Motivation

Relation between admissions criteria and chance of admit.

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
0	1	337	118	4	4.5	4.5	9.65	1	0.92
1	2	324	107	4	4.0	4.5	8.87	1	0.76
2	3	316	104	3	3.0	3.5	8.00	1	0.72
3	4	322	110	3	3.5	2.5	8.67	1	0.80
4	5	314	103	2	2.0	3.0	8.21	0	0.65
5	6	330	115	5	4.5	3.0	9.34	1	0.90

=

	GRE	TOEFL	Uni_Rank	SOP	LOR	CGPA	Research	Chance_of_Admit
0	337	118	4	4.5	4.5	9.65	1	0.92
1	324	107	4	4.0	4.5	8.87	1	0.76
2	316	104	3	3.0	3.5	8.00	1	0.72
3	322	110	3	3.5	2.5	8.67	1	0.80
4	314	103	2	2.0	3.0	8.21	0	0.65
5	330	115	5	4.5	3.0	9.34	1	0.90

* change column names

```
# initialize spark environment
conf = pyspark.SparkConf().setAppName('odl').setMaster('local')
sc = pyspark.SparkContext.getOrCreate(conf=conf) # gets existing context otherwise creates new
sqlc = pyspark.sql.SQLContext(sc)
```

```
from pyspark.ml.linalg import Vectors
from pyspark.ml.feature import VectorAssembler

# vectorize the data frame features
assembler = VectorAssembler(
    inputCols=df.columns[:7],
    outputCol="features")
trainingVDF = assembler.transform(trainingDF)
validationVDF = assembler.transform(validationDF)

# view the vectorized data frames
print("Assembled data frame columns to vector column 'features'")
trainingVDF.select("features").show(truncate=False)
validationVDF.select("features").show(truncate=False)
```

```
Assembled data frame columns to vector column 'features'
+---------------------------------+
|features                         |
+---------------------------------+
|[290.0,104.0,4.0,2.0,2.5,7.46,0.0]|
|[294.0,93.0,1.0,1.5,2.0,7.36,0.0] |
|[294.0,95.0,1.0,1.5,1.5,7.64,0.0] |
|[295.0,93.0,1.0,2.0,2.0,7.2,0.0]  |
|[295.0,99.0,1.0,2.0,1.5,7.57,0.0] |
|[295.0,101.0,2.0,2.5,2.0,7.86,0.0]|
|[296.0,95.0,2.0,3.0,2.0,7.54,1.0] |
|[296.0,97.0,2.0,1.5,2.0,7.8,0.0]  |
|[296.0,99.0,2.0,2.5,2.5,8.03,0.0] |
|[296.0,99.0,2.0,3.0,3.5,7.28,0.0] |
|[297.0,96.0,2.0,2.5,2.0,7.43,0.0] |
|[297.0,98.0,2.0,2.5,3.0,7.67,0.0] |
|[297.0,99.0,4.0,3.0,3.5,7.81,0.0] |
|[297.0,100.0,1.0,1.5,2.0,7.9,0.0] |
|[297.0,101.0,3.0,2.0,4.0,7.67,1.0]|
|[298.0,98.0,2.0,4.0,3.0,8.03,0.0] |
|[298.0,99.0,2.0,4.0,2.0,7.6,0.0]  |
|[298.0,101.0,2.0,1.5,2.0,7.86,0.0]|
|[298.0,101.0,4.0,2.5,4.5,7.69,1.0]|
|[299.0,94.0,1.0,1.0,1.0,7.34,0.0] |
+---------------------------------+
only showing top 20 rows
```

Code Snippets

Result

```
# performance on validation set
from pyspark.ml.evaluation import RegressionEvaluator

predictionsAndLabelsDF = lrModel.transform(validationVDF)
eval = RegressionEvaluator()
print(eval.setMetricName("rmse").evaluate(predictionsAndLabelsDF))
print(eval.setMetricName("r2").evaluate(predictionsAndLabelsDF))
```

```
0.06826302980488663
0.7893919278782331
```