In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import random
```

In [ ]:
```
```

Question - 1

Suppose that the data for analysis includes the attribute age. The age values for the data tuples are (in increasing order) 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70.

(a) Use min-max normalization to transform the values of age to the range[0:1].

(b) Use z-score normalization to transform the values of age.

(c) Use normalization by decimal scaling to transform the values of age such that the transformed value is less than 1.

In [2]:
```
age = (13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70)

age = np.array(age)
mean = np.mean(age)
std = np.std(age)

min_val = age[0]
max_val = age[len(age)-1]
min_max_normal = []
z_score_normal = []
decimal_scale_normal = []

for i in range(len(age)):
    min_max_normal.append((age[i]-min_val)/(max_val-min_val))
    z_score_normal.append( (age[i]-mean)/std )
    decimal_scale_normal.append( age[i]/10**len(str(max_val)) )


print(min_max_normal)
print()
print(z_score_normal)
print()
print(decimal_scale_normal)
```

```
[0.0, 0.03508771929824561, 0.05263157894736842, 0.05263157894736842, 0.10526315789473684, 0.12280701754385964, 0.12280701754385964, 0.140350877192982
45, 0.15789473684210525, 0.15789473684210525, 0.21052631578947367, 0.21052631578947367, 0.21052631578947367, 0.21052631578947367, 0.2982456140350877,
0.3508771929824561, 0.3508771929824561, 0.38596491228070173, 0.38596491228070173, 0.38596491228070173, 0.38596491228070173, 0.40350877192982454, 0.47
368421052631576, 0.5614035087719298, 0.5789473684210527, 0.6842105263157895, 1.0]

[-1.3356459850221374, -1.1781680741243308, -1.0994291186754275, -1.0994291186754275, -0.8632122523287176, -0.7844732968798143, -0.7844732968798143, -
0.705734341430911, -0.6269953859820077, -0.6269953859820077, -0.39077851963529775, -0.39077851963529775, -0.39077851963529775, -0.39077851963529775,
0.0029162576092187234, 0.23913312395592862, 0.23913312395592862, 0.3966110348537352, 0.3966110348537352, 0.3966110348537352, 0.3966110348537352, 0.47
53499903026385, 0.7903058120982517, 1.1840005893427683, 1.2627395447916716, 1.7351732774850914, 3.152474475565351]

[0.13, 0.15, 0.16, 0.16, 0.19, 0.2, 0.2, 0.21, 0.22, 0.22, 0.25, 0.25, 0.25, 0.25, 0.3, 0.33, 0.33, 0.35, 0.35, 0.35, 0.35, 0.36, 0.4, 0.45, 0.46, 0.
52, 0.7]
```

In [ ]:

Question - 2

Dataset description

In [3]:
```
df = pd.read_csv("Avocado.csv")
df
```

Out[3]:

	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region
0	27-12-2015	1.33	64236.62	1036.74	54454.85	48.16	8696.87	8603.62	93.25	0.0	conventional	2015	Albany
1	20-12-2015	1.35	54876.98	674.28	44638.81	58.33	9505.56	9408.07	97.49	0.0	conventional	2015	Albany
2	13-12-2015	0.93	118220.22	794.70	109149.67	130.50	8145.35	8042.21	103.14	0.0	conventional	2015	Albany
3	06-12-2015	1.08	78992.15	1132.00	71976.41	72.58	5811.16	5677.40	133.76	0.0	conventional	2015	Albany
4	29-11-2015	1.29	51039.60	941.48	43838.39	75.78	6183.95	5986.26	197.69	0.0	conventional	2015	Albany
...	...	...	...	...	...	...	...	...	...	...	...	...	...
18245	28-01-2018	1.71	13888.04	1191.70	3431.50	0.00	9264.84	8940.04	324.80	0.0	organic	2018	WestTexNewMexico
18246	21-01-2018	1.87	13766.76	1191.92	2452.79	727.94	9394.11	9351.80	42.31	0.0	organic	2018	WestTexNewMexico
18247	14-01-2018	1.93	16205.22	1527.63	2981.04	727.01	10969.54	10919.54	50.00	0.0	organic	2018	WestTexNewMexico
18248	07-01-2018	1.62	17489.58	2894.77	2356.13	224.53	12014.15	11988.14	26.01	0.0	organic	2018	WestTexNewMexico
18249	18-03-2018	1.56	15896.38	2055.35	1499.55	0.00	12341.48	12114.81	226.67	0.0	organic	2018	WestTexNewMexico

18250 rows × 13 columns

In [ ]:

In [4]:
```
sorted_bin_means_df = df.sort_values(by="Total Volume", ascending=True)
volume = sorted_bin_means_df['Total Volume']

