**Where should a drinks company run promotions?**

**1 Background**

**Project goal**

My company owns a chain of stores across Russia that sell a variety of alcoholic drinks. The company recently ran a very successful wine promotion in Saint Petersburg. It would like to run the same promotion in other regions, but budget constraints prevent it from running the campaign all over Russia. The marketing team has decided to focus on the 10 regions that show similar buying habits to Saint Petersburg.

**Available data**

We have access to the per capita sales of wine, beer, vodka, champagne and brandy between 1998 and 2016.

**Business questions**

Before making a recommendation to the marketing team, we shall focus on answering the following five business questions:

1. How does the alcohol consumption in Saint Petersburg compare to that in other regions?
2. How does the sale of wine correlate with the sale of other drinks?
3. Can we identify regions that show an overall similar alcohol consumption pattern as Saint Petersburg over the last 5 years?
4. Can we identify regions that show an overall similar wine consumption pattern as Saint Petersburg over the past 5 years?
5. Can we find correlations with the sales of other drinks in regions with a similar wine consumption as Saint Petersburg?

**2 Housekeeping**

**2.1 Load libraries**

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

from sklearn.cluster import KMeans

from sklearn.preprocessing import StandardScaler

*#set custom color palette*

colors = ["#A63B43", "#B18164", '#48132A', '#060A0E']

sns.set\_palette(sns.color\_palette(colors))

custom\_palette = sns.set\_palette(sns.color\_palette(colors))

*#custom function that creates a numerical data quality report*

def numerical\_dqr(df):

*#select numerical columns*

numerical = df.select\_dtypes(include = ['int', 'Int64', 'float']).columns.tolist()

*#data type*

data\_types = pd.DataFrame(

df[numerical].dtypes,

columns=['Data Type'])

*#missing data*

missing\_data = pd.DataFrame(

df[numerical].isnull().sum(),

columns=['Missing Values'])

*#unique values*

unique\_values = pd.DataFrame(

columns=['Unique Values'])

for row in list(df[numerical].columns.values):

unique\_values.loc[row] = [df[numerical][row].nunique()]

*#number of records*

count\_values = pd.DataFrame(

columns=['Records'])

for row in list(df[numerical].columns.values):

count\_values.loc[row] = [df[numerical][row].count()]

*#minimum value*

maximum\_values = pd.DataFrame(

columns=['Maximum Value'])

for row in list(df[numerical].columns.values):

maximum\_values.loc[row] = [df[numerical][row].max()]

*#maximum value*

minimum\_values = pd.DataFrame(

columns=['Minimum Value'])

for row in list(df[numerical].columns.values):

minimum\_values.loc[row] = [df[numerical][row].min()]

*#mean value*

mean\_values = pd.DataFrame(

columns=['Mean Value'])

for row in list(df[numerical].columns.values):

mean\_values.loc[row] = [df[numerical][row].mean()]

*#first quartile*

quartile\_1 = pd.DataFrame(

columns=['1st Quartile'])

for row in list(df[numerical].columns.values):

quartile\_1.loc[row] = [df[numerical][row].quantile(0.25)]

*#median*

median = pd.DataFrame(

columns=['Median'])

for row in list(df[numerical].columns.values):

median.loc[row] = [df[numerical][row].quantile(0.5)]

*#third quartile*

quartile\_3 = pd.DataFrame(

columns=['3rd Quartile'])

for row in list(df[numerical].columns.values):

quartile\_3.loc[row] = [df[numerical][row].quantile(0.75)]

*#join columns*

dq\_report\_num = data\_types.join(count\_values).join(missing\_data).join(unique\_values).join(minimum\_values).join(maximum\_values).join(mean\_values).join(quartile\_1).join(median).join(quartile\_3)

*#percentage missing*

dq\_report\_num['Missing %'] = (dq\_report\_num['Missing Values'] / len(df[numerical]) \*100)

*# change order of columns*

dq\_report\_num = dq\_report\_num[['Data Type', 'Records', 'Unique Values', 'Missing Values', 'Missing %', 'Mean Value', 'Minimum Value', '1st Quartile', 'Median', '3rd Quartile', 'Maximum Value', ]]

*#round*

dq\_report\_num[['Missing %', 'Mean Value', 'Minimum Value', '1st Quartile', 'Median', '3rd Quartile', 'Maximum Value']] = dq\_report\_num[['Missing %', 'Mean Value', 'Minimum Value', '1st Quartile', 'Median', '3rd Quartile', 'Maximum Value']].round(1)

*#return report*

return(dq\_report\_num)

**2.2 Load data**

df = pd.read\_csv(r'./data/russian\_alcohol\_consumption.csv')

df.head()

**3 Business questions**

**3.1 Question 1**

How does the alcohol consumption in Saint Petersburg compare to that in other regions?

