.

bernulli 1-organic, 0-conventional

[10]:
```
sns.histplot(data,x='AveragePrice',hue='type')
plt.xlabel('Price ($)')
plt.plot()
```

[10]: []

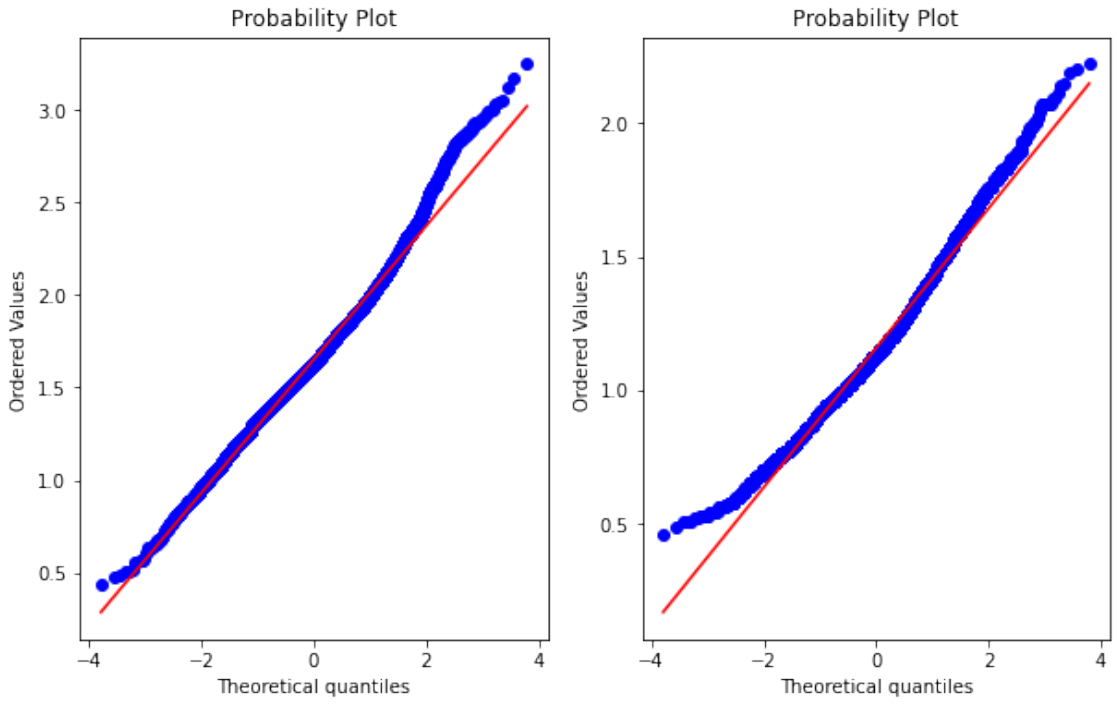


distribution fit

[11]:
```
f, ax = plt.subplots(1,2,figsize=(10,6))

stats.probplot(data.loc[data['type']=='organic','AveragePrice'],dist␣
↪='norm',plot=ax[0])
stats.probplot(data.loc[data['type']=='conventional','AveragePrice'],dist␣
↪='norm',plot=ax[1])
plt.plot()
```

[11]: []



the spread of values for organic is much larger. 4Q >3.0 compared to 4Q<3.0

[12]:
```
avo_type  = data.groupby('type').agg(['mean','std','count'])
avo_type = round(avo_type,2)
#mean diff
mean_diff =␣
 ↪avo_type['AveragePrice']['mean']['organic']-avo_type['AveragePrice']['mean']['conventional']
#standard deviation
organic_s1 = avo_type['AveragePrice']['std']['organic']
conventional_s2 = avo_type['AveragePrice']['std']['conventional']
print('mean difference',mean_diff,'p')
```

```
mean difference 0.49 p
```

$X_1 =$ Organic Avocados, AveragePrice dist.

$X_2 =$ Conventional Avocados, AveragePrice dist.

