Zeppelin

Graph1

default

FINISHED

```
import org.apache.spark.sql.functions._
import org.joda.time.format.DateTimeFormat

// Load data - adjust the path to the location of your data
val inputPath =  "/Users/neha/Documents/Capstone/pollution*"
val data1 = sqlContext.read
        .format("com.databricks.spark.csv")
        .option("header", "true") // Use first line of all files as header
        .option("delimiter", ",")
        .option("inferSchema", "true") // Automatically infer data types
        .load(inputPath)
data1.toDF().registerTempTable("data2")
```

```
import org.apache.spark.sql.functions._
import org.joda.time.format.DateTimeFormat
inputPath: String = /Users/neha/Documents/Capstone/pollution*
data1: org.apache.spark.sql.DataFrame = [ozone: int, particullate_matter: int ... 6 more field
s]
warning: there was one deprecation warning; re-run with -deprecation for details
```

Took 46 sec. Last updated by anonymous at April 16 2017, 4:41:08 PM.

FINISHED

```
%pyspark
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns

import StringIO
def show(p):
  img = StringIO.StringIO()
  p.savefig(img, format='svg')
  img.seek(0)
  print "%html " + img.buf
```

Took 24 sec. Last updated by anonymous at April 16 2017, 4:58:14 PM.

FINISHED

```
%pyspark
DF = sqlContext.sql("select avg(ozone) as avg_ozone, avg(particullate_matter) as particulate,
, avg(nitrogen_dioxide) as nitrogen from data2 group by longitude,latitude")
data = DF.toPandas()
```

Took 17 sec. Last updated by anonymous at April 16 2017, 4:58:54 PM.

FINISHED

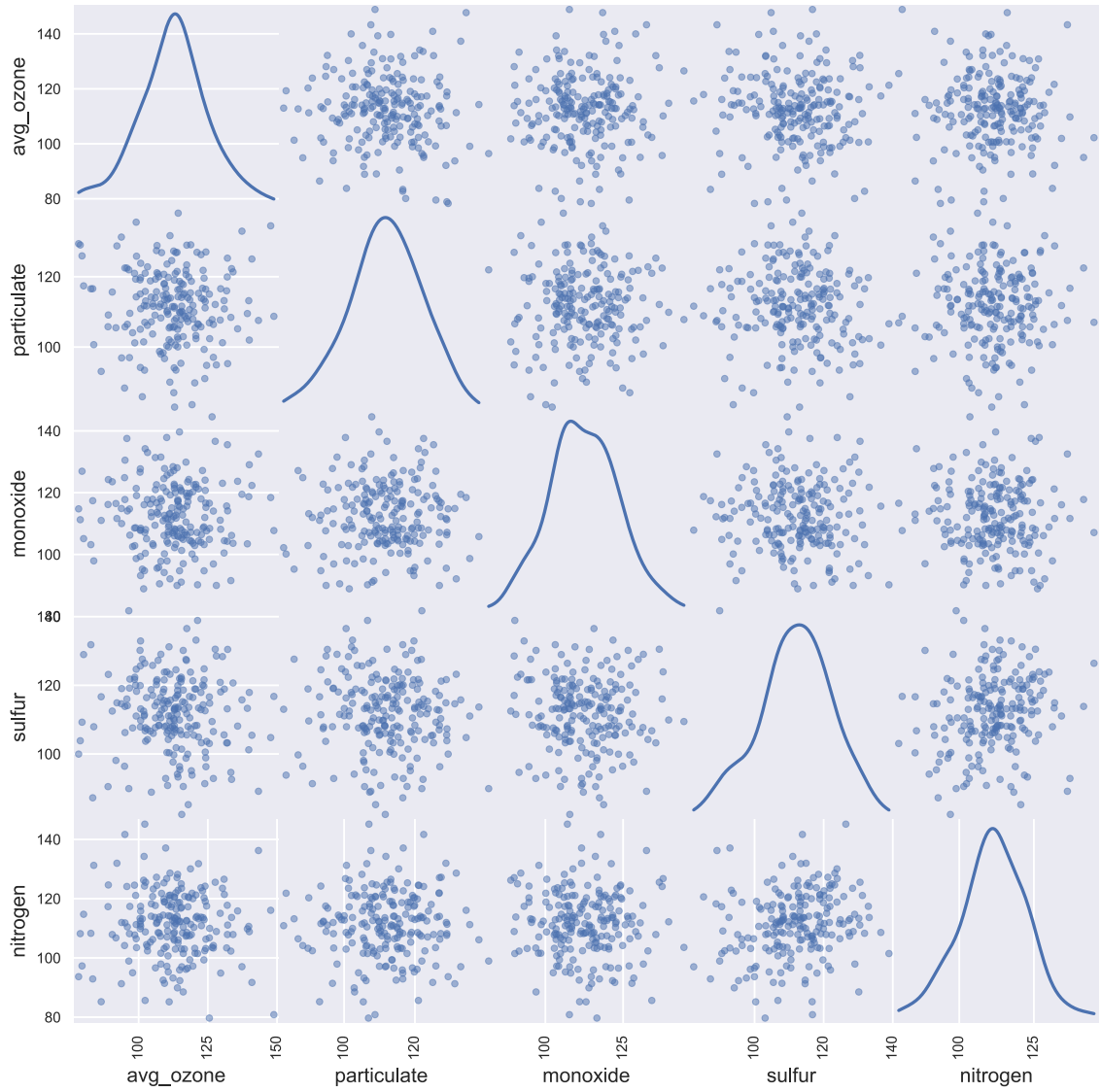
```
%pyspark
```

```
from pandas.tools.plotting import scatter_matrix
scatter_matrix(data, alpha=0.5, figsize=(10,10), diagonal='kde')
show(plt)
```

Zeppelin

Graph1

default



Took 4 sec. Last updated by anonymous at April 16 2017, 4:59:01 PM.

READY