Task 2

For this task, Python is used with library Pandas.

The label “Flags” and “Flag Text” are obtained in Task 1, so they are used directly in this task.

First step is read the file and convert the label “Date” to type *datetime*, which makes further operation convenient.

```
data = pd.read_csv("rawpvr_2018-02-01_28d_1083 TueFri.csv")
data["Date"] = pd.to_datetime(data["Date"], format="%Y-%m-%d %H:%M:%S.%f")
```

Second, create a new label called “Day_Number” to store the n^{th} day corresponding to the label “Date”. For example, “Date: 2018-02-02 00:00:03.050000” is “Day_Number: 2”. And also create a new label called “Hour” to store the specific hour corresponding to the label “Date”. For example, “Date: 2018-02-02 00:00:03.050000” is “Hour: 0”.

```
data["Day_Number"] = data["Date"].dt.day
data["Hour"] = data["Date"].dt.hour
```

Third, filter the data that satisfied the requirements of Tuesday between 9:00 and 10:00, and in North lanes. It has 9721 rows and 12 columns.

```
tue_data = data[(data["Hour"]==9) & (data["Direction Name"]=="North") & (data["Flags"]==2)]
print("Tuesday data shape: ", tue_data.shape)
```

✓ 0.8s

```
Tuesday data shape:  (9721, 12)
```

Now the first three rows of filtered data:

```
print(tue_data.head(3))
```

✓ 0.5s

	Date	Lane	Lane Name	Direction	Direction Name	\
78847	2018-02-06 09:00:01.030	2	NB_MID	1	North	
78848	2018-02-06 09:00:01.030	3	NB_OS	1	North	
78849	2018-02-06 09:00:01.170	1	NB_NS	1	North	

	Speed (mph)	Headway (s)	Gap (s)	Flags	Flag Text	Day_Number	Hour
78847	32.932	5.026	NaN	2	Tuesday	6	9
78848	32.310	3.946	NaN	2	Tuesday	6	9
78849	31.691	3.035	NaN	2	Tuesday	6	9

Now there are two circumstances, first is dividing the data depend on label “Day_Number” only, this gives four groups of data. Second is dividing the data depend on “Day_Number” and “Lane Name”, this gives 12 groups of data.

Four groups:

Use groupby() function and size() function to get the traffic volume of each Tuesday.

```
tue_traffic_volume = tue_data.groupby("Day_Number").size()
print(tue_traffic_volume)
```

✓ 0.3s

```
Day_Number
6     2537
13    2447
20    2311
27    2426
dtype: int64
```

Finally, use functions max()-min() to get range, quantile(0.25) to get first quartile, quantile(0.5) to get second quartile, quantile(0.75) to get third quantile and quantile(0.75)-quantile(0.25) to get interquartile range.

```
tue_traffic_volume_range = tue_traffic_volume.max() - tue_traffic_volume.min()
tue_traffic_volume_fir_quar = tue_traffic_volume.quantile(0.25)
tue_traffic_volume_sec_quar = tue_traffic_volume.quantile(0.5)
tue_traffic_volume_thir_quar = tue_traffic_volume.quantile(0.75)
tue_traffic_volume_inter_range = tue_traffic_volume_thir_quar - tue_traffic_volume_fir_quar
```

The result of 4 groups:

Range	226
1st Quartile	2397.25
2nd Quartile	2436.5
3rd Quartile	2469.5
Interquartile Range	72.25

Twelve Groups

Use groupby() function and size() function to get the traffic volume of each lane in each Tuesday.

```
tue_traffic_volume2 = tue_data.groupby(["Day_Number", "Lane Name"]).size()
print(tue_traffic_volume2)
```

```
Day_Number  Lane Name
6           NB_MID      879
            NB_NS       743
            NB_OS       915
13          NB_MID      856
            NB_NS       710
            NB_OS       881
20          NB_MID      823
            NB_NS       682
            NB_OS       806
27          NB_MID      831
            NB_NS       780
            NB_OS       815
dtype: int64
```