*#create df with only Sait Petersburg*

df\_sp =df[df['region']== 'Saint Petersburg']

*#create df without Saint Petersburg*

df\_no\_sp = df[df['region']!= 'Saint Petersburg']

*#calculate average consumption df*

df\_average = df\_no\_sp.groupby('year').mean()

*#plot sales*

fig, axes = plt.subplots(2, 3, figsize=(25, 15))

axes[1,2].set\_visible(False)

*#plot Saint Petersburg*

sns.lineplot(data = df\_sp, x = 'year', y ='wine' , ax = axes[0,0], color = '#A63B43', label = 'Saint Petersburg')

sns.lineplot(data = df\_sp, x = 'year', y ='beer' , ax = axes[0,1], color = '#A63B43', label = 'Saint Petersburg')

sns.lineplot(data = df\_sp, x = 'year', y ='vodka' , ax = axes[0,2], color = '#A63B43', label = 'Saint Petersburg')

sns.lineplot(data = df\_sp, x = 'year', y ='champagne' , ax = axes[1,0], color = '#A63B43', label = 'Saint Petersburg')

sns.lineplot(data = df\_sp, x = 'year', y ='brandy' , ax = axes[1,1], color = '#A63B43', label = 'Saint Petersburg')

*#plot avearge*

sns.lineplot(data = df\_average, x = 'year', y ='wine' , ax = axes[0,0], color = '#B18164', label = 'Average other regions')

sns.lineplot(data = df\_average, x = 'year', y ='beer' , ax = axes[0,1], color = '#B18164', label = 'Average other regions')

sns.lineplot(data = df\_average, x = 'year', y ='vodka' , ax = axes[0,2], color = '#B18164', label = 'Average other regions')

sns.lineplot(data = df\_average, x = 'year', y ='champagne' , ax = axes[1,0], color = '#B18164', label = 'Average other regions')

sns.lineplot(data = df\_average, x = 'year', y ='brandy' , ax = axes[1,1], color = '#B18164', label = 'Average other regions')

*#set labels*

axes[0,0].set(title = 'Per capita wine sales', xlabel = "Year", ylabel = 'liters')

axes[0,1].set(title = 'Per capita beer sales', xlabel = "Year", ylabel = 'liters')

axes[0,2].set(title = 'Per capita vodka sales', xlabel = "Year", ylabel = 'liters')

axes[1,0].set(title = 'Per capita champagne sales', xlabel = "Year", ylabel = 'liters')

axes[1,1].set(title = 'Per capita brandy sales', xlabel = "Year", ylabel = 'liters')

*#set format x-axis*

from matplotlib.ticker import FormatStrFormatter

for c in axes:

for ax in c:

ax.xaxis.set\_major\_formatter(FormatStrFormatter('%.0f'))

plt.show()

**Conclusion**

The wine consumption in Saint Petersburg is above average when compared to other regions in Russia, fluctuating between 7 and 8 liters per person per since 2009. A similar pattern is visible in the consumption of champagne and brandy. In comparison to other regions, people in Saint Petersburg consume less beer.

Like other regions in Russia, the vodka consumption in Saint Petersburg has dropped rapidly over the last decade. This fits with the conclusions from a recently published [report](https://iwsc.net/news/wine/iwsc-market-insight-the-russian-wine-market) from IWSC that states that younger people tend to favor wine, whereas the older generation prefers spirits.

The market in Saint Petersburg can therefore be defined as having a high wine, brandy and champagne consumption, but a low beer and vodka consumption.

**3.2 Question 2**

How does the sale of wine correlate with the sale of other drinks?

*#create Pearson correlation matrix*

correlation = df.corr(method='pearson')

mask = np.triu(np.ones\_like(correlation, dtype=bool))

fig, ax = plt.subplots (figsize = (5, 5))

ax = sns.heatmap(correlation, mask=mask, cmap= "RdYlGn", center=0, linewidths=1, annot=True, fmt=".2f")

ax.set(title = 'Pearson correlation matrix')

ax.set\_xticklabels(ax.get\_xticklabels(), rotation=80)

ax.set\_yticklabels(ax.get\_xticklabels(), rotation=0)

plt.show()

**Conclusion**

Pearson’s Correlation coefficient is a measure for the strength of a linear relationship between two variables. A coefficient between 0.3 and 0.5 is generally considered as weak, meaning that the two variables show no strong linear relationship. A coefficient between 0.5 and 0.7 is considered as moderate, whereas a correlation between 0.7 and 0.9 is considered strong.

