

```
1) import org.apache.spark.sql.functions.  
2) import org.joda.time.DateTimeFormat  
3) val inputpath = "/users/gokulkrishnan/Downloads/pollution/*" val data  
= sqlContext.read  
4) .format("com.databricks.spark.csv") .option("header", "true")  
5) header.option("delimiter", ",") .option("inferSchema", "true")  
6) .load(inputpath)  
7) data1.toDF().registerTempTable("data1")  
8) select ozone, timestamp from data1  
9) data1.show()  
10) import matplotlib  
11) import matplotlib.pyplot as plt  
12) import seaborn as sns  
13) import stringio  
14) dot.show(p)  
15) img = (stringio; format='svg')  
17) img.getvalue()  
18) print "<!DOCTYPE html>" + img.getvalue()  
19) df = SqlContextSQL("Select ozone, latitude, longitude  
From data group by latitude longitude")  
20) data = df.toPandasC()  
21) value = 'ozone'  
22) x = 'latitude, longitude'
```

```
24) heatmapdata = data.pivot_table(values= value, index= x,
columns= grouping)
you
t1
25) heatmapdata = heatmapdata[0:100]
26) a4-dims = len(heatmap_data.columns), so
27) fig,ax = plt.subplots(figsize=a4-dims)
28) ax.set_title("avg Arrival Delay")
29) sns.heatmap(heatmapdata, ax=ax, annot=True, fmt=".1f")
30) show(ax)
31) data1.registerTempTable("data1")
32) from pyspark.sql import SQLContext
33) df=sqlContext.table("data1")
34) df.show(10)
35) df.count()
36) import os
37) import pandas as pd
38) import glob
39) listf = []
40) for now in glob.glob('/Users/krishna/boundary'
pollution/*'):
    data = pd.read_csv(now)
    list.append(data)
41) list.append(data)
42) data.count()
43) import plotly.plotly as py
```

44) import plotly.graph_objs as go

45) py.signin('gkrishnan', 'race2vHowUkyr2V1p1L')
46) trace=go.Scatter(x=data['timestamp'], y=data['ozone'])

47) layout=dict(title='Time Series plot of ozone', xaxis=dic
title='Timestamp'),
48) yaxis=dict(title='Ozone'),
49) plot=[trace]

50) fig=dict(data=plot, layout=layout)

51) py.plot(fig, filename='line chart')

52) for col in df.columns: df[col]=df[col].astype(str)
53) scl=[{"c": "E5.0", "y": [242, 240, 247]}, {"c": "F1.1", "y": [248, 247, 250]},
[{"c": "C4", "y": [188, 189, 220]}],

54) [{"x": 158, "y": 154, "z": 200}, {"x": 0.8, "y": 117, "z": 177}, {"x": 0.1,

"rgb[84, 39, 143]"}])

55) df['text']=df['ozone']

56) mapplot=dict(type='choropleth', colorscale=scl,
autoColorScale=False,
autocolorscale=False)

57) locationz=df['latitude'].z=df['ozone'].os_typefloat),
locationMode='USA - states', ExtentFilter)

58) locationMode='USA - states', ExtentFilter),
colorscale=dic[("color": "rgb[255, 255, 255]"),

59) marker=dict(title='Million USD'),

width=211,

60) Colorbar=dic[("title": "Million USD")]

63) `show_lakes = True, lakecolor = 'rgb(255, 255, 255)')`

64) `fig = dict(data=mapplot, layout=layout)`

65) `py.iplot(fig, filename='d3-coloropleth-map')`

66) `nis = go.Histogram(x=x)]`

67) `layout = dict(title = "Distribution of ozone", axis=dict(title = 'count'))`

68) `yaxis = dict(title = 'zone'),]`

69) `fig = dict(data=nis, layout=layout)`

70) `py.plot(figu1)`

71) `import plotly.plotly as py`

72) `import plotly.graph_objs as go`

73) `x = data['carbon-monoxide']`

74) `hist1 = [go.Histogram(x=x)]`

75) `layout = dict(title='Distribution of carbon monoxide',`

76) `axis=dict(title='count'), yaxis=dict(title='carbon monoxide'),)`

77) `figu2 = dict(data=his1, layout=layout)`

78 py.plot (fig1h2)

79 import plotly.plotly as py

80) import plotly.graph_objs as go

81) ~~x = data ['gas']~~

81) import numpy as np

82) x = data ['sulfur_dioxide'],

83) his2 = [go.Histogram (x=x)]

84) * layout = dict (title = 'Distribution of Sulfur dioxide',

85) * xaxis = dict (title = 'First count')

86) * axis = dict (title = 'Sulfur dioxide'),

87) fig2 = dict (data=his2, layout=layout)

88) x = data ['nitrogen_dioxide']

89) his4 = [go.Histogram (x=x)]

90) layout = dict (title = 'Distribution of Nitrogen dioxide',

91) xaxis = dict (title = 'Nitrogen dioxide'),

92) fig4 = dict (data=his4, layout=layout)

93) traces3 = go.Scatter (x = data ['Cimestar'], y=

94) yaxis = dict (title = 'Nitrogen dioxide')

95) data ['nitrogen_dioxide']

96) layout = dict (title = 'Time series'

97) plot3 = [traces3, layout = layout = 'Nitrogen dioxide',

98) plot3 = plot3 + fig4

(as) xaxis dict title = 'Timespan'), yaxis = dict title = 'Nitrogen Dioxide',

96 / Fig 4 = dict (data = pbt, layout = layout)

97) From: tweepy-streaming import StreamListener

18) From twelve import Author and 10k

99) from every import strategy

100) access_key = "824796411886188288"

Y11CCP2DQNT61GVS64ZV

8 * 5 Lohmsjor

101) access token = "b85w997f+fvzA64ta0jefj

Mwhj 7DchS8GPaclE9V35>

102) consumer-key = "SArrL053#f" ,
103) consumer-secret = "ewkclhj04c w xBfr.e01ZH19y
Buy7ny6aY6i28d0mAEGvGtrvdy