$\bar{X_1} - \bar{X_2} = 0.49$

$\alpha = 0.01$

$$\sigma^2_{\bar{X_1}+\bar{X_2}} \approx \frac{S_1}{n_1} + \frac{S_2}{n_2}$$

[13]:
```
sampling_stderr = (organic_s1/
 ↪avo_type['AveragePrice']['count']['organic'])+(conventional_s2/
 ↪avo_type['AveragePrice']['count']['conventional'])
sampling_stderr = np.sqrt(sampling_stderr)
print('sampling std error approx',sampling_stderr)

crit_limit =2.33
limit = crit_limit*sampling_stderr
print(round(mean_diff - limit,2),'to',round(mean_diff +limit,2))
```

```
sampling std error approx 0.008243223411136655
0.47 to 0.51
```

Confident (That the true Mean difference in price between Organic avocados and conventional avocados is between 0.47p and 0.51p) $\approx 99\%$

0.2 Region

Groupby region, with lambda agg to reduce oulier prices.

[14]:
```
region_data = data.groupby(['region','type'])['AveragePrice'].agg(['mean',
                                                    lambda x : np.
 ↪quantile(x,.15),
                                                    lambda x : np.
 ↪quantile(x,.85)])
region_data = region_data.unstack()
region_data.rename(columns={'<lambda_0>':'q15','<lambda_1>':'q85'},inplace=True)
err_pdata = region_data.copy()
region_data.head()
```

[14]:

	mean		q15		q85	\
type	conventional	organic	conventional	organic	conventional	
region						
Albany	1.348757	1.773314	1.110	1.54	1.588	
Atlanta	1.068817	1.607101	0.902	1.25	1.230	
BaltimoreWashington	1.344201	1.724260	1.120	1.51	1.600	
Boise	1.076036	1.620237	0.836	1.16	1.268	
Boston	1.304379	1.757396	1.070	1.49	1.580	

type	organic
region	
Albany	1.970
Atlanta	1.906
BaltimoreWashington	2.028
Boise	2.088
Boston	2.010

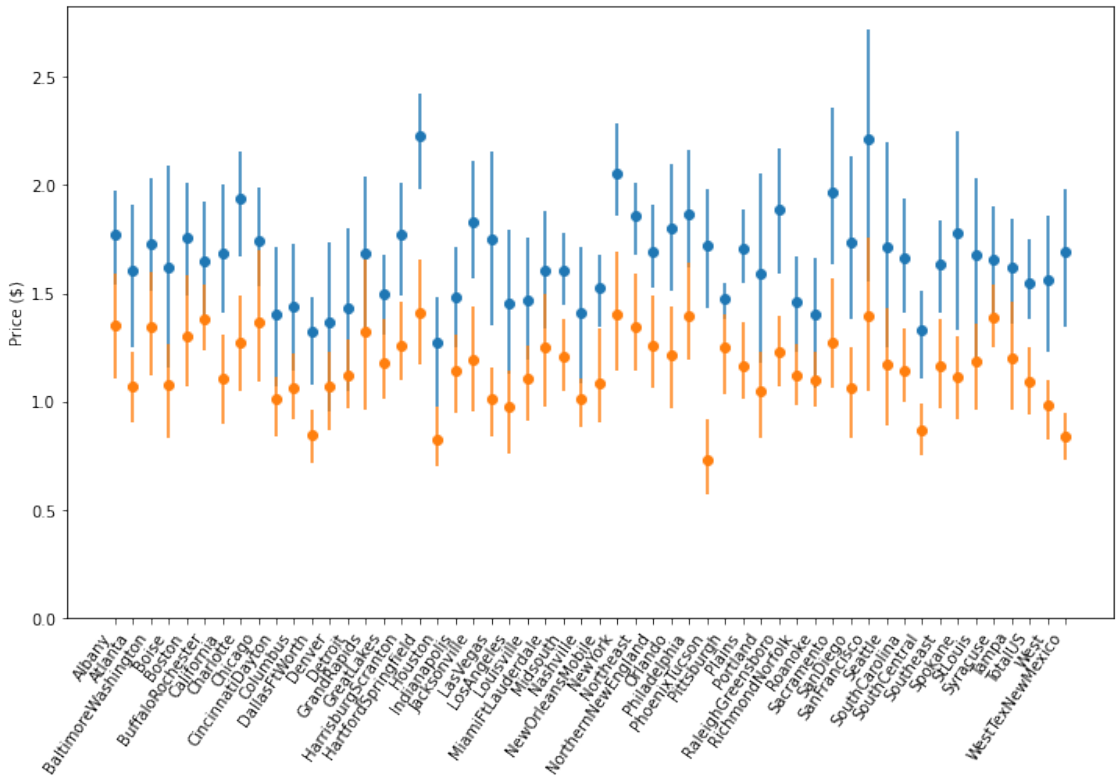
[15]:
```
#reformating the quartile columns for the error bar plot
err_pdata['q15'] = err_pdata['mean'] - err_pdata['q15']
err_pdata['q85']= err_pdata['q85'] - err_pdata['mean']
err_pdata.head()
```