There is an overall moderate positive relationship between the consumption of wine and the consumption of brandy, indicating that regions that have a high wine consumption also tend to have a high brandy consumption. The correlation between wine consumption and both champagne and beer consumption balances on the edge of a weak and moderate. This could very well be an economic effect, indicating that regions that have a higher income tend to spend more money on alcohol.

It is interssting to see that there is only a weak relationship between wine and vodka consumption. Again, this fits with the idea that wine drinkers and generally younger Russians that want to distance themselves from the high-alcohol drinking observed by older generations and preferring a lower-alcohol drink to show sophistication. [Reference](https://eur03.safelinks.protection.outlook.com/?url=https%3A%2F%2Fwww.wineintelligence.com%2Frussias-wine-revolution%2F&data=04%7C01%7Cb.loznik%40agrifirm.com%7C7be88e27f05e4c56ea9908d998b76590%7Cd819b495b2a34cb799e7f5b00a8e874f%7C0%7C0%7C637708735713695281%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C1000&sdata=pvhK5MkvAuX%2FrO6woWMnmdMt1rKhuILyyNEu5xMS2Bo%3D&reserved=0).

Overall we could conclude that most wine is consumed in regions that also sell large amounts of brandy (and to a lesser extent beer and champagne). This cannot be said for the vodka market, as vodka tends to be consumed by the older generation, whereas wine is preferred by the younger generation.

**3.3 Question 3**

Can we identify regions that show an overall similar alcohol consumption pattern over the past 5 years as Saint Petersburg?

A company does not tend to make major business decisions based on one year of market data. On the other hand, historical data may be less irrelevant to the current market, as the market will have changed over time. We will therefore focus on a 5-year period between 2012 and 2016.

*#create df that only contains data from the last 5-year*

df\_5\_year = df[df['year']>=2012]

df\_5\_year.head()

**What is clustering**

Clustering is a type of unsupervised machine learning that divides data points into a number of groups such that the data points within a group are more similar to other data points within the group than data points in other groups.

In order to cluster we need to:

* Deal with any missing values in the dataset
* Determine the optimal number of clusters (for example, using the elbow method)
* Scale the dataset so that greater values are not considered as being more important than lower values

*#after printing a data quality report we can see that 3.3% of the data is missing*

numerical\_dqr(df\_5\_year)

*#drop regions with no data over the last 5 years*

df\_5\_year = df\_5\_year [(df\_5\_year ['region']!= 'Republic of Ingushetia') & (df\_5\_year ['region']!= 'Chechen Republic')]

*#for other regions: impute missing values with the mean of the region*

columns = ['wine', 'vodka', 'champagne', 'brandy', 'beer']

for column in columns:

df\_5\_year[column] = df\_5\_year[column].fillna(df\_5\_year.groupby('region')[column].transform('mean'))

df\_5\_year[column] = round(df\_5\_year[column],1)

*#pivot year column*

df\_5\_year\_pivot = df\_5\_year.pivot\_table(index = 'region',

columns = 'year',

fill\_value = np.nan,

aggfunc = np.mean)

*#create list of multilevel column names*

mi = df\_5\_year\_pivot .columns

mi.tolist()

*#create merged column names*

ind = pd.Index([ str(e[0]) + "-" + str(e[1]) for e in mi.tolist()])

*#set the list as the column titels*

df\_5\_year\_pivot .columns = ind

df\_5\_year\_pivot.head()

*#transform data*

scaler = StandardScaler()

scaled\_features = scaler.fit\_transform(df\_5\_year\_pivot)

*#create scaled df*

scaled\_5year = pd.DataFrame(scaled\_features , index=df\_5\_year\_pivot.index, columns=df\_5\_year\_pivot.columns)

scaled\_5year.head()

*#Create an ellbow plot to determine the optimal number of clusters*

ks = range(1, 10)

inertias = []

for k in ks:

*# Create a KMeans instance with k clusters: model*

kmeans = KMeans(n\_clusters=k).fit(scaled\_5year)

*# Append the inertia to the list of inertias*

inertias.append(kmeans.inertia\_)

*# Plot ks vs inertias*

plt.plot(ks, inertias, '-o')

plt.xlabel('number of clusters, k')

plt.ylabel('inertia')

plt.title("Elbow method")

plt.xticks(ks)

plt.show()

Based on the elbow plot and some trial-and-error we can conclude that two clusters is the most optimal for the current dataset.

