```
""""""

Walter Stevens

Linear regression exampleus using Spark datasets

23/9/2017

"""""""""

from __future__ import print_function

from pyspark.ml.regression import LinearRegression

from pyspark.sql import SparkSession
# later versions of Spark use Sessions, not Contexts

from pyspark.ml.linalg import Vectors

if __name__ == "__main__":

    # Create a SparkSession (Note, the config section is only for Windows!)
    spark = SparkSession.builder.config("spark.sql.warehouse.dir", "file:///C:/temp").appName("LinearRegression").getOrCreate()

    # Load up our data and convert it to the format MLLib expects.
    inputLines = spark.sparkContext.textFile("regression.txt")
    data = inputLines.map(lambda x: x.split(",")).map(lambda x: (float(x[0]), Vectors.dense(float(x[1]))))

    # Convert this RDD to a DataFrame
```

```
colNames = ["label", "features"]
df = data.toDF(colNames)

# Note, there are lots of cases where you can avoid going from an RDD to a DataFrame.
# Perhaps you're importing data from a real database. Or you are using structured streaming
# to get your data.

# Let's split our data into training data and testing data
trainTest = df.randomSplit([0.5, 0.5])
trainingDF = trainTest[0]
testDF = trainTest[1]

# Now create our linear regression model
lir = LinearRegression(maxIter=10, regParam=0.3, elasticNetParam=0.8)

# Train the model using our training data
model = lir.fit(trainingDF)

# Now see if we can predict values in our test data.
# Generate predictions using our linear regression model for all features in our
# test dataframe:
fullPredictions = model.transform(testDF).cache()

# Extract the predictions and the "known" correct labels.
predictions = fullPredictions.select("prediction").rdd.map(lambda x: x[0])
labels = fullPredictions.select("label").rdd.map(lambda x: x[0])

# Zip them together
predictionAndLabel = predictions.zip(labels).collect()

# Print out the predicted and actual values for each point
```

```
for prediction in predictionAndLabel:
  print(prediction)


# Stop the session
spark.stop()
```