[15]:

	mean		q15		\
type	conventional	organic	conventional	organic	
region					
Albany	1.348757	1.773314	0.238757	0.233314	
Atlanta	1.068817	1.607101	0.166817	0.357101	
BaltimoreWashington	1.344201	1.724260	0.224201	0.214260	
Boise	1.076036	1.620237	0.240036	0.460237	
Boston	1.304379	1.757396	0.234379	0.267396	

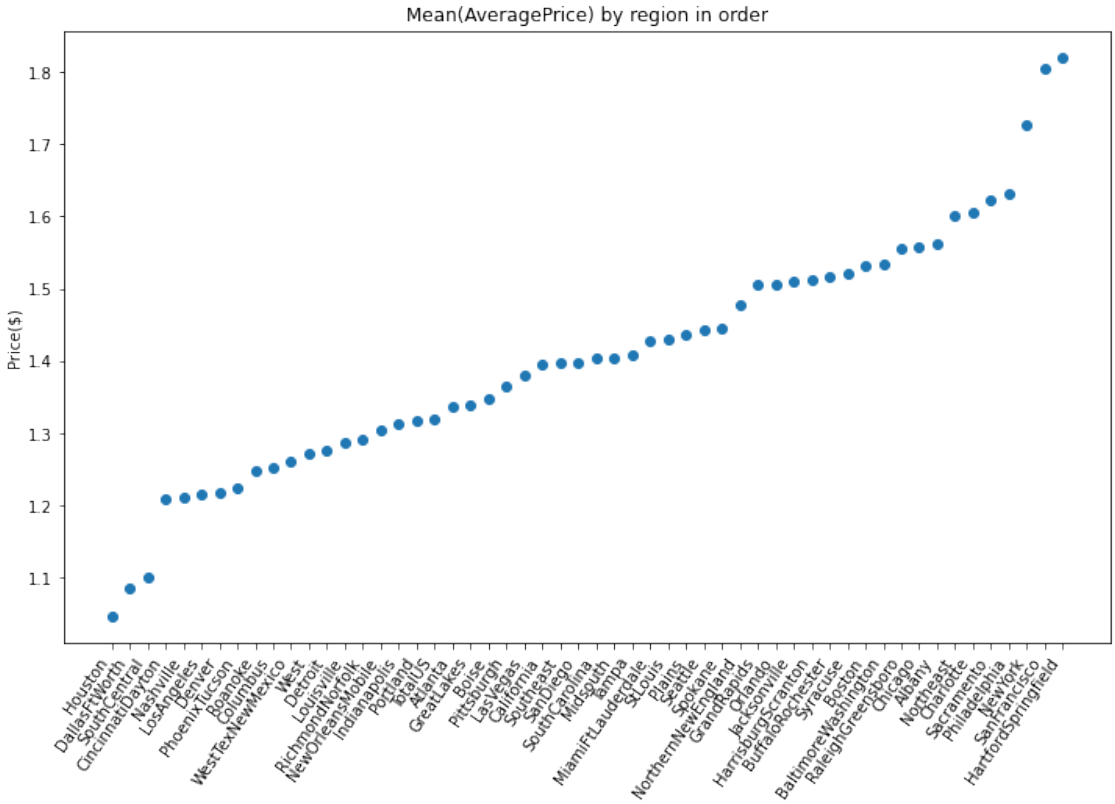
	q85	
type	conventional	organic
region		
Albany	0.239243	0.196686
Atlanta	0.161183	0.298899
BaltimoreWashington	0.255799	0.303740
Boise	0.191964	0.467763
Boston	0.275621	0.252604