Use same steps as 4 groups to get range, first quartile, second quartile, third quartile and interquartile range.

```
tue_traffic_volume2_range = tue_traffic_volume2.max()-tue_traffic_volume2.min()
tue_traffic_volume2_fir_quar = tue_traffic_volume2.quantile(0.25)
tue_traffic_volume2_sec_quar = tue_traffic_volume2.quantile(0.5)
tue_traffic_volume2_thir_quar = tue_traffic_volume2.quantile(0.75)
tue_traffic_volume2_inter_range = tue_traffic_volume2_thir_quar - tue_traffic_volume2_fir_quar
```

The result of 12 groups:

Range	233
1st Quartile	770.75
2nd Quartile	819
3rd Quartile	861.75
Interquartile Range	91

Interpretation:

Range and interquartile distance both describe the dispersion of data. However, range is easily affected by outliers. For interquartile distance, outliers are excluded, but only partial data is used. The analysis is performed by combining these values and described in 4 groups and 12 groups respectively.

Four Groups

Both of the values of range and interquartile range are not too large, which means that the traffic volume in the north direction every Tuesday between 9am to 10am of the month does not vary much. It can be inferred that if there are no special circumstances (such as festivals or emergencies), the traffic volume on every Tuesday between 9am to 10am in the following months should be around the average value ($\frac{2537+2447+2311+2426}{4} \approx 2430$). In addition, there may not many outliers on this specific traffic volume of north direction.

Twelve Groups

Both of the values of range and interquartile range are not too large, which means that the traffic volume in the north direction of **each lane** every Tuesday between 9am to 10am of the month does not vary much. It can be inferred that if there are no special circumstances (such as festivals or emergencies), the traffic volume of **each lane** on every Tuesday between 9am to 10am in the following months should be around the average value ($\frac{\frac{897+743+915}{3}+\frac{856+710+881}{3}+\frac{823+682+806}{3}+\frac{931+780+815}{3}}{4} \approx 810$). In addition, there may not many outliers on this specific traffic volume of north direction.

Task 3

For this task, Python is used with libraries Pandas and Matplotlib.

Tuesday is chose to visualize the average traffic volume for each hour.

The label “Flags” and “Flag Text” are obtained in Task 1, so they are used directly in this task.

The first and second steps are the same as task 2 first two steps. Following is the codes:

```
data = pd.read_csv("rawpvr_2018-02-01_28d_1083 TueFri.csv")
data["Date"] = pd.to_datetime(data["Date"], format="%Y-%m-%d %H:%M:%S.%f")
data["Day_Number"] = data["Date"].dt.day
data["Hour"] = data["Date"].dt.hour
```

From third step, there are two circumstances. First is north direction, and second is south direction.

North Direction

Third, filter the data that satisfied the requirements of Tuesday and in North lanes. It has 124578 rows and 12 columns.

```
tue_data_north = data[(data["Direction Name"]=="North") & (data["Flags"]==2)]
print("Tuesday data north lane shape: ", tue_data_north.shape)
```

✓ 0.8s

```
Tuesday data north lane shape:  (124578, 12)
```

Use groupby() function and size() function to get the traffic volume of each hour on Tuesday.

```
tue_data_north_total = tue_data_north.groupby(["Hour"]).size()
```

The values currently obtained are the total traffic volume of each hour, the average value should be divided by 4.

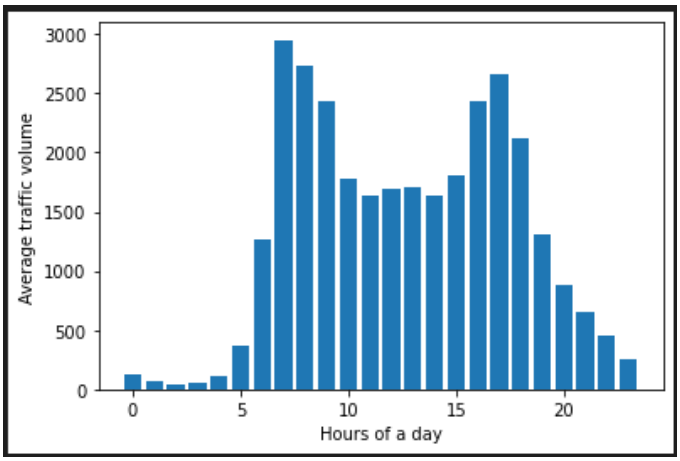
```
    tue_data_north_average = tue_data_north_total/4
    print("Average traffic volume of each hour", tue_data_north_average)
```

```
Average traffic volume of each hour Hour
0      126.50
1       63.00
2       42.50
3       53.00
4      108.25
5      361.50
6     1260.00
7     2951.50
8     2727.00
9     2430.25
10    1776.75
11    1634.25
12    1698.75
13    1707.50
14    1642.00
15    1805.25
16    2429.00
17    2655.75
18    2121.50
19    1308.25
20     882.75
21     648.75
22     453.75
23     256.75
dtype: float64
```

