Data Preparation & Data Cleansing

0.1 Description of the dataset

This dataset includes the detailed sales data for all flavours of an ice-cream store in HKU from 2017-04-01 till 2017-09-30. The timeframe covers 3 major phases: the exam period, the summer vacation and the back to school period during August/September.

0.2 Data Cleansing

a) Drop redundant columns 'student', 'Tourism' and 'staff'.
b) Based on the goal of the analysis, remove the identity with 'event', and only keep the ones with 'tourism', 'Staff' and 'student'.
c) Modify the flavour to match the ones in two categories:
   1) Firstly, use fuzzy matching to find the relevant flavours in two categories;

```
result = []
for item in df['Product']:
    if (item.find('Chocolate')) != -1:
        result.append('Chocolate')
    elif (item.find('Hazelnut')) != -1:
        result.append('Hazelnut')
    elif (item.find('Coconut')) != -1:
        result.append('Coconut')
    elif (item.find('Green Tea')) != -1:
        result.append('Green Tea')
    elif (item.find('Caramel')) != -1:
        result.append('Caramel')
    elif (item.find('Vanilla')) != -1:
        result.append('Vanilla')
    elif (item.find('Pina Colada')) != -1:
        result.append('Pina Colada')
    elif (item.find('Coffee')) != -1:
        result.append('Coffee')
    elif (item.find('Almond')) != -1:
        result.append('Almond')
    elif (item.find('Waffle Cone')) != -1:
        result.append('Waffle Cone')
    elif (item.find('Bean')) != -1:
        result.append('Bean')
    else:
        result.append('N/A')

df['product_cleaned'] = result
```

   2) Then export the output to csv file to check whether there are some missed observations due to the spelling mistake/recording error and so on;
   3) Modify the fuzzy matching criteria to do the matching again, and export to csv to check again.

```
# the modified/improved code to categorize falvours
result = []
for item in df['Product']:
    if (item.find('Choco')) != -1:
        result.append('Chocolate')
    elif ((item.find('Hazelnut')) != -1) or ((item.find('Hezelnut')) != -1):
        result.append('Hazelnut')
    elif (item.find('Coco')) != -1:
        result.append('Coconut')
    elif (item.find('Green Tea')) != -1:
        result.append('Green Tea')
    elif (item.find('Caramel')) != -1:
        result.append('Caramel')
    elif (item.find('Vanilla')) != -1:
        result.append('Vanilla')
    elif (item.find('Pina Colada')) != -1:
        result.append('Pina Colada')
    elif (item.find('Coffee')) != -1:
        result.append('Coffee')
    elif (item.find('Almond')) != -1:
        result.append('Almond')
    elif (item.find('Waffle Cone')) != -1:
        result.append('Waffle Cone')
    elif (item.find('Bean')) != -1:
        result.append('Bean')
    else:
        result.append('N/A')

df['product_cleaned'] = result
```

d) Categorize these flavours into two categories (popular level 1 & popular level 2).
e) Drop irrelevant columns for the following 2 questions (overall 'Revenue', original uncleaned 'Product).
f) Modify the identity records to a more consistent format: tourism – tourist; Staff – staff; student – student.
g) The sample of the cleaned dataset:

```
df.head(10)
```