[240]:
```
fig,ax = plt.subplots(figsize=(12,8))
fig.autofmt_xdate(rotation=55 )
ax.errorbar(x = err_pdata.
 ↪index,y=err_pdata['mean']['organic'],yerr=[err_pdata['q15']['organic'],err_pdata['q85']['or
 ↪='o')
ax.errorbar(x = err_pdata.
 ↪index,y=err_pdata['mean']['conventional'],yerr=[err_pdata['q15']['conventional'],err_pdata[
 ↪='o')
ax.set_yticks([0,0.5,1,1.5,2,2.5])
ax.set_ylabel('Price ($)')
plt.show()
```



[232]:
```
price_by_region = data.groupby('region')['AveragePrice'].agg(['mean'])
price_by_region = price_by_region.sort_values('mean')
price_by_region = price_by_region.reset_index()
fig,ax = plt.subplots(figsize = (12,8))
fig.autofmt_xdate(rotation=55 )
ax.scatter(x=price_by_region['region'],y=price_by_region['mean'])
ax.set_title('Mean(AveragePrice) by region in order')
ax.set_ylabel('Price($)')
plt.plot()
```

[232]: []



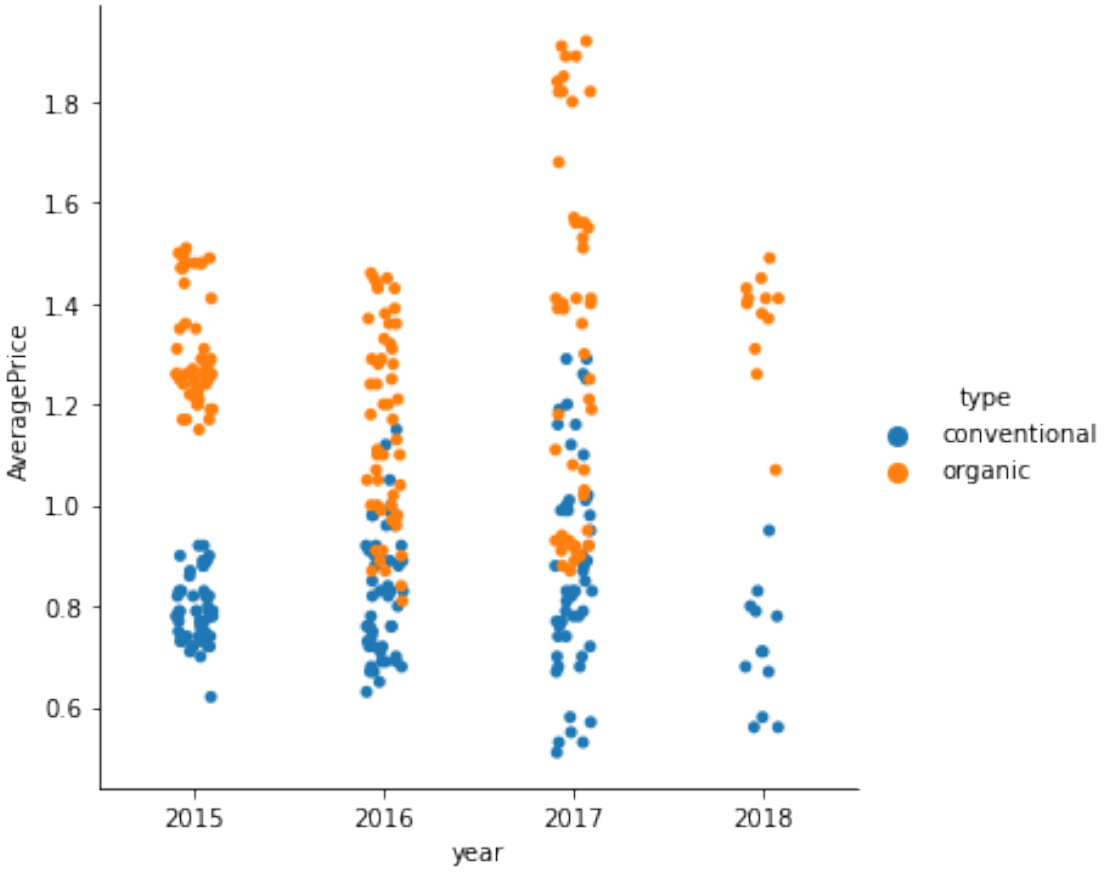
- Houston on average has the cheapest avocados
- Houston also has on average the cheapest organic avocados
- pitsburg on average has the cheapest Conventional avocados