sorted_bin_means_df
```

Out[4]:

	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region
10381	08-11-2015	1.59	84.56	3.95	3.95	0.00	76.66	73.33	3.33	0.00	organic	2015	MiamiFtLauderdale
9437	04-01-2015	1.73	379.82	0.00	59.82	0.00	320.00	320.00	0.00	0.00	organic	2015	BuffaloRochester
13189	30-10-2016	1.58	385.55	8.13	47.42	0.00	330.00	330.00	0.00	0.00	organic	2016	MiamiFtLauderdale
11698	12-07-2015	2.05	419.98	0.00	63.42	0.00	356.56	356.56	0.00	0.00	organic	2015	Syracuse
13193	02-10-2016	1.49	472.82	10.50	18.99	0.00	443.33	440.00	3.33	0.00	organic	2016	MiamiFtLauderdale
...	...	...	...	...	...	...	...	...	...	...	...	...	...
5493	08-05-2016	0.82	46324529.70	14223304.98	17896391.60	1993645.36	12211187.76	8747756.84	3342780.83	120650.09	conventional	2016	TotalUS
8353	07-05-2017	1.09	47293921.60	17076650.82	13549102.59	863471.88	15804696.31	11228049.63	4324231.19	252415.49	conventional	2017	TotalUS
5506	07-02-2016	0.76	52288697.89	16573573.78	20470572.61	2546439.11	12698112.39	9083373.04	3373077.87	241661.48	conventional	2016	TotalUS
8366	05-02-2017	0.77	61034457.10	22743616.17	20328161.55	1664383.09	16298296.29	12567155.58	3618270.75	112869.96	conventional	2017	TotalUS
9097	04-02-2018	0.87	62505646.52	21620180.90	20445501.03	1066830.22	19373134.37	13384586.80	5719096.61	269450.96	conventional	2018	TotalUS

18250 rows × 13 columns

Bin Mean

In [5]:
```
bin_size = volume.shape[0]//250
start = 0

for i in range(250):
    sorted_bin_means_df.iloc[start:start+bin_size, [2]] = [volume[start:start+bin_size].mean()] * bin_size
    start += bin_size


sorted_bin_means_df
```

Out[5]:

	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region
10381	08-11-2015	1.59	7.357130e+02	3.95	3.95	0.00	76.66	73.33	3.33	0.00	organic	2015	MiamiFtLauderdale
9437	04-01-2015	1.73	7.357130e+02	0.00	59.82	0.00	320.00	320.00	0.00	0.00	organic	2015	BuffaloRochester
13189	30-10-2016	1.58	7.357130e+02	8.13	47.42	0.00	330.00	330.00	0.00	0.00	organic	2016	MiamiFtLauderdale
11698	12-07-2015	2.05	7.357130e+02	0.00	63.42	0.00	356.56	356.56	0.00	0.00	organic	2015	Syracuse
13193	02-10-2016	1.49	7.357130e+02	10.50	18.99	0.00	443.33	440.00	3.33	0.00	organic	2016	MiamiFtLauderdale
...	...	...	...	...	...	...	...	...	...	...	...	...	...
5493	08-05-2016	0.82	3.897352e+07	14223304.98	17896391.60	1993645.36	12211187.76	8747756.84	3342780.83	120650.09	conventional	2016	TotalUS
8353	07-05-2017	1.09	3.897352e+07	17076650.82	13549102.59	863471.88	15804696.31	11228049.63	4324231.19	252415.49	conventional	2017	TotalUS
5506	07-02-2016	0.76	3.897352e+07	16573573.78	20470572.61	2546439.11	12698112.39	9083373.04	3373077.87	241661.48	conventional	2016	TotalUS
8366	05-02-2017	0.77	3.897352e+07	22743616.17	20328161.55	1664383.09	16298296.29	12567155.58	3618270.75	112869.96	conventional	2017	TotalUS
9097	04-02-2018	0.87	3.897352e+07	21620180.90	20445501.03	1066830.22	19373134.37	13384586.80	5719096.61	269450.96	conventional	2018	TotalUS

18250 rows × 13 columns

In [ ]:

Bin Median

In [6]:

```
sorted_bin_median_df = df.sort_values(by="Total Volume", ascending=True)
volume = df['Total Volume'].sort_values(ascending=True)

bin_size = volume.shape[0]//250
start = 0

for i in range(250):
    sorted_bin_median_df.iloc[start:start+bin_size, [2]] = [volume[start:start+bin_size].median()] * bin_size
    start += bin_size



sorted_bin_median_df
```

Out[6]:

	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region
10381	08-11-2015	1.59	774.20	3.95	3.95	0.00	76.66	73.33	3.33	0.00	organic	2015	MiamiFtLauderdale
9437	04-01-2015	1.73	774.20	0.00	59.82	0.00	320.00	320.00	0.00	0.00	organic	2015	BuffaloRochester
13189	30-10-2016	1.58	774.20	8.13	47.42	0.00	330.00	330.00	0.00	0.00	organic	2016	MiamiFtLauderdale
11698	12-07-2015	2.05	774.20	0.00	63.42	0.00	356.56	356.56	0.00	0.00	organic	2015	Syracuse
13193	02-10-2016	1.49	774.20	10.50	18.99	0.00	443.33	440.00	3.33	0.00	organic	2016	MiamiFtLauderdale
...	...	...	...	...	...	...	...	...	...	...	...	...	...
5493	08-05-2016	0.82	37352360.59	14223304.98	17896391.60	1993645.36	12211187.76	8747756.84	3342780.83	120650.09	conventional	2016	TotalUS
8353	07-05-2017	1.09	37352360.59	17076650.82	13549102.59	863471.88	15804696.31	11228049.63	4324231.19	252415.49	conventional	2017	TotalUS
5506	07-02-2016	0.76	37352360.59	16573573.78	20470572.61	2546439.11	12698112.39	9083373.04	3373077.87	241661.48	conventional	2016	TotalUS
8366	05-02-2017	0.77	37352360.59	22743616.17	20328161.55	1664383.09	16298296.29	12567155.58	3618270.75	112869.96	conventional	2017	TotalUS
9097	04-02-2018	0.87	37352360.59	21620180.90	20445501.03	1066830.22	19373134.37	13384586.80	5719096.61	269450.96	conventional	2018	TotalUS

18250 rows × 13 columns

In [ ]:

bin-boundaries

In [7]:
```
sorted_bin_boundaries_df = df.sort_values(by="Total Volume", ascending=True)
volume = df['Total Volume'].sort_values(ascending=True)

bin_size = volume.shape[0]//250
start = 0

for i in range(250):
    currentBin = volume[start:start+bin_size]
    max_bin = currentBin.max()
    min_bin = currentBin.min()

    for i in range(len(currentBin)):
        if( (currentBin.iloc[i]-max_bin) > (currentBin.iloc[i]-min_bin) ):
            currentBin.iloc[i] = min_bin
        else:
            currentBin.iloc[i] = max_bin


    sorted_bin_boundaries_df.iloc[start:start+bin_size, [2]] = currentBin
    start += bin_size



sorted_bin_boundaries_df
```

Out[7]:

	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region
10381	08-11-2015	1.59	934.95	3.95	3.95	0.00	76.66	73.33	3.33	0.00	organic	2015	MiamiFtLauderdale
9437	04-01-2015	1.73	934.95	0.00	59.82	0.00	320.00	320.00	0.00	0.00	organic	2015	BuffaloRochester
13189	30-10-2016	1.58	934.95	8.13	47.42	0.00	330.00	330.00	0.00	0.00	organic	2016	MiamiFtLauderdale
11698	12-07-2015	2.05	934.95	0.00	63.42	0.00	356.56	356.56	0.00	0.00	organic	2015	Syracuse
13193	02-10-2016	1.49	934.95	10.50	18.99	0.00	443.33	440.00	3.33	0.00	organic	2016	MiamiFtLauderdale
...	...	...	...	...	...	...	...	...	...	...	...	...	...
5493	08-05-2016	0.82	62505646.52	14223304.98	17896391.60	1993645.36	12211187.76	8747756.84	3342780.83	120650.09	conventional	2016	TotalUS
8353	07-05-2017	1.09	62505646.52	17076650.82	13549102.59	863471.88	15804696.31	11228049.63	4324231.19	252415.49	conventional	2017	TotalUS
5506	07-02-2016	0.76	62505646.52	16573573.78	20470572.61	2546439.11	12698112.39	9083373.04	3373077.87	241661.48	conventional	2016	TotalUS
8366	05-02-2017	0.77	62505646.52	22743616.17	20328161.55	1664383.09	16298296.29	12567155.58	3618270.75	112869.96	conventional	2017	TotalUS
9097	04-02-2018	0.87	62505646.52	21620180.90	20445501.03	1066830.22	19373134.37	13384586.80	5719096.61	269450.96	conventional	2018	TotalUS

18250 rows × 13 columns

In [ ]:

b. The dataset represents weekly retail scan data for National retail volume (units) and price. However, the company is interested in knowing the monthly (total per month) and annual sales (total per year), rather than the total per week. So, reduce the data accordingly.

In [52]:
```
# AveragePrice has string value.. so run question D. first and then this one..but run before E, because dates are changed
df['Date'] = pd.to_datetime(df['Date'], infer_datetime_format=True, errors='ignore')

