

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 from sklearn.preprocessing import Imputer
5
6 #southbound 5 minutes aggregation
7 data_south_5 = pd.read_csv('/Users/
artyomkholodkov/Downloads/1014/
aggregated_1014_south_5.csv')
8 data_north_5 = pd.read_csv('/Users/
artyomkholodkov/Downloads/1014/
aggregated_1014_north_5.csv')
9
10 def clean_data(dataset):
11     imputer = Imputer(strategy="median")
12     data_noindex = dataset.drop("TIME", axis
= 1)
13     imputer.fit(data_noindex)
14     filled_data = imputer.transform(
data_noindex)
15     data_transformed = pd.DataFrame(
filled_data, index=dataset.TIME)
16     data_transformed.columns = ["SPEED", "
HEADWAY", "SPACE_HEADWAY", "DENSITY", "FLOW"]
17     return data_transformed
18
19 def select_10_tickers(ts_array):
20     idx = np.round(np.linspace(0, len(
ts_array) - 1, 10)).astype(int)
21     return ts_array[idx]
22
23 def plot_dataset(dataset):
24     for column in dataset.columns:
25         ax=plt.subplot(len(dataset.columns),
1, dataset.columns.get_loc(column) + 1)
26         plt.title(column, y=0.5, loc='right')
27         ax.xaxis_date()
28         ax.xaxis.set_ticks(select_10_tickers(
dataset.index))
```

```
29         plt.plot(dataset.index, np.array(
dataset[column]))
30     plt.show()
31
32
33 from scipy.signal import lfilter
34 def denoise_data(cleaned_dataset):
35     n = 2
36     b = [1.0 / n] * n
37     a = 1
38     filtered_data = lfilter(b,a,
cleaned_dataset.FLOW)
39     return np.array(filtered_data)
40
41 cleaned_data_south_5 = clean_data(
data_south_5)
42 dropped_flow_pd = cleaned_data_south_5.drop([
'FLOW'], axis = 1)
43 denoised_flow_south = denoise_data(
cleaned_data_south_5)
44 dropped_flow_pd["FLOW"] = denoised_flow_south
45 dropped_flow_pd.to_csv('
denoised_data_1015_south.csv')
46
47 cleaned_data_north_5 = clean_data(
data_north_5)
48 dropped_flow_pd_north = cleaned_data_north_5.
drop(['FLOW'], axis = 1)
49 denoised_flow_north = denoise_data(
cleaned_data_north_5)
50 dropped_flow_pd_north["FLOW"] =
denoised_flow_north
51 dropped_flow_pd_north.to_csv('
denoised_data_1015_north.csv')
52
53
54
```