Even though i think i will be a terrible fit, i want to fit a linear reggresion line to Price over year.

[241]:
```
houston = data[data['region']=='Houston']
h_organic = houston[houston['type']=='organic']
h_conv = houston[houston['type']=='conventional']
```

[242]:
```
sns.catplot(data = houston,x='year',y='AveragePrice',hue='type')
```

[242]:
```
<seaborn.axisgrid.FacetGrid at 0x28edce48cc8>
```

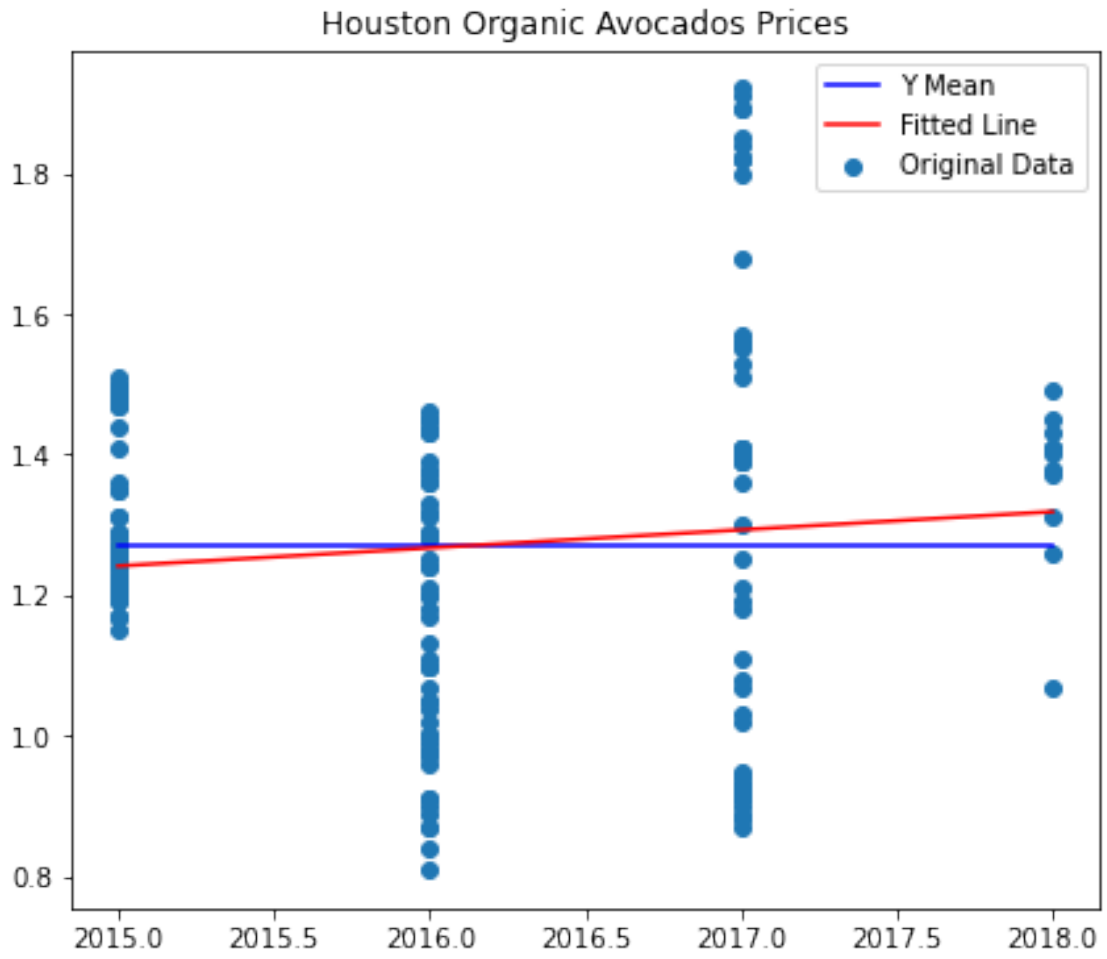


0.2.1 Organic

[248]:
```
slope_org, intercept_org, r_org, p_org, se_org =␣
↪linregress(h_organic['year'],h_organic['AveragePrice'])
```

[249]:
```
fig = plt.figure(figsize=(7,6))
plt.scatter(h_organic['year'],h_organic['AveragePrice'],label='Original Data')
plt.plot(h_organic['year'],h_organic['year'].apply(lambda val :
↪h_organic['AveragePrice'].mean()),label = 'Y Mean',c='b')
plt.plot(h_organic['year'],h_organic['year'].apply(lambda val : val*slope_org +␣
↪intercept_org),label='Fitted Line',c='r')
plt.title('Houston Organic Avocados Prices')
plt.legend()
plt.plot()
```

[249]: []



coefficient of determination

[250]:
```
h_organic = h_organic.copy()
h_organic.loc[:,'y'] = h_organic.loc[:,'AveragePrice']
h_organic.loc[:,'mx+b'] = h_organic.loc[:,'year'].apply(lambda val :␣
 ↪val*slope_org + intercept_org)
h_organic.loc[:,'SE_line']  = (h_organic.loc[:,'y'] - h_organic.loc[:
 ↪,'mx+b'])**2
```

[251]:
```
one_minusr2 = round((1- h_organic.loc[:,'SE_line'].sum()/((h_organic.loc[:
 ↪,'y']-h_organic.loc[:,'y'].mean())**2).sum()),2)
print(one_minusr2*100,'% of the total variation is described by the regression␣
 ↪line y=',round(slope_org,2),'x',round(intercept_org,2))
```

