Exercise 12

A simple example of Spark Machine Learning

Prior Knowledge
Exercises 7 and 8

Learning Objectives
Understanding how to use Spark MLLib k-means clustering
Simple machine learning algorithms

Software Requirements
(see separate document for installation of these)

- Apache Spark 3.0.0
- Python 3.8.x
- code text editor or other text editor

Overall plan – finding geographic clusters of SFPD incidents

In a previous exercise, we managed to get all the data together for incidents in SF for 2014. This data included the [lat,long] for each incident.

In this exercise, we will simply provide that data to the K-means clustering algorithm that is built into Apache Spark MLLib.

1. In order to call the kmeans learning phase, we need to create an RDD where each element is a **numpy.array**.

This array will contain the [Y,X] data from the SFPD CSV file.

If you have an RDD with the Y, X values in a dataframe, you can create the array using the following lambda:

```
from numpy import array
geoarray = rdd.map(lambda row: array([row.Y,row.X]))
```

2. Once you have an RDD of this form you can simply train the KMeans model like this:

```
from pyspark.mllib.clustering import KMeans, KMeansModel

numclusters = 5

clusters = KMeans.train(geoarray, numclusters,
     maxIterations=10, initializationMode="random")
```

3. This code will print out the cluster center points in lat,long format:

```
for arr in clusters.centers:
     list = arr.tolist()
     print (str(list[0]) +","+ str(list[1]))
```

4. Put all this together into a Spark program and run it. The output should be a set of data lines like:

```
37.7855185124,-122.408836684
```

You can cut and paste this data into the following webpage which will plot those points on a map:

http://www.hamstermap.com/quickmap.php

You can compare the results to the heat map of crime produced by Trulia here:
http://www.trulia.com/blog/trends/trulia-local/

Note that the clusters represent all incidents not just crime, so we aren't exactly comparing like with like.

Try running this again. Would you expect to see exactly the same results? Compare the new results to the old results on the map.

That's all!