Then visualize the values as bar plot

```
plt.bar(list(range(0,24)), tue_data_north_average)
plt.xlabel("Hours of a day")
plt.ylabel("Average traffic volume")
plt.show()
```

The final result of North Lanes average traffic volume per hour:



South Direction

Third, filter the data that satisfied the requirements of Tuesday and in South lanes. It has 123439 rows and 12 columns.

```
tue_data_south = data[(data["Direction Name"]=="South") & (data["Flags"]==2)]
print("Tuesday data south lane shape: ", tue_data_south.shape)
```

✓ 0.7s

```
Tuesday data south lane shape:  (123439, 12)
```

Use groupby() function and size() function to get the traffic volume of each hour on Tuesday.

```
tue_data_south_total = tue_data_south.groupby("Hour").size()
```

The values currently obtained are the total traffic volume of each hour, the average value should be divided by 4.

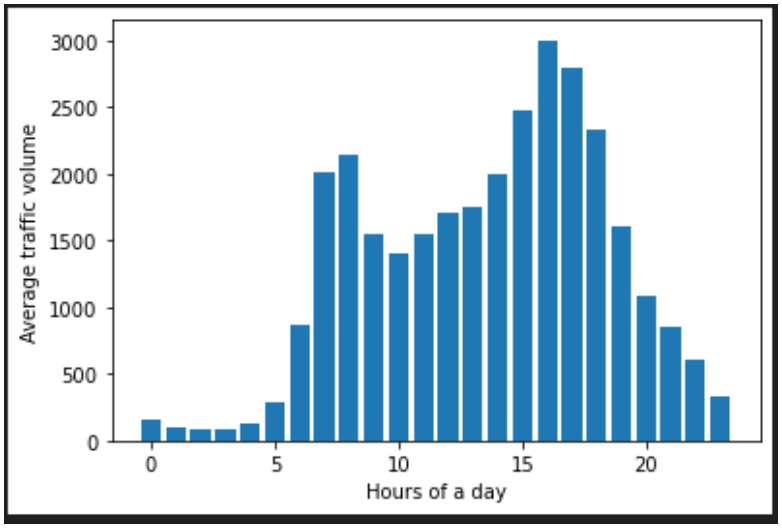
```
tue_data_south_average = tue_data_south_total/4
print(tue_data_south_average)
```

```
Hour
0      162.75
1       92.75
2       77.50
3       79.75
4      120.50
5      289.75
6      862.25
7     2015.00
8     2135.00
9     1548.25
10    1402.00
11    1543.50
12    1709.25
13    1749.50
14    1992.00
15    2481.50
16    3001.25
17    2794.00
18    2330.50
19    1601.75
20    1080.25
21     847.00
22     612.75
23     331.00
dtype: float64
```

Then visualize the values as bar plot

```
plt.bar(list(range(0,24)), tue_data_south_average)
plt.xlabel("Hours of a day")
plt.ylabel("Average traffic volume")
plt.show()
```

The final result of South Lanes average traffic volume per hour:



Interpretation:

As can be seen from the two bar plot above, for Tuesday, the general trend of north direction and south direction traffic volume is the same. There is a low peak time at 0 to 4 o'clock and a peak time at 7 to 9 o'clock, and 15 to 18 o'clock. For the north direction, the lowest peak time occurs at 2:00 with a value of 42.5 and the highest peak time occurs at 7:00 with a value of 2951.5. For the south direction, the lowest peak time also occurs at 2:00 with a value of 77.50 and the highest peak time occurs at 16:00 with a value of 3001.25.