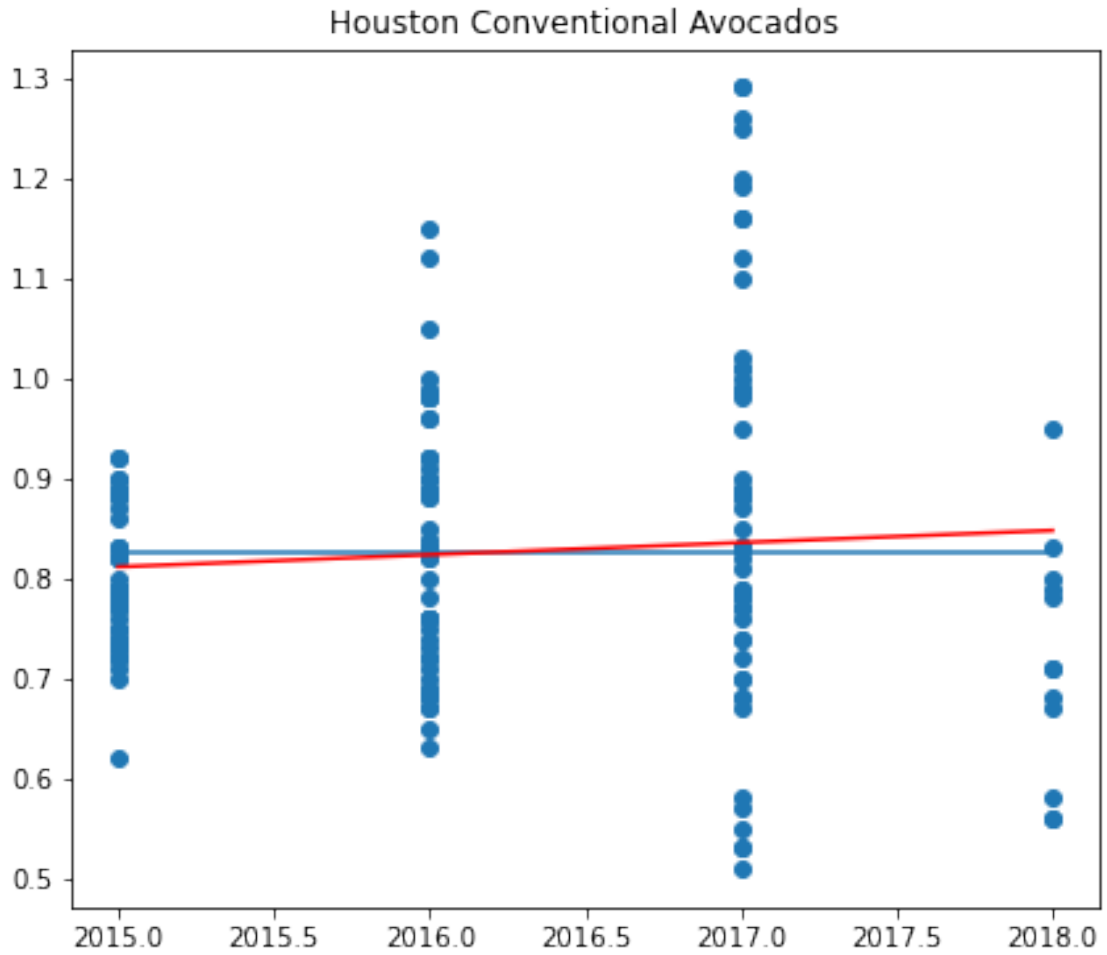
```
1.0 % of the total variation is described by the regression line y= 0.03 x
-50.59
```

0.2.2 Conventional

[252]:
```
slope_con, intercept_con, r, p, se =␣
↪linregress(h_conv['year'],h_conv['AveragePrice'])
```

[253]:
```
fig = plt.figure(figsize=(7,6))
plt.scatter(h_conv['year'],h_conv['AveragePrice'])
plt.plot(h_conv['year'],h_conv['year'].apply(lambda val :h_conv['AveragePrice'].
↪mean()))
plt.plot(h_conv['year'],h_conv['year'].apply(lambda val : val*slope_con +␣
↪intercept_con),c='r')
plt.title('Houston Conventional Avocados')
plt.plot()
```

[253]: []



coefficient of determination

[254]:
```
h_conv = h_conv.copy()
h_conv.loc[:,'y'] = h_conv.loc[:,'AveragePrice']
h_conv.loc[:,'mx+b'] = h_conv.loc[:,'year'].apply(lambda val : val*slope_con +␣
 ↪intercept_con)
h_conv.loc[:,'SE_line']  = (h_conv.loc[:,'y'] - h_conv.loc[:,'mx+b'])**2
```

[255]:
```
one_minusr2_con = round((1- h_conv.loc[:,'SE_line'].sum()/((h_conv.loc[:
 ↪,'y']-h_conv.loc[:,'y'].mean())**2).sum()),2)
print(one_minusr2_con*100,'% of the total variation is described by the␣
 ↪regression line y=',round(slope_con,2),'x',round(intercept_con,2))
```

```
1.0 % of the total variation is described by the regression line y= 0.01 x
-23.65
```

- Not surprisingly simple linear regression, least squared line has not produced a reliable predictor, non the the less i wanted to put it into practive and visualize it.