	identity	Sat 1st Apr 2017	Sun 2nd Apr 2017	Mon 3rd Apr 2017	Tue 4th Apr 2017	Wed 5th Apr 2017	Thu 6th Apr 2017	Fri 7th Apr 2017	Sat 8th Apr 2017	Sun 9th Apr 2017	...	Sat 23rd Sep 2017	Sun 24th Sep 2017	Mon 25th Sep 2017	Tue 26th Sep 2017	Wed 27th Sep 2017	Thu 28th Sep 2017	Fri 29th Sep 2017	Sat 30th Sep 2017	product_cleaned	category
0	tourist	0.0	0.0	40.0	0.0	0.0	0.0	0.0	0.0	0.0	...	80.0	0.0	0.0	40.0	0.0	0.0	0.0	80.0	N/A	N/A
1	staff	0.0	0.0	0.0	0.0	60.0	60.0	30.0	30.0	0.0	...	30.0	30.0	0.0	60.0	60.0	30.0	30.0	30.0	Caramel	popular level 1
2	student	390.0	0.0	806.0	0.0	650.0	520.0	494.0	208.0	0.0	...	26.0	26.0	494.0	390.0	494.0	156.0	442.0	78.0	Caramel	popular level 1
3	tourist	40.0	0.0	0.0	0.0	120.0	0.0	0.0	80.0	0.0	...	40.0	0.0	0.0	0.0	0.0	0.0	0.0	40.0	Chocolate	popular level 1
4	staff	30.0	0.0	60.0	0.0	90.0	60.0	0.0	30.0	0.0	...	60.0	60.0	0.0	0.0	0.0	0.0	0.0	0.0	N/A	N/A
5	tourist	0.0	0.0	40.0	0.0	0.0	0.0	0.0	0.0	0.0	...	40.0	0.0	0.0	0.0	0.0	0.0	0.0	40.0	N/A	N/A
6	student	208.0	0.0	338.0	0.0	416.0	104.0	312.0	208.0	0.0	...	104.0	26.0	182.0	104.0	0.0	78.0	78.0	0.0	Chocolate	popular level 1
7	staff	30.0	0.0	60.0	0.0	60.0	30.0	30.0	90.0	0.0	...	30.0	0.0	0.0	0.0	0.0	270.0	30.0	0.0	Chocolate	popular level 1
8	tourist	40.0	0.0	40.0	0.0	80.0	0.0	0.0	0.0	0.0	...	80.0	0.0	40.0	40.0	0.0	40.0	0.0	40.0	Caramel	popular level 1
9	student	338.0	0.0	234.0	0.0	260.0	468.0	130.0	26.0	0.0	...	156.0	78.0	572.0	52.0	26.0	0.0	52.0	26.0	N/A	N/A

10 rows × 186 columns

Part 1

1.1 Description of analysis

In this question, there are 2 parts: 1) Plot the time series of monthly sales for two different categories; 2) Test whether the average sales of 'popular level 1' flavors is greater than the average sales of 'popular level 2' by 2000 units, which is a two-sample t test.

1.2 Procedures

a) Drop irrelevant columns for this question ('identity', 'product_cleaned').
b) Filter the dataset with categories in 'product level 1' and 'product level 2'.
c) Group by the 'category' to see the aggregated sales on each day for both categories.
d) Transpose the dataset for further analysis.

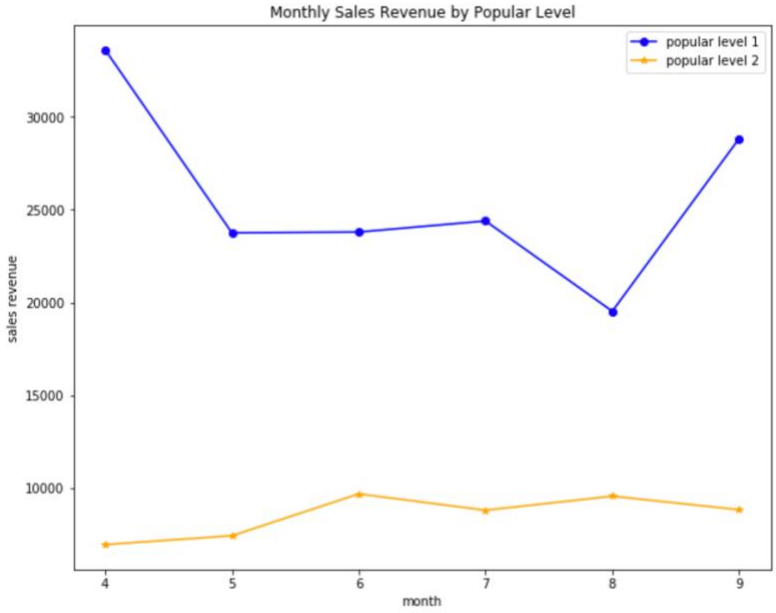
	date	popular_level_1	popular_level_2
0	Sat 1st Apr 2017	1412	291
1	Sun 2nd Apr 2017	0	0
2	Mon 3rd Apr 2017	2128	321
3	Tue 4th Apr 2017	0	0
4	Wed 5th Apr 2017	3078	253
...	...	...	...
178	Tue 26th Sep 2017	1262	318
179	Wed 27th Sep 2017	1529	374
180	Thu 28th Sep 2017	1632	150
181	Fri 29th Sep 2017	1802	358
182	Sat 30th Sep 2017	610	227

183 rows × 3 columns

e) Modify the 'date' column to the date format, and extract the month of each day.
f) Group by the 'month' to see the aggregated sales in each month for both categories.

	month	popular_level_1	popular_level_2
0	4	33606.84100	6973.68278
1	5	23755.66663	7449.99999
2	6	23797.86359	9700.89392
3	7	24393.24528	8823.84906
4	8	19534.00000	9577.83333
5	9	28822.16668	8850.33334

g) Plot the time series:



h) Hypothesis testing:

1) Let sample 1 be the monthly sales of popular level 1, and sample 2 the monthly sales of popular level 2;
2) Problem formulation:
H0: $\mu_1 - \mu_2 \leq 2000$; H1: $\mu_1 - \mu_2 > 2000$
3) For simplification, we modify μ_2 as μ_{2_new}, which is $\mu_2 + 2000$. In this case, the problem becomes H0: $\mu_1 - \mu_{2\_new} \leq 0$; H1: $\mu_1 - \mu_{2\_new} > 0$
4) Check summary statistics of two samples (mean, standard deviation) and do Levene test to see whether the variances of two groups are equal;

```
# to test u1-u2>2000, modify u2 to u2+2000 as u2_new
df_1['popular_level_2_modified'] = df_1['popular_level_2'] + 2000
```

```
# summary statistics
u1,u2_new=np.mean(df_1['popular_level_1']),np.mean(df_1['popular_level_2_modified'])
sd1,sd2_new=np.std(df_1['popular_level_1']),np.std(df_1['popular_level_2_modified'])
print(u1,u2_new)
print(sd1,sd2_new)
```

```
25651.63053 10562.765403333333
4460.5257690359085 1019.8030740511662
```

```
stat, p = levene(df_1['popular_level_1'],df_1['popular_level_2_modified'])
stat, p
# p>0.05, we do not reject H0, which means the variances of two samples are statistically equal
```

```
(2.563551402081902, 0.14043548283378937)
```

5) After implementation in Python, the following output is generated:

```
# H0:u1<=u2_new  H1:u1>u2_new (here u2_new is the modified u2, which is the original u2+2000)
# equal variance
def t_test(x,y,alternative='both-sided'):
            t, double_p = ttest_ind(x,y,equal_var = True)
            if alternative == 'both-sided':
                pval = double_p
            elif alternative == 'greater':
                if np.mean(x) > np.mean(y):
                    pval = double_p/2.
                else:
                    pval = 1.0 - double_p/2.
            elif alternative == 'less':
                if np.mean(x) < np.mean(y):
                    pval = double_p/2.
                else:
                    pval = 1.0 - double_p/2.
            return (t,pval)

t_test(df_1['popular_level_1'], df_1['popular_level_2_modified'],alternative='greater')
# p<0.001, so we reject H0
```

```
(7.373806420674885, 1.1933167674502398e-05)
```

6) Conclusion:
t=7.37, p<0.001, so we reject H0. That being said, the original claim holds. The average sales of 'popular level 1' flavors is greater than the average sales of 'popular level 2' by 2000 units.

Part 2

2.1 Description of analysis

In this question, there are 2 parts: 1) Plot the time series of daily sales for different customer groups under two different categories; 2) Provide recommendations based on the time series visualization.

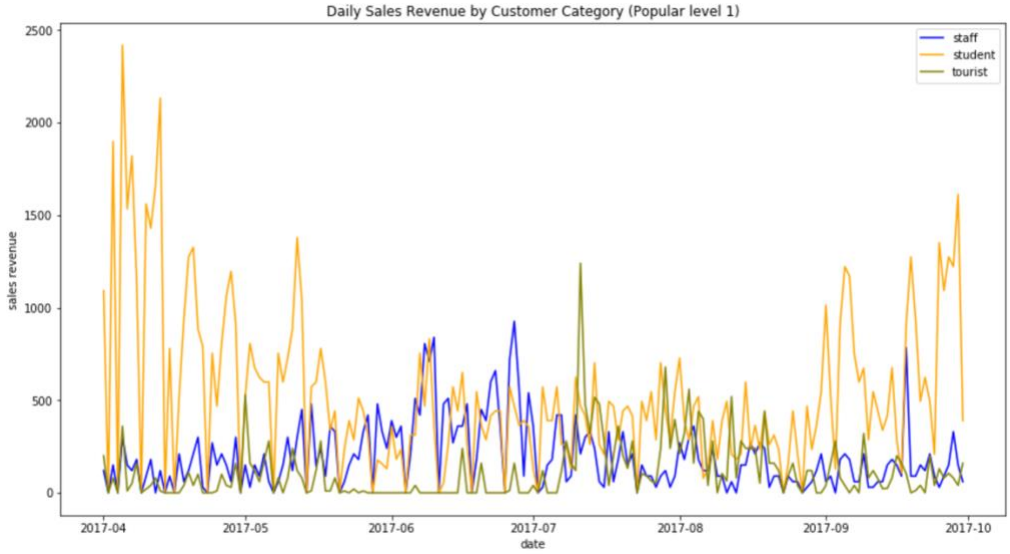
2.2 Procedures

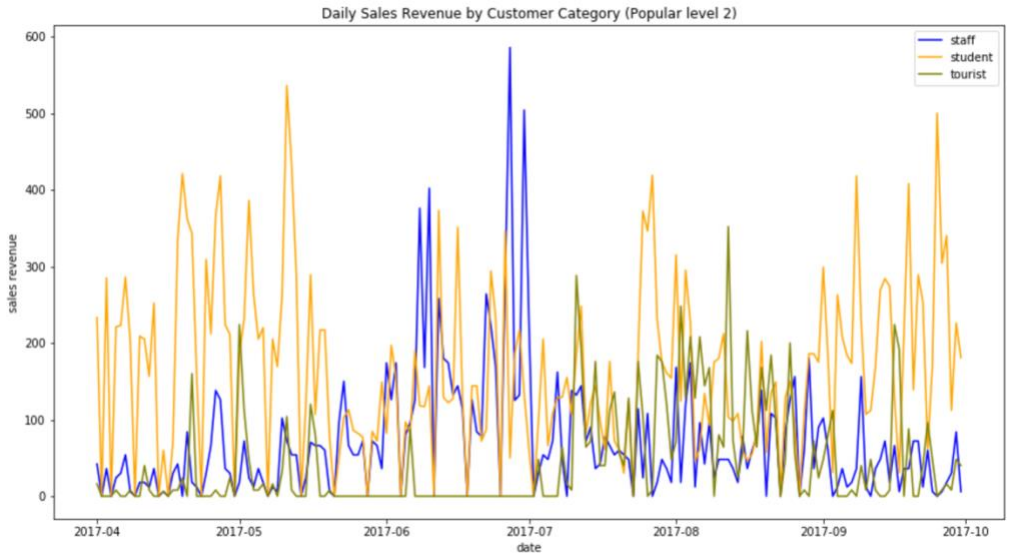
a) Similar to the data processing steps in Question 1, generate 2 tables, indicating the time series for popular level 1 (left) and popular level 2 (right).

	staff	student	tourist	date_new
0	120	1092	200	2017-04-01
1	0	0	0	2017-04-02
2	150	1898	80	2017-04-03
3	0	0	0	2017-04-04
4	300	2418	360	2017-04-05
...	...	...	...	...
178	90	1092	80	2017-09-26
179	150	1274	105	2017-09-27
180	330	1222	80	2017-09-28
181	150	1612	40	2017-09-29
182	60	390	160	2017-09-30

	staff	student	tourist	date_new
0	42	233	16	2017-04-01
1	0	0	0	2017-04-02
2	36	285	0	2017-04-03
3	0	0	0	2017-04-04
4	24	221	8	2017-04-05
...	...	...	...	...
178	6	304	8	2017-09-26
179	18	340	16	2017-09-27
180	30	112	8	2017-09-28
181	84	226	48	2017-09-29
182	6	181	40	2017-09-30

b) Plot the time series of these two categories:

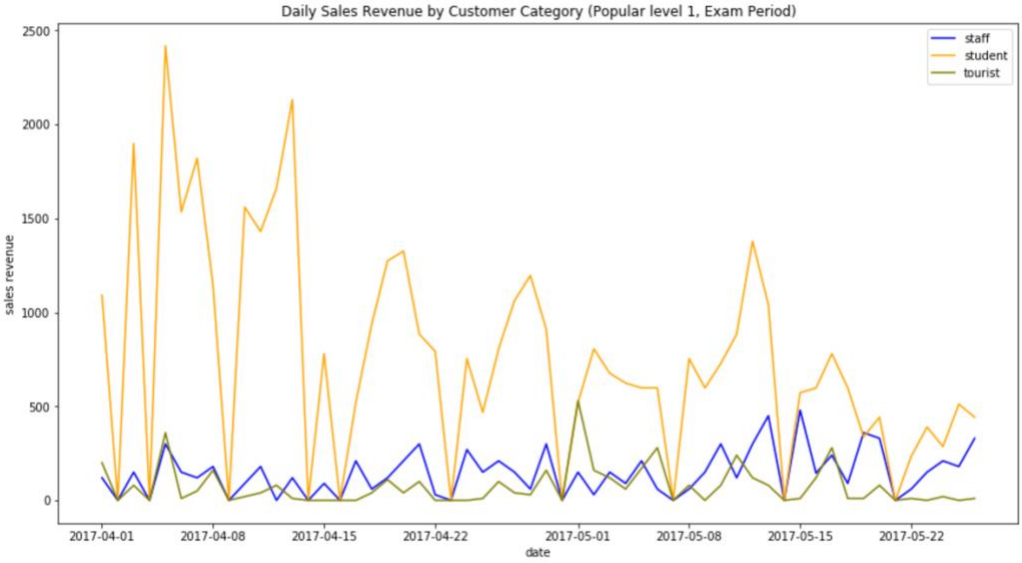


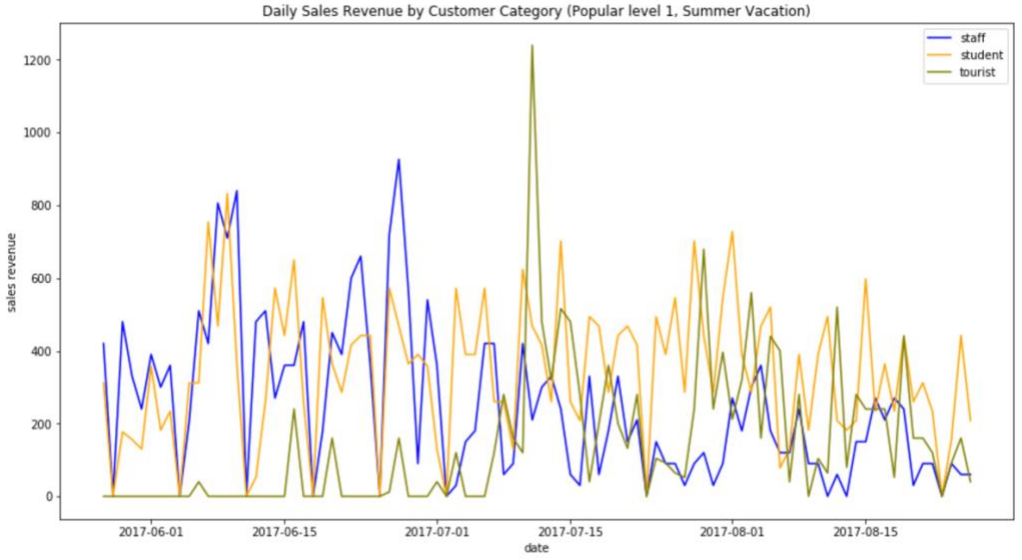


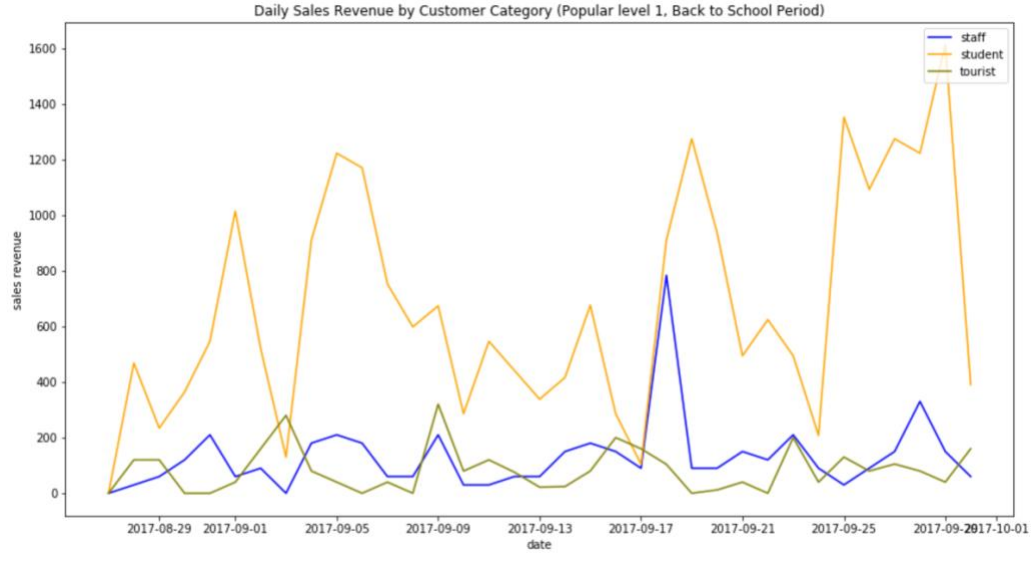
c) Discussion:
As can be seen, there are some zero values as well as some potential outliers. From the business perspective, on some Sundays the store is closed, so the zero values are meaningful, indicating the store is not in operation on certain days. As for potential outliers, we can see in popular level 1, there is a huge jump in July for tourists. After examining the causes, it is due to an event happening then, and many attendees came to buy the 'Green Tea' flavour. Based on this, such data points are meaningful too, as they represent the true business situation. So for here we do not remove or modify any values.

d) To look at the time series more clearly, the series is divided into 3 parts based on the academic calendar of HKU: Exam period (2017-04-01~2017-05-26), Summer vacation (2017-05-27~2017-08-26), Back to school period (2017-08-27~2017-09-30).

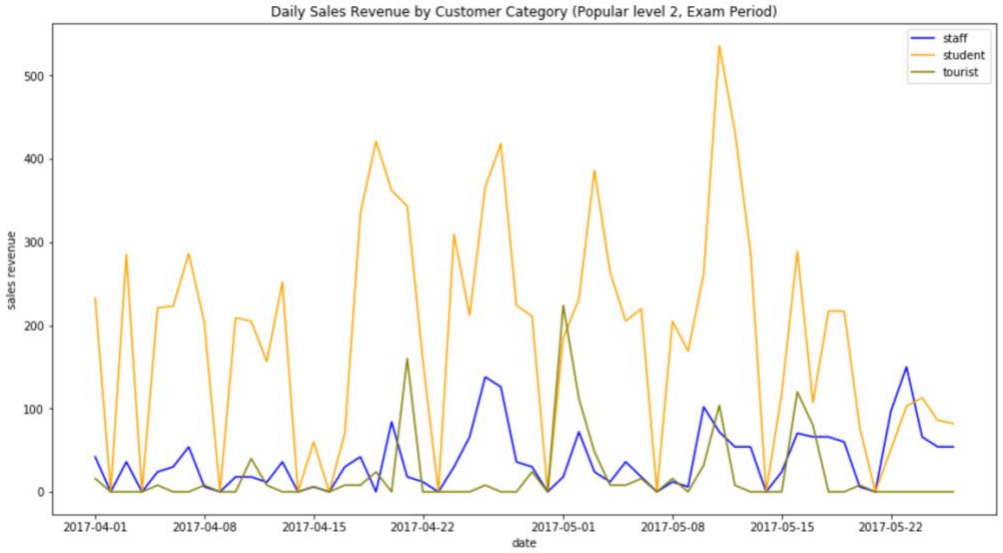
e) Subplots for popular level 1:

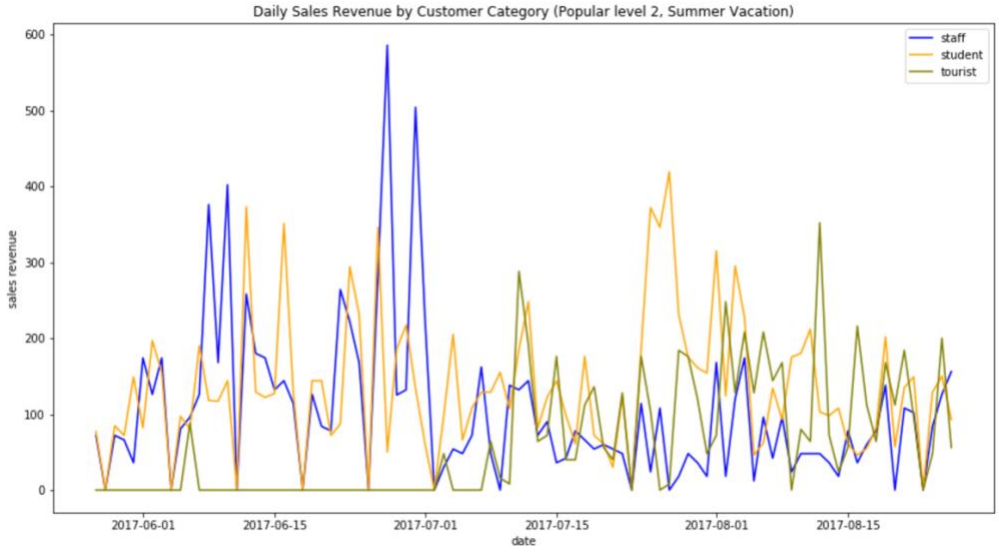

Daily Sales Revenue by Customer Category (Popular level 1, Exam Period)
staff
student
tourist
sales revenue
date
2017-04-01
2017-04-08
2017-04-15
2017-04-22
2017-05-01
2017-05-08
2017-05-15
2017-05-22

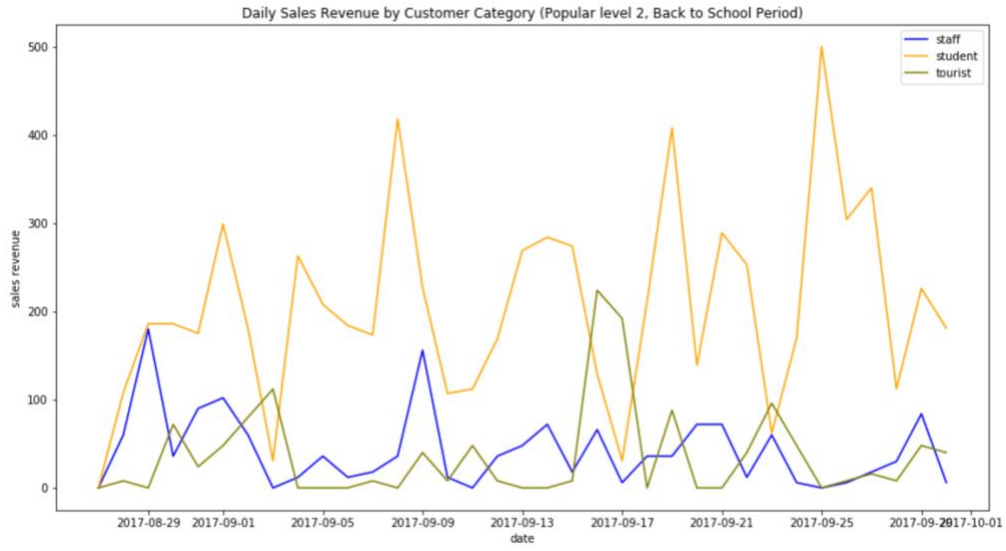

Daily Sales Revenue by Customer Category (Popular level 1, Summer Vacation)
staff
student
tourist
sales revenue
date
2017-06-01
2017-06-15
2017-07-01
2017-07-15
2017-08-01
2017-08-15


Daily Sales Revenue by Customer Category (Popular level 1, Back to School Period)
staff
student
tourist
sales revenue
date
2017-08-29
2017-09-01
2017-09-05
2017-09-09
2017-09-13
2017-09-17
2017-09-21
2017-09-25
2017-09-29
2017-10-01

f) Subplots for popular level 2:







g) Summary:
    1) During exam period, sales from students are much more than those from staff and tourists for both categories.
    2) During summer vacation, there seems to be a stable customer flow from students. The sales from staff are high in the beginning of this period, but decrease quickly starting July. Tourists sales boom starting July till the end of summer vacation. The overall sales are smaller than those in exam period for popular level 1, but kind of similar for popular level 2.
    3) During back to school period, sales from students become dominant again. The overall sales are smaller than those in exam period for popular level 1, but similar for popular level 2.
h) Recommendations:
    1) Offer flavours that students are more in favor of during exam period;
    2) Conduct promotion and organize events during summer vacation. For one thing, the customer flow is the most diverse in summer, by conducting promotion, it can attract more customers to come, and can facilitate the market research on customers' flavour preference from different customer groups. For another, organizing events can potentially attract a large group of customers(tourists) on certain days, which can make up for the smaller customer flow during summer vacation.
    3) Conduct promotion to attract new students during back to school period. By offering special discounts and recommending popular flavours to new students, it can build a new group of loyal customers.