df['Month'] = df['Date'].dt.month
# df['Year'] = df['Date'].dt.year

d = {
    1: "Jan",
    2: "Feb",
    3: "March",
    4: "April",
    5: "May",
    6: "June",
    7: "July",
    8: "Aug",
    9: "Sept",
    10: "Oct",
    11: "Nov",
    12: "Dec",
}


for each in d.keys():
    df.loc[df['Month'] == each, 'Month'] = d[each]

df
```

Out[52]:

	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region	Month
0	2015-12-27	1.33	64236.62	1036.74	54454.85	48.16	8696.87	8603.62	93.25	0.0	conventional	2015	Albany	Dec
1	2015-12-20	1.35	54876.98	674.28	44638.81	58.33	9505.56	9408.07	97.49	0.0	conventional	2015	Albany	Dec
2	2015-12-13	0.93	118220.22	794.70	109149.67	130.50	8145.35	8042.21	103.14	0.0	conventional	2015	Albany	Dec
3	2015-12-06	1.08	78992.15	1132.00	71976.41	72.58	5811.16	5677.40	133.76	0.0	conventional	2015	Albany	Dec
4	2015-11-29	1.29	51039.60	941.48	43838.39	75.78	6183.95	5986.26	197.69	0.0	conventional	2015	Albany	Nov
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
18245	2018-01-28	1.71	13888.04	1191.70	3431.50	0.00	9264.84	8940.04	324.80	0.0	organic	2018	WestTexNewMexico	Jan
18246	2018-01-21	1.87	13766.76	1191.92	2452.79	727.94	9394.11	9351.80	42.31	0.0	organic	2018	WestTexNewMexico	Jan
18247	2018-01-14	1.93	16205.22	1527.63	2981.04	727.01	10969.54	10919.54	50.00	0.0	organic	2018	WestTexNewMexico	Jan
18248	2018-01-07	1.62	17489.58	2894.77	2356.13	224.53	12014.15	11988.14	26.01	0.0	organic	2018	WestTexNewMexico	Jan
18249	2018-03-18	1.56	15896.38	2055.35	1499.55	0.00	12341.48	12114.81	226.67	0.0	organic	2018	WestTexNewMexico	March

18250 rows × 14 columns

In [53]:
```
# monthly data...
df.groupby(['year', 'Month', 'region']).agg({'Total Volume':sum, 'AveragePrice':sum})
```

Out[53]:

year	Month	region	Total Volume	AveragePrice
2015	**April**	**Albany**	1.999831e+05	12.25
		Atlanta	1.737203e+06	11.44
		BaltimoreWashington	3.096271e+06	11.64
		Boise	3.298737e+05	10.52
		Boston	1.841870e+06	12.53
...	**...**	**...**	...	...
2018	**March**	**Syracuse**	3.754490e+05	9.70
		Tampa	2.260110e+06	10.49
		TotalUS	1.725214e+08	10.37
		West	3.029377e+07	10.30
		WestTexNewMexico	3.740595e+06	11.38

2106 rows × 2 columns

In [54]:
```
# Yearly data...
df.groupby(['year', 'region']).agg({'Total Volume':sum, 'AveragePrice':sum})
```

Out[54]:

year	region	Total Volume	AveragePrice
2015	Albany	4.029896e+06	161.90
	Atlanta	2.323170e+07	143.58
	BaltimoreWashington	4.064558e+07	144.78
	Boise	3.784357e+06	144.01
	Boston	2.745499e+07	159.31
...	...	...	...
2018	Syracuse	1.046988e+06	30.01
	Tampa	6.642055e+06	31.94
	TotalUS	5.236323e+08	31.37
	West	9.247709e+07	31.13
	WestTexNewMexico	1.181672e+07	31.59

216 rows × 2 columns

In [ ]:

c. Summarize the number of missing values for each attribute

In [8]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18250 entries, 0 to 18249
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Date          18250 non-null  object
 1   AveragePrice  18222 non-null  object
 2   Total Volume  18250 non-null  float64
 3   4046          18250 non-null  float64
 4   4225          18250 non-null  float64
 5   4770          18250 non-null  float64
 6   Total Bags    18250 non-null  float64
 7   Small Bags    18250 non-null  float64
 8   Large Bags    18250 non-null  float64
 9   XLarge Bags   18250 non-null  float64
 10  type          18250 non-null  object
 11  year          18250 non-null  int64
 12  region        18250 non-null  object
dtypes: float64(8), int64(1), object(4)
memory usage: 1.8+ MB
```