*#fit K means with 2 clusters*

kmeans = KMeans(n\_clusters=2, random\_state =11 )

kmeans.fit(scaled\_5year)

*# Calculate the cluster labels: labels*

clusters = kmeans.predict(scaled\_5year)

df\_5\_year\_pivot["cluster"] = clusters

*#create a 5 year average column for each drink for plotting*

df\_5\_year\_pivot['5yr\_avg\_beer'] = round(df\_5\_year\_pivot[['beer-2012', 'beer-2013', 'beer-2014', 'beer-2015', 'beer-2016']].mean(axis=1),1)

df\_5\_year\_pivot['5yr\_avg\_wine'] = round(df\_5\_year\_pivot[['wine-2012', 'wine-2013', 'wine-2014', 'wine-2015', 'wine-2016']].mean(axis=1),1)

df\_5\_year\_pivot['5yr\_avg\_brandy'] = round(df\_5\_year\_pivot[['brandy-2012', 'brandy-2013', 'brandy-2014', 'brandy-2015', 'brandy-2016']].mean(axis=1),1)

df\_5\_year\_pivot['5yr\_avg\_champagne'] = round(df\_5\_year\_pivot[['champagne-2012', 'champagne-2013', 'champagne-2014', 'champagne-2015', 'champagne-2016']].mean(axis=1),1)

df\_5\_year\_pivot['5yr\_avg\_vodka'] = round(df\_5\_year\_pivot[['vodka-2012', 'vodka-2013', 'vodka-2014', 'vodka-2015', 'vodka-2016']].mean(axis=1),1)

*#summarize dataframe by cluster*

cluster\_agg = df\_5\_year\_pivot.groupby('cluster').agg({

'5yr\_avg\_beer': 'mean',

'5yr\_avg\_wine': 'mean',

'5yr\_avg\_brandy' : 'mean',

'5yr\_avg\_champagne' : 'mean',

'5yr\_avg\_vodka' : 'mean'

}).round(1).reset\_index()

cluster\_agg = cluster\_agg.rename(columns={"5yr\_avg\_beer": "avg 5yr beer consumption (L)",

"5yr\_avg\_wine": "avg 5yr wine consumption (L)",

"5yr\_avg\_brandy": "avg 5yr brandy consumption (L)",

"5yr\_avg\_champagne": "avg 5yr champagne consumption (L)",

"5yr\_avg\_vodka": "avg 5yr vodka consumption (L)"})

cluster\_agg

Clustering has identified two clusters. Cluster 0 can in general be identified as a having an overall high alcohol consumption, whereas cluster 1 has a lower overall alcohol consumption.

Saint Petersburg can be found in cluster 0.

fig, axes = plt.subplots(1, 5, figsize=(24, 5), sharex = True)

*#plot Saint Petersburg*

sns.boxplot(data = df\_5\_year\_pivot, x = 'cluster', y ='5yr\_avg\_wine' , ax = axes[0])

sns.boxplot(data = df\_5\_year\_pivot, x = 'cluster', y ='5yr\_avg\_beer' , ax = axes[1])

sns.boxplot(data = df\_5\_year\_pivot, x = 'cluster', y ='5yr\_avg\_brandy' , ax = axes[2])

sns.boxplot(data = df\_5\_year\_pivot, x = 'cluster', y ='5yr\_avg\_champagne' , ax = axes[3])

sns.boxplot(data = df\_5\_year\_pivot, x = 'cluster', y ='5yr\_avg\_vodka' , ax = axes[4])

*#set labels*

axes[0].set(title = 'Mean per capita wine sales by cluster', xlabel = "Cluster", ylabel = 'liters')

axes[1].set(title = 'Mean per capita beer sales by cluster', xlabel = "Cluster", ylabel = 'liters')

axes[2].set(title = 'Mean per capita brandy sales by cluster', xlabel = "Cluster", ylabel = 'liters')

axes[3].set(title = 'Mean per capita champagne sales by cluster', xlabel = "Cluster", ylabel = 'liters')

axes[4].set(title = 'Mean per capita vodka sales by cluster', xlabel = "Cluster", ylabel = 'liters')

*#create variables for St Petersburg*

avg\_wine = df\_5\_year\_pivot.loc['Saint Petersburg', '5yr\_avg\_wine']

avg\_beer = df\_5\_year\_pivot.loc['Saint Petersburg', '5yr\_avg\_beer']

avg\_brandy = df\_5\_year\_pivot.loc['Saint Petersburg', '5yr\_avg\_brandy']

avg\_champagne= df\_5\_year\_pivot.loc['Saint Petersburg', '5yr\_avg\_champagne']

avg\_vodka= df\_5\_year\_pivot.loc['Saint Petersburg', '5yr\_avg\_vodka']

