Visualization

December 5, 2017

In [1]:
```
import plotly as py
import plotly.graph_objs as go
import numpy as np
py.offline.init_notebook_mode(connected=True)
```

In [2]:
```
from pyspark.sql.types import StructType, StructField, FloatType, LongType, StringType
from pyspark.shell import spark

feats = []
f = open('features.txt')
for line_num, line in enumerate(f):
    if line_num == 0:
        # Timestamp
        feats.append(StructField(line.strip(), LongType(), True))
    elif line_num == 1:
        # Geohash
        feats.append(StructField(line.strip(), StringType(), True))
    else:
        # Other features
        feats.append(StructField(line.strip(), FloatType(), True))

schema = StructType(feats)
```

```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 2.2.0
      /_/

Using Python version 3.6.3 (default, Oct  6 2017 12:04:38)
SparkSession available as 'spark'.
```

In [3]:
```
df = spark.read.format('csv').option('sep', '\t').schema(schema).load('inputs/mini-sam
```

In [8]:
```
with_prefix_column = df.withColumn("Prefix4",df.Geohash.substr(0,2))
```

```
In [14]: import pyspark.sql.functions as sf

         agg_values = with_prefix_column.groupBy("Prefix4").agg(sf.sum(with_prefix_column.light

         agg_values.select(agg_values.Prefix4,agg_values.num_times_lightning).show(n=5)
```

```
+-------+------------------+
|Prefix4|num_times_lightning|
+-------+------------------+
|     f2|            2080.0|
|     c0|            2994.0|
|     f6|             397.0|
|     cc|            1876.0|
|     bc|            3743.0|
+-------+------------------+
only showing top 5 rows
```

```
In [16]: agg_values.count()
```

```
Out[16]: 77
```

```
In [17]: agg_values.take(1)
```

```
Out[17]: [Row(Prefix4='f2', num_times_lightning=2080.0)]
```

```
In [44]: data = [go.Bar(x=agg_values.toPandas()['Prefix4'],y=agg_values.toPandas()['num_times_l
```

```
In [45]: py.offline.iplot(data, filename="spark/lightning_times_bar.png")
```

```
In [76]: import datetime
         from pyspark.sql.functions import udf

         def conv_to_str_ts(unix_ts):
             str_ts = datetime.datetime.fromtimestamp(
                             int(unix_ts/1000.0)).strftime("%m")
             return str_ts

         udf_myFunction = udf(conv_to_str_ts, StringType())
```

```
In [77]: df_month = df.withColumn("Month", udf_myFunction("Timestamp"))
         df_month.take(1)
```

```
Out[77]: [Row(Timestamp=1430352000000, Geohash='dtb8zh79hs80', geopotential_height_lltw=1729.8
```

```
In [78]: month_avgtemp_avghumidity = df_month.groupBy("Month").agg(sf.avg(df_month.temperature_
                                                sf.avg(df_month.relative_humidity_zerodegc_isother
```

```
In [ ]: month_avgtemp_avghumidity.take(12)
```

```
In [79]: data_temp_humidity = [go.Bar(x=month_avgtemp_avghumidity.toPandas()['Month'],y=month_a
                 go.Bar(x=month_avgtemp_avghumidity.toPandas()['Month'],y=month_avgtemp_avghum

In [80]: py.offline.iplot(data_temp_humidity, filename="spark/month_avgtemp_avghumidity.png")

In [81]: #From the above graph and results:
         #August month has highest temperature all through North America - February has lowest
         #January month has highest humidity - September has lowest

In [87]: df_month_sample = df_month.sample(False, .05)
         df_month_sample.count()

Out[87]: 27311

In [88]: df_month_temp = df_month_sample.select(*df_month_sample.columns)

In [89]: trace = go.Scatter(
             x = df_month_temp.toPandas()['Month'],
             y = df_month_temp.toPandas()['temperature_surface'],
             mode = 'markers'
         )

In [90]: data_scatter_month_temp = [trace]
         py.offline.iplot(data_scatter_month_temp, filename='Scatter_Temperature_Month.png')

In [ ]: #Above scattter plot shows temperature data points in each month in NorthAmerica for s
```