In [9]:
```
df.isnull().sum()
```

Out[9]:
```
Date             0
AveragePrice    28
Total Volume     0
4046             0
4225             0
4770             0
Total Bags       0
Small Bags       0
Large Bags       0
XLarge Bags      0
type             0
year             0
region           0
dtype: int64
```

In [ ]:
```

```

d. GroupBy Avg Price

In [10]:
```
col = []
count = 0
for i in range(len(df['AveragePrice'])):
    flag = 1
    for x in str(df['AveragePrice'].iloc[i]):
        if( not (x.isdigit() or x=='.') ):
            col.append(float(0))
            flag = 0
            count += 1
            break

    if(flag):
        col.append( float(df['AveragePrice'].iloc[i]) )

df['AveragePrice'] = col

df_group_region = df.groupby(['region']).mean()
```

In [11]:
```
df_group_region
```

Out[11]:

region	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	year
Albany	1.538225	4.753787e+04	1.824082e+03	3.762121e+04	162.832337	7.929747e+03	6.647765e+03	1.153496e+03	128.488639	2016.147929
Atlanta	1.337959	2.621453e+05	1.461169e+05	3.121851e+04	311.385769	8.449856e+04	5.160573e+04	3.207004e+04	822.786036	2016.147929
BaltimoreWashington	1.510118	3.985619e+05	3.565622e+04	2.459829e+05	12466.730976	1.044561e+05	1.009397e+05	2.903985e+03	612.382722	2016.147929
Boise	1.331834	4.264257e+04	2.001951e+04	3.461682e+03	3186.787840	1.597459e+04	1.384004e+04	2.103634e+03	30.915207	2016.147929
Boston	1.483136	2.877929e+05	4.994610e+03	2.142199e+05	4982.294970	6.359609e+04	5.890659e+04	4.438365e+03	251.124231	2016.147929
BuffaloRochester	1.458195	6.793630e+04	1.776671e+03	3.166395e+04	144.335976	3.435134e+04	3.037998e+04	3.729050e+03	242.313432	2016.147929
California	1.395325	3.044324e+06	1.180376e+06	1.039915e+06	94204.276746	7.298290e+05	6.778038e+05	3.642585e+04	15599.342929	2016.147929
Charlotte	1.606036	1.051939e+05	2.237814e+04	3.662802e+04	11578.467574	3.460930e+04	3.149069e+04	2.912684e+03	205.927781	2016.147929
Chicago	1.556775	3.955690e+05	3.208308e+04	2.545190e+05	59069.204290	4.989763e+04	4.377116e+04	5.005547e+03	1120.925178	2016.147929
CincinnatiDayton	1.209201	1.317219e+05	5.411698e+03	6.105890e+04	3421.026598	6.182816e+04	1.675117e+04	4.429643e+04	780.564290	2016.147929