*#set lines for St Petersburg*

axes[0].axhline(y=avg\_wine, color = 'k', ls = '--', alpha = 0.5)

axes[1].axhline(y=avg\_beer, color = 'k', ls = '--', alpha = 0.5)

axes[2].axhline(y=avg\_brandy, color = 'k', ls = '--', alpha = 0.5)

axes[3].axhline(y=avg\_champagne, color = 'k', ls = '--', alpha = 0.5)

axes[4].axhline(y=avg\_vodka, color = 'k', ls = '--', alpha = 0.5)

plt.show()

**Conclusion**

Two clusters were identified, where cluster 0 has an overall high alcohol consumption and cluster 1 has a lower alcohol consumption. Saint Petersburg can be found in cluster 0.

The boxplots show how the data is distributed for the two clusters and five different drinks. The horizontal line shows the mean sales for Saint Petersburg over the past 5 years.

These plots show that Saint Petersburg falls quite neatly in the cluster when we look at wine, brandy and champagne consumption, but not when we look at the consumption of beer or vodka.

We can therefore conclude that it is not possible to define clear clusters that have an overall similar alcohol consumption as Saint Petersburg. It may therefore be better to simply focus on the wine market.

**3.4 Question 4**

Can we identify regions that show an overall similar wine consumption pattern over the past 5 years as Saint Petersburg?

*#create df with wine consumption over the last 5 years*

wine\_5\_year = df[df['year']>=2012]

wine\_5\_year = wine\_5\_year[['year', 'region', 'wine']]

wine\_5\_year.head()

*#drop regions with no data over the last 5 years*

wine\_5\_year = wine\_5\_year [(wine\_5\_year ['region']!= 'Republic of Ingushetia') & (wine\_5\_year ['region']!= 'Chechen Republic')]

*#impute missing values with the mean of the region*

columns = ['wine']

for column in columns:

wine\_5\_year[column] = wine\_5\_year[column].fillna(wine\_5\_year.groupby('region')[column].transform('mean'))

wine\_5\_year[column] = round(wine\_5\_year[column],1)

*#create pivot table*

wine\_5\_year\_pivot = wine\_5\_year.pivot\_table(index = 'region', columns = 'year', fill\_value = np.nan, aggfunc = np.mean)

*#create list of multilevel column names*

mi = wine\_5\_year\_pivot .columns

mi.tolist()

*#create merged columnnames*

ind = pd.Index([ str(e[0]) + "-" + str(e[1]) for e in mi.tolist()])

*#set the list as the column titels*

wine\_5\_year\_pivot .columns = ind

wine\_5\_year\_pivot.head()

*#create mean wine consumption column*

wine\_5\_year\_pivot['avg\_wine\_5yr'] = round(wine\_5\_year\_pivot[['wine-2012', 'wine-2013', 'wine-2014', 'wine-2015', 'wine-2016']].mean(axis=1),1)

*#create minimum wine consumption column*

wine\_5\_year\_pivot['min\_wine\_5yr'] = round(wine\_5\_year\_pivot[['wine-2012', 'wine-2013', 'wine-2014', 'wine-2015', 'wine-2016']].min(axis=1),1)

*#create maximum wine consumption column*

wine\_5\_year\_pivot['max\_wine\_5yr'] = round(wine\_5\_year\_pivot[['wine-2012', 'wine-2013', 'wine-2014', 'wine-2015', 'wine-2016']].max(axis=1),1)

*#create 5 year change in wine consumption column*

wine\_5\_year\_pivot['perc\_change\_5yr'] = round((wine\_5\_year\_pivot['wine-2016'] - wine\_5\_year\_pivot['wine-2012'])/(wine\_5\_year\_pivot['wine-2012'])\*100,1)

wine\_5\_year\_pivot.head()

**Clustering**

*# transform data*

scaler = StandardScaler()

scaled\_features = scaler.fit\_transform(wine\_5\_year\_pivot)

*#create scaled df*

scaled\_wine\_5year = pd.DataFrame(scaled\_features , index=wine\_5\_year\_pivot.index, columns=wine\_5\_year\_pivot.columns)

*#Create an ellbow plot to determine the optimal number of features*

ks = range(1, 10)

inertias = []

for k in ks:

*# Create a KMeans instance with k clusters: model*

kmeans = KMeans(n\_clusters=k).fit(scaled\_wine\_5year)

*# Append the inertia to the list of inertias*

inertias.append(kmeans.inertia\_)

*# Plot ks vs inertias*

plt.plot(ks, inertias, '-o')

plt.xlabel('number of clusters, k')

plt.ylabel('inertia')

plt.title("Elbow method")

plt.xticks(ks)

plt.show()

From the elbow plot that shows a clear bend at 3 clusters, we can conclude that 3 clusters is the optimal amount of clusters for this dataset.