It can be inferred that more people go to work in the north direction, and similarly more people go home after work in the south direction. And most people go to work at 7:00 and 8:00, and leave work at 16:00 to 17:00.

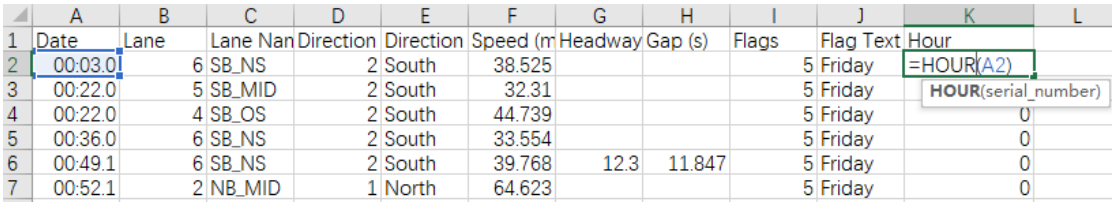
Task 4

For this task, excel is used.

Tuesday is chose to visualize the average traffic volume for each hour.

The label “Flags” and “Flag Text” are obtained in Task 1, so they are used directly in this task.

First create a new label called “Hour”, this column stores the specific hour corresponding to label “Date”. This can be done by using HOUR() function



	A	B	C	D	E	F	G	H	I	J	K	L
1	Date	Lane	Lane Nan	Direction	Direction	Speed (m	Headway	Gap (s)	Flags	Flag Text	Hour	
2	00:03.0	6	SB_NS	2	South	38.525			5	Friday	=HOUR(A2)	
3	00:22.0	5	SB_MID	2	South	32.31			5	Friday		
4	00:22.0	4	SB_OS	2	South	44.739			5	Friday	0	
5	00:36.0	6	SB_NS	2	South	33.554			5	Friday	0	
6	00:49.1	6	SB_NS	2	South	39.768	12.3	11.847	5	Friday	0	
7	00:52.1	2	NB_MID	1	North	64.623			5	Friday	0	

HOUR(serial_number)

Second create a column with 24 rows, with each cell filled with a single number from 0 to 23. This can be done easily by enter 0 and 1 in first two rows and then apply to 23th row.

From third step, there are two circumstances. First is north direction, and second is south direction.

A	B	C	D	E	F	G	H	I	J	K	L
Date	Lane	Lane Nan	Direction	Direction	Speed (m	Headway	Gap (s)	Flags	Flag Text	Hour	
00:03.0	6	SB_NS	2	South	38.525			5	Friday	0	0
00:22.0	5	SB_MID	2	South	32.31			5	Friday	0	1
00:22.0	4	SB_OS	2	South	44.739			5	Friday	0	2
00:36.0	6	SB_NS	2	South	33.554			5	Friday	0	3
00:49.1	6	SB_NS	2	South	39.768	12.3	11.847	5	Friday	0	4
00:52.1	2	NB_MID	1	North	64.623			5	Friday	0	5
00:55.1	1	NB_NS	1	North	29.205	6.319		5	Friday	0	6
00:58.0	2	NB_MID	1	North	37.283	6.2	6.089	5	Friday	0	7
01:03.0	6	SB_NS	2	South	44.739	14.8	14.575	5	Friday	0	8
01:04.1	2	NB_MID	1	North	41.01	5.155	5.242	5	Friday	0	9
01:05.1	2	NB_MID	1	North	37.283	1.47	0.949	5	Friday	0	10
01:09.0	5	SB_MID	2	South	36.039	47.1	47.017	5	Friday	0	11
01:16.1	6	SB_NS	2	South	36.661	12.3	12.24	5	Friday	0	12
01:40.0	3	NB_OS	1	North	45.361			5	Friday	0	13
01:46.0	2	NB_MID	1	North	38.525	41.3	41.06	5	Friday	0	14
01:48.0	5	SB_MID	2	South	47.224	38.9	38.639	5	Friday	0	15
01:51.0	6	SB_NS	2	South	57.787	35.7	35.438	5	Friday	0	16
######	6	SB_NS	2	South	47.846	4.301	3.334	5	Friday	0	17
######	1	NB_NS	1	North	44.117	61.4	61.086	5	Friday	0	18
01:57.1	6	SB_NS	2	South	49.709	1.957	1.599	5	Friday	0	19
01:58.1	3	NB_OS	1	North	39.146	17.7	17.488	5	Friday	0	20
02:04.0	2	NB_MID	1	North	29.825	18.4	17.744	5	Friday	0	21
######	6	SB_NS	2	South	41.01	37.2	36.997	5	Friday	0	22
02:34.0	4	SB_OS	2	South	45.982	131.9	131.79	5	Friday	0	23
02:36.1	6	SB_NS	2	South	42.253	1.721	1.482	5	Friday	0	