Columbus	1.252781	8.873776e+04	3.723706e+04	1.691030e+04	4955.845030	2.963428e+04	2.387808e+04	5.103195e+03	653.004497	2016.147929
DallasFtWorth	1.085592	6.166251e+05	3.270901e+05	1.395577e+05	12492.822811	1.374845e+05	1.207744e+05	1.543113e+04	1278.973609	2016.147929
Denver	1.218580	4.109542e+05	7.819187e+04	1.503270e+05	8006.405296	1.744290e+05	5.551362e+04	1.187601e+05	155.270059	2016.147929
Detroit	1.276095	1.876403e+05	5.557145e+04	3.479935e+04	25487.582544	7.177534e+04	5.899134e+04	7.905232e+03	4878.770533	2016.147929
GrandRapids	1.505000	8.938383e+04	1.459368e+03	5.101043e+04	9786.499467	2.712750e+04	2.228839e+04	2.357398e+03	2481.702278	2016.147929
GreatLakes	1.338550	1.744505e+06	2.772693e+05	7.907033e+05	148153.761095	5.283303e+05	3.647559e+05	1.446617e+05	18912.792781	2016.147929
HarrisburgScranton	1.513284	1.236948e+05	2.223527e+04	6.121225e+04	219.942692	4.002738e+04	3.778956e+04	1.780372e+03	457.450178	2016.147929
HartfordSpringfield	1.818639	1.499128e+05	3.904301e+03	1.119752e+05	503.708994	3.352962e+04	3.188923e+04	1.410794e+03	229.590888	2016.147929
Houston	1.047929	6.010884e+05	2.951861e+05	1.411965e+05	16140.446805	1.485653e+05	9.622831e+04	5.137208e+04	964.889793	2016.147929
Indianapolis	1.313994	8.953666e+04	7.176066e+03	4.279241e+04	4115.862071	3.544618e+04	1.955936e+04	1.504810e+04	838.727692	2016.147929
Jacksonville	1.510947	8.517753e+04	4.577429e+04	1.043414e+04	217.236834	2.875186e+04	1.283961e+04	1.560060e+04	311.644290	2016.147929
LasVegas	1.380917	1.608784e+05	5.854181e+04	4.545688e+04	3468.510976	5.341122e+04	2.656102e+04	2.680271e+04	47.497988	2016.147929
LosAngeles	1.216006	1.502653e+06	6.623650e+05	3.021715e+05	40291.995118	4.978240e+05	4.607281e+05	2.503889e+04	12057.041272	2016.147929
Louisville	1.286686	4.762427e+04	2.099263e+03	2.488955e+04	1004.427959	1.963103e+04	8.326157e+03	1.098558e+04	319.298757	2016.147929
MiamiFtLauderdale	1.428491	2.889740e+05	1.749707e+05	4.411622e+04	311.835680	6.957527e+04	3.319519e+04	3.547577e+04	904.313580	2016.147929
Midsouth	1.404763	1.503992e+06	3.267179e+05	6.579085e+05	56307.566065	4.630581e+05	3.975279e+05	6.145227e+04	4077.861331	2016.147929