*#fit Kmeans with 3 clusters*

kmeans = KMeans(n\_clusters=3, random\_state =11 )

kmeans.fit(scaled\_wine\_5year)

*# Calculate the cluster labels: labels*

cluster = kmeans.predict(scaled\_wine\_5year)

scaled\_wine\_5year['cluster'] = cluster

wine\_5\_year\_pivot['cluster'] = cluster

fig, axes = plt.subplots(figsize=(7, 5), sharex = True)

*#set variables for Saint Petersburg*

avg\_wine = scaled\_wine\_5year.loc['Saint Petersburg', 'avg\_wine\_5yr']

change\_wine = scaled\_wine\_5year.loc['Saint Petersburg', 'perc\_change\_5yr']

*#plot 3 clusters*

ax =sns.scatterplot(data = scaled\_wine\_5year, x = 'perc\_change\_5yr', y ='avg\_wine\_5yr' , hue = 'cluster', palette = custom\_palette)

ax.set (title = 'Identified clusters', xlabel = 'Scaled 5 year change in wine consumption', ylabel = 'Scaled 5 year mean wine consumption')

*#annotate cluster*

ax.annotate("St Petersburg",

xy=(change\_wine, avg\_wine), xycoords='data',

xytext=(-2.5, 1), textcoords='data',

arrowprops=dict(arrowstyle="->",

connectionstyle="arc3") )

plt.show()

cluster\_agg = wine\_5\_year\_pivot.groupby('cluster').agg({

'avg\_wine\_5yr': 'mean',

'min\_wine\_5yr': 'mean',

'max\_wine\_5yr' : 'mean',

'perc\_change\_5yr' : 'mean'

}).round(1).reset\_index()

cluster\_agg = cluster\_agg.rename(columns={"avg\_wine\_5yr": "avg 5yr wine consumption (L)",

"min\_wine\_5yr": "avg minimum 5yr wine consumption (L)",

"max\_wine\_5yr": "avg maximum 5yr wine consumption (L)",

"perc\_change\_5yr": "Change in wine consumption 2012-2016 (%)"})

cluster\_agg

When only looking at the wine consumption over the last 5 years, we can identify 3 clusters:

* Cluster 1 has a low wine consumption and has seen a big drop in overall wine consumption over the last 5 years.
* Cluster 0 has an intermediate wine consumption and has seen an 8% decline in wine consumption over the past 5 years.
* Cluster 2 has the highest wine consumption and has a fairly similar decline in wine consumption as was observed in cluster 0.

Saint Petersburg can be found in cluster 2 and can therefore be defined as a high wine consuming region.

fig, axes = plt.subplots(1, 4, figsize=(25, 5), sharex = True)

*#plot clusters*

sns.boxplot(data = wine\_5\_year\_pivot, x = 'cluster', y ='avg\_wine\_5yr', ax = axes[0] )

sns.boxplot(data = wine\_5\_year\_pivot, x = 'cluster', y ='min\_wine\_5yr', ax = axes[1] )

sns.boxplot(data = wine\_5\_year\_pivot, x = 'cluster', y ='max\_wine\_5yr', ax = axes[2] )

sns.boxplot(data = wine\_5\_year\_pivot, x = 'cluster', y ='perc\_change\_5yr', ax = axes[3] )

*#create variables for Saint Petersburg*

avg\_wine = wine\_5\_year\_pivot.loc['Saint Petersburg', 'avg\_wine\_5yr']

avg\_wine\_change = wine\_5\_year\_pivot.loc['Saint Petersburg', 'perc\_change\_5yr']

min\_wine = wine\_5\_year\_pivot.loc['Saint Petersburg', 'min\_wine\_5yr']

max\_wine = wine\_5\_year\_pivot.loc['Saint Petersburg', 'max\_wine\_5yr']

*#set labels*

axes[0].set(title = 'Mean 5 year per capita wine sales by cluster', ylabel = 'liters')

axes[1].set(title = 'Minimum 5 year per capita wine sales by cluster', ylabel = 'liters')

axes[2].set(title = 'Maximum 5 year per capita wine sales by cluster', ylabel = 'liters')

axes[3].set(title = '5 Year change in per capita wine sales by cluster', ylabel = '%')

*#set lines for St Petersburg*

axes[0].axhline(y=avg\_wine, color = 'k', ls = '--', alpha = 0.5)

axes[1].axhline(y=min\_wine, color = 'k', ls = '--', alpha = 0.5)

axes[2].axhline(y=max\_wine, color = 'k', ls = '--', alpha = 0.5)

axes[3].axhline(y=avg\_wine\_change, color = 'k', ls = '--', alpha = 0.5)

plt.show()

**Conclusion**

When we only look at wine sales over the past 5 years, we can identify 3 separate clusters that can best be separated based on their mean wine consumption.

Saint Petersburg falls in the cluster that consumes that highest amount of wine. From a marketing perspective this would be the most interesting cluster to target. I would therefore recommend to focus on the regions that fall into cluster 2.

**3.5 Question 5**

Can we find correlations with the sales of other drinks in regions that have a similar wine consumption as Saint Petersburg?

*#create a list of all regions that are found in cluster 2: high wine consuming regions*

cluster2\_wine = wine\_5\_year\_pivot[wine\_5\_year\_pivot['cluster']==2]

wine\_regions = list(cluster2\_wine.index.values)

*#create a df with all high-consuming wine regions from 2012 onwards*

df\_5yr\_selected = df[(df['year']>=2012) & (df['region'].isin(wine\_regions))]

df\_5yr\_selected.head()

*#create pivot table*

df\_5yr\_selected\_pivot = df\_5yr\_selected.pivot\_table(index = 'region', columns = 'year', fill\_value = np.nan, aggfunc = np.mean)

*#create list of multilevel column names*

mi = df\_5yr\_selected\_pivot .columns

mi.tolist()

*#create merged columnnames*

ind = pd.Index([ str(e[0]) + "-" + str(e[1]) for e in mi.tolist()])

*#set the list as the column titels*

df\_5yr\_selected\_pivot .columns = ind

*#create a 5 year average column for each drink*

df\_5yr\_selected\_pivot['5yr\_avg\_beer'] = round(df\_5yr\_selected\_pivot[['beer-2012', 'beer-2013', 'beer-2014', 'beer-2015', 'beer-2016']].mean(axis=1),1)

df\_5yr\_selected\_pivot['5yr\_avg\_wine'] = round(df\_5yr\_selected\_pivot[['wine-2012', 'wine-2013', 'wine-2014', 'wine-2015', 'wine-2016']].mean(axis=1),1)

df\_5yr\_selected\_pivot['5yr\_avg\_brandy'] = round(df\_5yr\_selected\_pivot[['brandy-2012', 'brandy-2013', 'brandy-2014', 'brandy-2015', 'brandy-2016']].mean(axis=1),1)

df\_5yr\_selected\_pivot['5yr\_avg\_champagne'] = round(df\_5yr\_selected\_pivot[['champagne-2012', 'champagne-2013', 'champagne-2014', 'champagne-2015', 'champagne-2016']].mean(axis=1),1)

df\_5yr\_selected\_pivot['5yr\_avg\_vodka'] = round(df\_5yr\_selected\_pivot[['vodka-2012', 'vodka-2013', 'vodka-2014', 'vodka-2015', 'vodka-2016']].mean(axis=1),1)

*#create 5 year change in consumption column for each drink*

df\_5yr\_selected\_pivot['perc\_change\_5yr\_wine'] = round((df\_5yr\_selected\_pivot['wine-2016'] - df\_5yr\_selected\_pivot['wine-2012'])/(df\_5yr\_selected\_pivot['wine-2012'])\*100,1)

df\_5yr\_selected\_pivot['perc\_change\_5yr\_beer'] = round((df\_5yr\_selected\_pivot['beer-2016'] - df\_5yr\_selected\_pivot['beer-2012'])/(df\_5yr\_selected\_pivot['beer-2012'])\*100,1)

df\_5yr\_selected\_pivot['perc\_change\_5yr\_brandy'] = round((df\_5yr\_selected\_pivot['brandy-2016'] - df\_5yr\_selected\_pivot['brandy-2012'])/(df\_5yr\_selected\_pivot['brandy-2012'])\*100,1)

df\_5yr\_selected\_pivot['perc\_change\_5yr\_champagne'] = round((df\_5yr\_selected\_pivot['champagne-2016'] - df\_5yr\_selected\_pivot['champagne-2012'])/(df\_5yr\_selected\_pivot['champagne-2012'])\*100,1)

df\_5yr\_selected\_pivot['perc\_change\_5yr\_vodka'] = round((df\_5yr\_selected\_pivot['vodka-2016'] - df\_5yr\_selected\_pivot['vodka-2012'])/(df\_5yr\_selected\_pivot['vodka-2012'])\*100,1)

*#create Pearson correlation matrix*

correlation = df\_5yr\_selected\_pivot.corr(method='pearson')

mask = np.triu(np.ones\_like(correlation, dtype=bool))

fig, ax = plt.subplots (figsize = (25, 25))

ax = sns.heatmap(correlation, mask=mask, cmap= "RdYlGn", center=0, linewidths=1, annot=True, fmt=".2f")

ax.set(title = 'Pearson correlation matrix')

ax.set\_xticklabels(ax.get\_xticklabels(), rotation=80)

ax.set\_yticklabels(ax.get\_xticklabels(), rotation=0)

plt.show()

**Conclusion**

When we create a Pearson's correlation matrix for the regions that like Saint Petersburg show a high wine consumption, we notice that not a single column (apart from the wine columns) show a strong or even moderate correlation with the mean 5-year wine consumption column or 5 year change in wine consumption column.

I therefore conclude that within the high-consuming regions, there is no benefit at looking at the consumption of other drinks. I recommend to focus on wine consumption only.

**4 Conclusions and Recommendation**

**Summary of business questions**

**1 How does the alcohol consumption in Saint Petersburg compare to that in other regions?**

Saint Petersburg is a high wine, high brandy and high champagne consuming region with a low consumption of beer and vodka.

**2 How does the sale of wine correlate with the sale of other drinks?**

In general, we find that in regions with a high wine consumption the consumption, of other drinks (except for vodka) tends to be higher. This is likely an economic effect. The vodka market and wine market likely cater to different types of consumers.

**3 Can we identify regions that show an overall similar alcohol consumption pattern as Saint Petersburg over the last 5 years?**

Saint Petersburg fits nicely into a single cluster when we only look at wine, brandy and champagne consumption, but not when we also look at vodka and beer consumption. We can therefore not identify regions that show an overall similar alcohol consumption pattern as Saint Petersburg.

**4 Can we identify regions that show an overall similar wine consumption pattern as Saint Petersburg over the past 5 years?**

Yes, Saint Petersburg fits nicely into a cluster that includes regions with a fairly stable but high wine consumption pattern.

**5 Can we find correlations with the sales of other drinks in regions with a similar wine consumption as Saint Petersburg?**

Within high-consuming regions, no correlations with the consumption of other drinks were found.

**Recommendation**

fig, axes = plt.subplots( figsize=(20, 5) )

*#plot sorted wine df*

wine\_sorted = df\_5yr\_selected\_pivot.sort\_values('5yr\_avg\_wine')

ax = sns.barplot(data =wine\_sorted, y = '5yr\_avg\_wine', x = wine\_sorted.index, palette = 'Reds' )

*#set labels*

ax.set (title = 'Mean 5 year per capita wine consumption of high consuming regions', xlabel= 'Region', ylabel = 'liters')

ax.set\_xticklabels(ax.get\_xticklabels(), rotation=90)

ax.set\_ylim(5,)

*#set arrow*

ax.annotate("St Petersburg",

xy=(2, 8), xycoords='data',

xytext=(2,9), textcoords='data',

arrowprops=dict(arrowstyle="->",

connectionstyle="arc3") )

plt.show()

When we plot the average wine consumption of the high-consuming regions, we notice that Saint Petersburg has a relatively low wine consumption. This could be interesting from a marketing perspective because we can likely expect more growth in these regions than in the regions that already have a very high wine consumption.

I would therefore recommend to focus on the 10 regions with the lowest wine consumption.

*#create a series of the average wine consumption and region*

regions = list(df\_5yr\_selected\_pivot.index.values)

avg\_wine = list(df\_5yr\_selected\_pivot['5yr\_avg\_wine'])

wine\_series = pd.Series(avg\_wine, index = regions)

*#print the 10 regions with the lowest wine consumption*

wine\_series = wine\_series.drop(labels=['Saint Petersburg'])

print("Recommended regions with their average 5 year per capita wine consumption")

wine\_series.nsmallest(10)

Recommended regions with their average 5 year per capita wine consumption

Kostroma Oblast 7.2

Sverdlovsk Oblast 7.4

Magadan Oblast 7.7

Moscow Oblast 7.8

Bryansk Oblast 7.9

Kaliningrad Oblast 7.9

Kirov Oblast 7.9

Sakhalin Oblast 7.9

Smolensk Oblast 8.0

Kaluga Oblast 8.1

dtype: float64

The vast majority of the recommended regions are found around Moscow and western border of Russia. These oblasts contain several cities belonging to Russia's largest 50 cities, including: Moscow, Yekaterinburg, Kirov, Balashikha, Kaliningrad and Bryansk.

**Did you enjoy reading this publication?**  
Show your appreciation for