North Direction

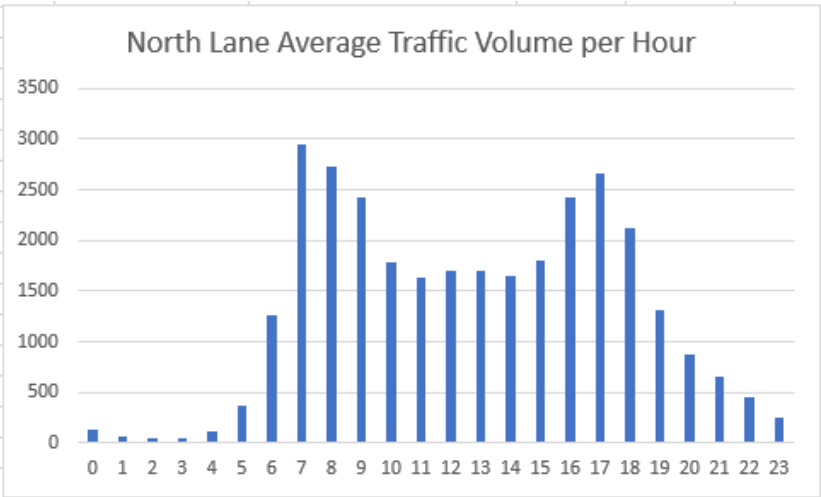
Third use COUNTIFS() function to count the data that satisfied the requirements of north lanes, Tuesday and each hour (from 0 to 23) of a day.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Date	Lane	Lane Nan	Direction	Direction	Speed (m	Headway	Gap (s)	Flags	Flag Text	Hour		
2	00:03.0	6	SB_NS	2	South	38.525			5	Friday	0	0	=COUNTIFS(E:E,"North", I:I, 2, K:K, L2)
3	00:22.0	5	SB_MID	2	South	32.31			5	Friday	0	1	252
4	00:22.0	4	SB_OS	2	South	44.739			5	Friday	0	2	170
5	00:36.0	6	SB_NS	2	South	33.554			5	Friday	0	3	212
6	00:49.1	6	SB_NS	2	South	39.768	12.3	11.847	5	Friday	0	4	433
7	00:52.1	2	NB_MID	1	North	64.623			5	Friday	0	5	1446
8	00:55.1	1	NB_NS	1	North	29.205	6.319		5	Friday	0	6	5040
9	00:58.0	2	NB_MID	1	North	37.283	6.2	6.089	5	Friday	0	7	11806
10	01:03.0	6	SB_NS	2	South	44.739	14.8	14.575	5	Friday	0	8	10908
11	01:04.1	2	NB_MID	1	North	41.01	5.155	5.242	5	Friday	0	9	9721
12	01:05.1	2	NB_MID	1	North	37.283	1.47	0.949	5	Friday	0	10	7107
13	01:09.0	5	SB_MID	2	South	36.039	47.1	47.017	5	Friday	0	11	6537
14	01:16.1	6	SB_NS	2	South	36.661	12.3	12.24	5	Friday	0	12	6795
15	01:40.0	3	NB_OS	1	North	45.361			5	Friday	0	13	6830
16	01:46.0	2	NB_MID	1	North	38.525	41.3	41.06	5	Friday	0	14	6568
17	01:48.0	5	SB_MID	2	South	47.224	38.9	38.639	5	Friday	0	15	7221
18	01:51.0	6	SB_NS	2	South	57.787	35.7	35.438	5	Friday	0	16	9716
19	######	6	SB_NS	2	South	47.846	4.301	3.334	5	Friday	0	17	10623
20	######	1	NB_NS	1	North	44.117	61.4	61.086	5	Friday	0	18	8486
21	01:57.1	6	SB_NS	2	South	49.709	1.957	1.599	5	Friday	0	19	5233
22	01:58.1	3	NB_OS	1	North	39.146	17.7	17.488	5	Friday	0	20	3531
23	02:04.0	2	NB_MID	1	North	29.825	18.4	17.744	5	Friday	0	21	2595
24	######	6	SB_NS	2	South	41.01	37.2	36.997	5	Friday	0	22	1815
25	02:34.0	4	SB_OS	2	South	45.982	131.9	131.79	5	Friday	0	23	1027