region	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	year
Nashville	1.212101	1.053612e+05	5.381327e+04	1.127370e+04	1909.847426	3.836440e+04	2.971239e+04	8.322173e+03	329.840710	2016.147929
NewOrleansMobile	1.304793	1.351927e+05	7.930396e+04	1.305686e+04	616.088373	4.221580e+04	3.508744e+04	5.611536e+03	1516.830473	2016.147929
NewYork	1.727574	7.122311e+05	2.260156e+04	4.832954e+05	5168.053580	2.011661e+05	1.727860e+05	2.731656e+04	1063.543432	2016.147929
Northeast	1.601923	2.110299e+06	1.035243e+05	1.403801e+06	20167.587692	5.828059e+05	5.136774e+05	6.525460e+04	3873.868521	2016.147929
NorthernNewEngland	1.477396	2.116358e+05	7.997724e+03	1.586073e+05	3891.676598	4.113913e+04	3.336382e+04	7.447946e+03	327.367692	2016.147929
Orlando	1.506213	1.735524e+05	9.878966e+04	2.473253e+04	143.676124	4.988651e+04	2.641384e+04	2.275300e+04	719.665769	2016.147929
Philadelphia	1.632130	2.125408e+05	1.365444e+04	1.264747e+05	1774.944053	7.063678e+04	6.336398e+04	6.958744e+03	314.058343	2016.147929
PhoenixTucson	1.224438	5.788264e+05	3.415021e+05	1.158520e+05	6359.112189	1.151131e+05	6.589535e+04	4.875947e+04	458.308817	2016.147929
Pittsburgh	1.364320	5.564008e+04	1.175277e+04	2.160202e+04	849.011272	2.143628e+04	1.600053e+04	5.265377e+03	170.369882	2016.147929
Plains	1.436509	9.206761e+05	4.245898e+05	2.583886e+05	7380.101923	2.303172e+05	2.034338e+05	2.359066e+04	3292.713876	2016.147929
Portland	1.317722	3.270775e+05	6.866250e+04	1.035079e+05	8444.570473	1.464626e+05	1.228066e+05	2.342522e+04	230.805000	2016.147929
RaleighGreensboro	1.555118	1.426116e+05	3.497282e+04	4.934356e+04	11914.537219	4.638070e+04	4.449847e+04	1.601024e+03	281.207396	2016.147929
RichmondNorfolk	1.291331	1.249433e+05	3.691376e+04	4.756319e+04	3459.803284	3.700659e+04	3.448670e+04	2.153671e+03	366.224408	2016.147929
Roanoke	1.247929	7.408879e+04	2.205014e+04	2.358443e+04	78.497574	2.837570e+04	2.501876e+04	3.121787e+03	235.157840	2016.147929
Sacramento	1.621568	2.223779e+05	6.938285e+04	1.218252e+05	4595.151893	2.657476e+04	2.573030e+04	2.269676e+02	617.488491	2016.147929
SanDiego	1.398166	2.656566e+05	1.008363e+05	8.656350e+04	9195.626834	6.906115e+04	5.998754e+04	8.644449e+03	429.156716	2016.147929
SanFrancisco	1.804201	4.018645e+05	1.009967e+05	2.460150e+05	10796.791361	4.405599e+04	4.290636e+04	5.234889e+02	626.137840	2016.147929
Seattle	1.442574	3.231189e+05	6.351081e+04	1.066295e+05	3003.980030	1.499746e+05	1.255833e+05	2.418919e+04	202.101893	2016.147929
SouthCarolina	1.403284	1.797449e+05	8.405432e+04	3.737238e+04	4610.574793	5.370761e+04	3.968446e+04	1.335283e+04	670.327308	2016.147929
SouthCentral	1.101243	2.991952e+06	1.582963e+06	6.522190e+05	66259.744793	6.905102e+05	5.467914e+05	1.359078e+05	7810.979704	2016.147929
Southeast	1.398018	1.820232e+06	1.004991e+06	2.688179e+05	9358.670533	5.370640e+05	3.162819e+05	2.133361e+05	7446.044172	2016.147929
Spokane	1.445592	4.605111e+04	1.214222e+04	1.457577e+04	829.050562	1.850407e+04	1.588258e+04	2.600242e+03	21.250651	2016.147929
StLouis	1.430621	9.489004e+04	3.997294e+04	1.124101e+04	78.368846	4.359772e+04	3.530135e+04	7.631293e+03	665.082515	2016.147929
Syracuse	1.520325	3.237476e+04	9.816233e+02	1.890633e+04	83.167396	1.240363e+04	1.078595e+04	1.494128e+03	123.556036	2016.147929
Tampa	1.408846	1.952797e+05	1.057626e+05	3.171925e+04	115.353580	5.768252e+04	3.138484e+04	2.560189e+04	695.795059	2016.147929
TotalUS	1.319024	1.735130e+07	6.079693e+06	5.961573e+06	462056.754822	4.847931e+06	3.679175e+06	1.105279e+06	63477.013432	2016.147929
West	1.272219	3.215323e+06	1.179262e+06	8.898193e+05	60225.061746	1.086017e+06	6.589032e+05	4.246500e+05	2463.416124	2016.147929

