***Chicago Police Department Project Proposal - Checkpoint 4 Queries***

**K-Means Clustering**

from pyspark.ml.clustering import KMeans

from pyspark.ml.evaluation import ClusteringEvaluator

import csv

from pyspark.ml.linalg import Vectors

from pyspark.ml.feature import VectorAssembler

# Loads data.

dataset = spark.read.format("libsvm").load("/FileStore/tables/k\_means\_5.txt")

# Trains a k-means model.

kmeans = KMeans().setK(2).setSeed(1)

model = kmeans.fit(dataset)

# Make predictions

predictions = model.transform(dataset)

# Evaluate clustering by computing Silhouette score

evaluator = ClusteringEvaluator()

silhouette = evaluator.evaluate(predictions)

print("Silhouette with squared euclidean distance = " + str(silhouette))

# Shows the result.

centers = model.clusterCenters()

print("Cluster Centers: ")

for center in centers:

print(center)

**SQL queries for data pre-processing**

select doff.id, doff.gender, doff.rank, doff.race, (extract(year from current\_date) - extract(year from appointed\_date)) as tenure,

(extract(year from current\_date) - doff.birth\_year) as age , a.number\_of\_awards, dok."District\_Number"

from data\_officer doff

inner join data\_officer\_kmeans dok on doff."id"=dok."officer\_id"

inner join (select officer\_id, count(\*) as number\_of\_awards from data\_award group by officer\_id) a on doff.id=a.officer\_id

Note: We created the table *data\_officer\_kmeans* to include the district number associated with the officer id that we got in the checkpoint 2 from the crime database.

**Decision Tree Prediction**

See revised PDF of Databricks Notebook. Cmd 13-Cmd 22. We ran into an error with the display of the resulted prediction from the decision tree model.