Now the current data from M2 to M25 is the total traffic volume of each hour every Tuesday. To obtain the average traffic volume, these data need to be divided by 4.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Date	Lane	Lane Nan	Direction	Direction	Speed (m	Headway	Gap (s)	Flags	Flag Text	Hour			
2	00:03.0	6	SB_NS	2	South	38.525			5	Friday	0	0	506	=M2/4
3	00:22.0	5	SB_MID	2	South	32.31			5	Friday	0	1	252	63
4	00:22.0	4	SB_OS	2	South	44.739			5	Friday	0	2	170	42.5
5	00:36.0	6	SB_NS	2	South	33.554			5	Friday	0	3	212	53
6	00:49.1	6	SB_NS	2	South	39.768	12.3	11.847	5	Friday	0	4	433	108.25
7	00:52.1	2	NB_MID	1	North	64.623			5	Friday	0	5	1446	361.5
8	00:55.1	1	NB_NS	1	North	29.205	6.319		5	Friday	0	6	5040	1260
9	00:58.0	2	NB_MID	1	North	37.283	6.2	6.089	5	Friday	0	7	11806	2951.5
10	01:03.0	6	SB_NS	2	South	44.739	14.8	14.575	5	Friday	0	8	10908	2727
11	01:04.1	2	NB_MID	1	North	41.01	5.155	5.242	5	Friday	0	9	9721	2430.25
12	01:05.1	2	NB_MID	1	North	37.283	1.47	0.949	5	Friday	0	10	7107	1776.75
13	01:09.0	5	SB_MID	2	South	36.039	47.1	47.017	5	Friday	0	11	6537	1634.25
14	01:16.1	6	SB_NS	2	South	36.661	12.3	12.24	5	Friday	0	12	6795	1698.75
15	01:40.0	3	NB_OS	1	North	45.361			5	Friday	0	13	6830	1707.5
16	01:46.0	2	NB_MID	1	North	38.525	41.3	41.06	5	Friday	0	14	6568	1642
17	01:48.0	5	SB_MID	2	South	47.224	38.9	38.639	5	Friday	0	15	7221	1805.25
18	01:51.0	6	SB_NS	2	South	57.787	35.7	35.438	5	Friday	0	16	9716	2429
19	######	6	SB_NS	2	South	47.846	4.301	3.334	5	Friday	0	17	10623	2655.75
20	######	1	NB_NS	1	North	44.117	61.4	61.086	5	Friday	0	18	8486	2121.5
21	01:57.1	6	SB_NS	2	South	49.709	1.957	1.599	5	Friday	0	19	5233	1308.25
22	01:58.1	3	NB_OS	1	North	39.146	17.7	17.488	5	Friday	0	20	3531	882.75
23	02:04.0	2	NB_MID	1	North	29.825	18.4	17.744	5	Friday	0	21	2595	648.75
24	######	6	SB_NS	2	South	41.01	37.2	36.997	5	Friday	0	22	1815	453.75
25	02:34.0	4	SB_OS	2	South	45.982	131.9	131.79	5	Friday	0	23	1027	256.75

Finally, select L column and N column, go to Insert -> Insert Column or Bar Chart -> More Column Charts… -> Clustered Column then select the second chart to obtain the bar plot. The bar plot is:



South Direction

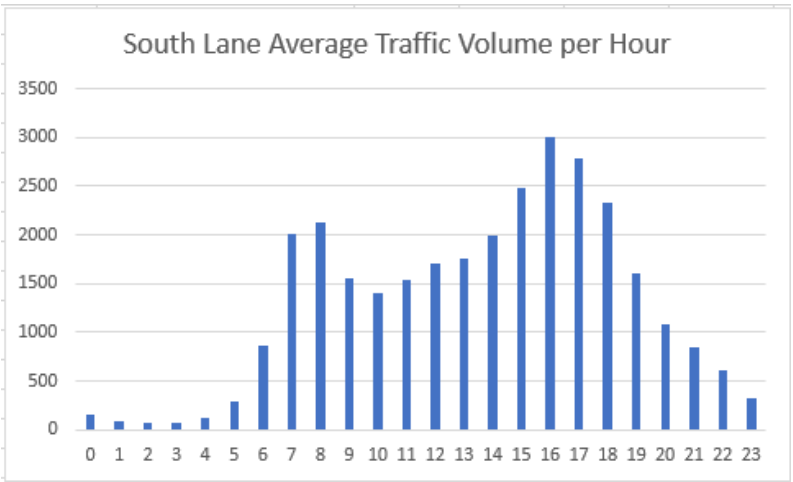
Third use COUNTIFS() function to count the data that satisfied the requirements of south lanes, Tuesday and each hour (from 0 to 23) of a day.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Date	Lane	Lane Nan	Direction	Direction	Speed (m	Headway	Gap (s)	Flags	Flag Text	Hour		
2	00:03.0	6	SB_NS	2	South	38.525			5	Friday	0	0	=COUNTIFS(E:E, "South", I:I, 2, K:K,L2)
3	00:22.0	5	SB_MID	2	South	32.31			5	Friday	0	1	371
4	00:22.0	4	SB_OS	2	South	44.739			5	Friday	0	2	310
5	00:36.0	6	SB_NS	2	South	33.554			5	Friday	0	3	319
6	00:49.1	6	SB_NS	2	South	39.768	12.3	11.847	5	Friday	0	4	482
7	00:52.1	2	NB_MID	1	North	64.623			5	Friday	0	5	1159
8	00:55.1	1	NB_NS	1	North	29.205	6.319		5	Friday	0	6	3449
9	00:58.0	2	NB_MID	1	North	37.283	6.2	6.089	5	Friday	0	7	8060
10	01:03.0	6	SB_NS	2	South	44.739	14.8	14.575	5	Friday	0	8	8540
11	01:04.1	2	NB_MID	1	North	41.01	5.155	5.242	5	Friday	0	9	6193
12	01:05.1	2	NB_MID	1	North	37.283	1.47	0.949	5	Friday	0	10	5608
13	01:09.0	5	SB_MID	2	South	36.039	47.1	47.017	5	Friday	0	11	6174
14	01:16.1	6	SB_NS	2	South	36.661	12.3	12.24	5	Friday	0	12	6837
15	01:40.0	3	NB_OS	1	North	45.361			5	Friday	0	13	6998
16	01:46.0	2	NB_MID	1	North	38.525	41.3	41.06	5	Friday	0	14	7968
17	01:48.0	5	SB_MID	2	South	47.224	38.9	38.639	5	Friday	0	15	9926
18	01:51.0	6	SB_NS	2	South	57.787	35.7	35.438	5	Friday	0	16	12005
19	######	6	SB_NS	2	South	47.846	4.301	3.334	5	Friday	0	17	11176
20	######	1	NB_NS	1	North	44.117	61.4	61.086	5	Friday	0	18	9322
21	01:57.1	6	SB_NS	2	South	49.709	1.957	1.599	5	Friday	0	19	6407
22	01:58.1	3	NB_OS	1	North	39.146	17.7	17.488	5	Friday	0	20	4321
23	02:04.0	2	NB_MID	1	North	29.825	18.4	17.744	5	Friday	0	21	3388
24	######	6	SB_NS	2	South	41.01	37.2	36.997	5	Friday	0	22	2451
25	02:34.0	4	SB_OS	2	South	45.982	131.9	131.79	5	Friday	0	23	1324