region	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	year
WestTexNewMexico	1.262589	4.301718e+05	2.502364e+05	6.601179e+04	8837.720774	1.050859e+05	5.935676e+04	4.479605e+04	933.068601	2016.151786

In [51]:

```
df1 = df.loc[df.AveragePrice==0]['region']

for i in df1.index:
#     print(df1.loc[index])
    df.loc[df.index == i, 'AveragePrice'] = round(df_group_region.loc[df_group_region.index == df1.loc[i]]['AveragePrice'].values[0], 2)


df.head(20)
```

Out[51]:

	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region
0	27-12-2015	1.33	64236.62	1036.74	54454.85	48.16	8696.87	8603.62	93.25	0.0	conventional	2015	Albany
1	20-12-2015	1.35	54876.98	674.28	44638.81	58.33	9505.56	9408.07	97.49	0.0	conventional	2015	Albany
2	13-12-2015	0.93	118220.22	794.70	109149.67	130.50	8145.35	8042.21	103.14	0.0	conventional	2015	Albany
3	06-12-2015	1.08	78992.15	1132.00	71976.41	72.58	5811.16	5677.40	133.76	0.0	conventional	2015	Albany
4	29-11-2015	1.29	51039.60	941.48	43838.39	75.78	6183.95	5986.26	197.69	0.0	conventional	2015	Albany
5	22-11-2015	1.46	55979.78	1184.27	48067.99	43.61	6683.91	6556.47	127.44	0.0	conventional	2015	Albany
6	15-11-2015	1.40	83453.76	1368.92	73672.72	93.26	8318.86	8196.81	122.05	0.0	conventional	2015	Albany
7	08-11-2015	1.61	109428.33	703.75	101815.36	80.00	6829.22	6266.85	562.37	0.0	conventional	2015	Albany
8	01-11-2015	1.56	99811.42	1022.15	87315.57	85.34	11388.36	11104.53	283.83	0.0	conventional	2015	Albany
9	25-10-2015	1.21	74338.76	842.40	64757.44	113.00	8625.92	8061.47	564.45	0.0	conventional	2015	Albany
10	18-10-2015	1.25	84843.44	924.86	75595.85	117.07	8205.66	7877.86	327.80	0.0	conventional	2015	Albany
11	11-10-2015	1.09	64489.17	1582.03	52677.92	105.32	10123.90	9866.27	257.63	0.0	conventional	2015	Albany
12	04-10-2015	1.31	61007.10	2268.32	49880.67	101.36	8756.75	8379.98	376.77	0.0	conventional	2015	Albany
13	27-09-2015	0.99	106803.39	1204.88	99409.21	154.84	6034.46	5888.87	145.59	0.0	conventional	2015	Albany
14	20-09-2015	1.33	69759.01	1028.03	59313.12	150.50	9267.36	8489.10	778.26	0.0	conventional	2015	Albany
15	13-09-2015	1.28	76111.27	985.73	65696.86	142.00	9286.68	8665.19	621.49	0.0	conventional	2015	Albany
16	06-09-2015	1.11	99172.96	879.45	90062.62	240.79	7990.10	7762.87	227.23	0.0	conventional	2015	Albany
17	30-08-2015	1.07	105693.84	689.01	94362.67	335.43	10306.73	10218.93	87.80	0.0	conventional	2015	Albany
18	23-08-2015	1.34	79992.09	733.16	67933.79	444.78	10880.36	10745.79	134.57	0.0	conventional	2015	Albany
19	16-08-2015	1.33	80043.78	539.65	68666.01	394.90	10443.22	10297.68	145.54	0.0	conventional	2015	Albany

In [ ]:

In [ ]:

e. Discretize

In [55]:
```
d = {
    2015: "Old",
    2016: "Old",
    2017: "New",
    2018: "Recent"
}


for each in d.keys():
    df.loc[df['year'] == each, 'Date'] = d[each]


df
```

Out[55]:

	Date	AveragePrice	Total Volume	4046	4225	4770	Total Bags	Small Bags	Large Bags	XLarge Bags	type	year	region	Month
0	Old	1.33	64236.62	1036.74	54454.85	48.16	8696.87	8603.62	93.25	0.0	conventional	2015	Albany	Dec
1	Old	1.35	54876.98	674.28	44638.81	58.33	9505.56	9408.07	97.49	0.0	conventional	2015	Albany	Dec
2	Old	0.93	118220.22	794.70	109149.67	130.50	8145.35	8042.21	103.14	0.0	conventional	2015	Albany	Dec
3	Old	1.08	78992.15	1132.00	71976.41	72.58	5811.16	5677.40	133.76	0.0	conventional	2015	Albany	Dec
4	Old	1.29	51039.60	941.48	43838.39	75.78	6183.95	5986.26	197.69	0.0	conventional	2015	Albany	Nov
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
18245	Recent	1.71	13888.04	1191.70	3431.50	0.00	9264.84	8940.04	324.80	0.0	organic	2018	WestTexNewMexico	Jan
18246	Recent	1.87	13766.76	1191.92	2452.79	727.94	9394.11	9351.80	42.31	0.0	organic	2018	WestTexNewMexico	Jan
18247	Recent	1.93	16205.22	1527.63	2981.04	727.01	10969.54	10919.54	50.00	0.0	organic	2018	WestTexNewMexico	Jan
18248	Recent	1.62	17489.58	2894.77	2356.13	224.53	12014.15	11988.14	26.01	0.0	organic	2018	WestTexNewMexico	Jan
18249	Recent	1.56	15896.38	2055.35	1499.55	0.00	12341.48	12114.81	226.67	0.0	organic	2018	WestTexNewMexico	March

18250 rows × 14 columns

In [ ]:

In [ ]:

In [ ]:

In [ ]: