ZONE 1160

```
import pandas as pd
zone_1160_flow_df = pd.read_csv("/home/datascience/Downloads/cleaning1/1160/flow.tsv",
                    na_values=['-'], sep='\t', names = ['f_d1','f_d2','f_d3'])
zone_1160_speed_df = pd.read_csv("/home/datascience/Downloads/cleaning1/1160/speed.tsv",
                    na_values=['-'], sep='\t', names = ['s_d1','s_d2','s_d3'])
zone_1160_occupancy_df =
pd.read_csv("/home/datascience/Downloads/cleaning1/1160/occupancy.tsv",
                    na_values=['-'], sep='\t', names = ['o_d1','o_d2','o_d3'])
zone_1160_d1_vector = pd.concat([zone_1160_flow_df['f_d1'], zone_1160_speed_df['s_d1'],
zone_1160_occupancy_df['o_d1']], axis=1)
def isNaN(x):
    return (x == x) == False
is_1160_f_d1_Nan = ~isNaN(zone_1160_d1_vector['f_d1'])
zone_1160_d1_vector = zone_1160_d1_vector[is_1160_f_d1_Nan]
is_1160_s_d1_Nan = ~isNaN(zone_1160_d1_vector['s_d1'])
zone_1160_d1_vector = zone_1160_d1_vector[is_1160_s_d1_Nan]
is_1160_o_d1_Nan = ~isNaN(zone_1160_d1_vector['o_d1'])
zone_1160_d1_vector = zone_1160_d1_vector[is_1160_o_d1_Nan]
zone_1160_d2_vector = pd.concat([zone_1160_flow_df['f_d2'], zone_1160_speed_df['s_d2'],
zone_1160_occupancy_df['o_d2']], axis=1)
is_1160_f_d2_Nan = ~isNaN(zone_1160_d2_vector['f_d2'])
zone_1160_d2_vector = zone_1160_d2_vector[is_1160_f_d2_Nan]
is_1160_s_d2_Nan = ~isNaN(zone_1160_d2_vector['s_d2'])
zone_1160_d2_vector = zone_1160_d2_vector[is_1160_s_d2_Nan]
is_1160_o_d2_Nan = ~isNaN(zone_1160_d2_vector['o_d2'])
zone_1160_d2_vector = zone_1160_d2_vector[is_1160_o_d2_Nan]
zone_1160_d3_vector = pd.concat([zone_1160_flow_df['f_d3'], zone_1160_speed_df['s_d3'],
zone_1160_occupancy_df['o_d3']], axis=1)
is_1160_f_d3_Nan = ~isNaN(zone_1160_d3_vector['f_d3'])
zone_1160_d3_vector = zone_1160_d3_vector[is_1160_f_d3_Nan]
is_1160_s_d3_Nan = ~isNaN(zone_1160_d3_vector['s_d3'])
zone_1160_d3_vector = zone_1160_d3_vector[is_1160_s_d3_Nan]
is_1160_o_d3_Nan = ~isNaN(zone_1160_d3_vector['o_d3'])
zone_1160_d3_vector = zone_1160_d3_vector[is_1160_o_d3_Nan]
zone_1160_d1_vector.columns = ['flow','speed','occupancy']
zone_1160_d2_vector.columns = ['flow','speed','occupancy']
zone_1160_d3_vector.columns = ['flow','speed','occupancy']
zone_1160_vector =
zone_1160_d1_vector.append(zone_1160_d2_vector.append(zone_1160_d3_vector))
from scipy.stats import multivariate_normal
import numpy as np
mean = np.mean(zone_1160_vector)
```

```
cov = zone_1160_vector.cov()
zone_1160_vector['probability'] = multivariate_normal.pdf(zone_1160_vector, mean, cov)
zone_1160_vector['flow'] = zone_1160_vector['flow'].astype(int)
zone_1160_vector['speed'] = zone_1160_vector['speed'].astype(int)
zone_1160_vector['occupancy'] = zone_1160_vector['occupancy'].astype(int)
zone_1160_vector = zone_1160_vector.reset_index()
is_occ_neg = zone_1160_vector['occupancy'] <0
zone_1160_vector_is_neg_occ = zone_1160_vector[is_occ_neg]
is_speed_neg = zone_1160_vector['speed'] <0
zone_1160_vector_is_neg_speed = zone_1160_vector[is_speed_neg]
is_flow_neg = zone_1160_vector['flow'] <0
zone_1160_vector_is_neg_flow = zone_1160_vector[is_flow_neg]
zone_1160_vector_is_neg =
zone_1160_vector_is_neg_flow.append(zone_1160_vector_is_neg_speed.append(zone_1160_vector_is
_neg_occ))
zone_1160_vector_is_neg = zone_1160_vector_is_neg.drop_duplicates()
zone_1160_vector_is_neg['probability'] = 0
zone_1160_vector = zone_1160_vector.append(zone_1160_vector_is_neg)
zone_1160_vector['index'] = zone_1160_vector.index
zone_1160_vector= zone_1160_vector.drop_duplicates(['index'], take_last = True)
zone_1160_vector.set_index = zone_1160_vector['index']
zone_1160_vector = zone_1160_vector.drop('index', axis =1)
zone_1160_vector = zone_1160_vector.sort_index()
zone_1160_sorted_vector = zone_1160_vector.sort(['probability'])
zone_1160_sorted_vector = zone_1160_sorted_vector.reset_index(drop=True)
```