Now the current data from M2 to M25 is the total traffic volume of each hour every Tuesday. To obtain the average traffic volume, these data need to be divided by 4.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Date	Lane	Lane Nan	Direction	Direction	Speed (m	Headway	Gap (s)	Flags	Flag Text	Hour			
2	00:03.0	6	SB_NS	2	South	38.525			5	Friday	0	0	651	=M2/4
3	00:22.0	5	SB_MID	2	South	32.31			5	Friday	0	1	371	92.75
4	00:22.0	4	SB_OS	2	South	44.739			5	Friday	0	2	310	77.5
5	00:36.0	6	SB_NS	2	South	33.554			5	Friday	0	3	319	79.75
6	00:49.1	6	SB_NS	2	South	39.768	12.3	11.847	5	Friday	0	4	482	120.5
7	00:52.1	2	NB_MID	1	North	64.623			5	Friday	0	5	1159	289.75
8	00:55.1	1	NB_NS	1	North	29.205	6.319		5	Friday	0	6	3449	862.25
9	00:58.0	2	NB_MID	1	North	37.283	6.2	6.089	5	Friday	0	7	8060	2015
10	01:03.0	6	SB_NS	2	South	44.739	14.8	14.575	5	Friday	0	8	8540	2135
11	01:04.1	2	NB_MID	1	North	41.01	5.155	5.242	5	Friday	0	9	6193	1548.25
12	01:05.1	2	NB_MID	1	North	37.283	1.47	0.949	5	Friday	0	10	5608	1402
13	01:09.0	5	SB_MID	2	South	36.039	47.1	47.017	5	Friday	0	11	6174	1543.5
14	01:16.1	6	SB_NS	2	South	36.661	12.3	12.24	5	Friday	0	12	6837	1709.25
15	01:40.0	3	NB_OS	1	North	45.361			5	Friday	0	13	6998	1749.5
16	01:46.0	2	NB_MID	1	North	38.525	41.3	41.06	5	Friday	0	14	7968	1992
17	01:48.0	5	SB_MID	2	South	47.224	38.9	38.639	5	Friday	0	15	9926	2481.5
18	01:51.0	6	SB_NS	2	South	57.787	35.7	35.438	5	Friday	0	16	12005	3001.25
19	######	6	SB_NS	2	South	47.846	4.301	3.334	5	Friday	0	17	11176	2794
20	######	1	NB_NS	1	North	44.117	61.4	61.086	5	Friday	0	18	9322	2330.5
21	01:57.1	6	SB_NS	2	South	49.709	1.957	1.599	5	Friday	0	19	6407	1601.75
22	01:58.1	3	NB_OS	1	North	39.146	17.7	17.488	5	Friday	0	20	4321	1080.25
23	02:04.0	2	NB_MID	1	North	29.825	18.4	17.744	5	Friday	0	21	3388	847
24	######	6	SB_NS	2	South	41.01	37.2	36.997	5	Friday	0	22	2451	612.75
25	02:34.0	4	SB_OS	2	South	45.982	131.9	131.79	5	Friday	0	23	1324	331

Finally, select L column and N column, go to Insert -> Insert Column or Bar Chart -> More Column Charts… -> Clustered Column then select the second chart to obtain the bar plot. The bar plot is:



Python and Excel Comparison

Similarity

They can all be used for further processing, analysis and visualization of the data. Both have functions already defined and can be used directly. Both can operate on csv files.

Python

Advantage:

1. In addition to some self-contained functions, it is easy to download third-party libraries to make data processing simpler and more diverse (suck as machine learning).
2. More efficient when handling large amounts of data.
3. By writing the code once, repetitive operations can be performed.
4. The code is readable and can be saved, so it can be disseminated and improved.
5. It can be used in multiple platforms.
6. Can read and process many types of files

Disadvantage

1. Relatively difficult to learn.
2. May face various bugs.

Excel

Advantage

1. It has a graphical interface. Process data with a mouse click without writing a lot of code. More intuitive operation.
2. Easy to learn, excel can be used by short learning time.
3. High penetration rate, most computers have Excel.

Disadvantage

1. It gets laggy when processing large amounts of data
2. When performing the same operation on data, the operation needs to be repeated.
3. The process is not recorded after processing.