Feature Vectors

Lesson Objectives

- After completing this lesson, you should be able to:
  - Understand how feature vectors fit into the APIs for Spark's MLlib and spark.ml libraries
  - Assemble feature vectors
  - Extract specific dimensions from feature vectors

Creating feature vectors (1)

The output of your ETL process might be a DataFrame with various columns. For example, you might want to try to predict churn based on number of sessions, revenue, and recency:

churn	sessions	revenue	recency
1	20	61.24	103
1	8	80.64	23
0	4	100.94	42
0	8	99.48	26
1	17	120.56	47

Creating feature vectors (2)

```
from collections import namedtuple
Customer = namedtuple('Customer', ['churn','sessions','revenue','recency'])
```

```
customers = sc.parallelize([Customer(1, 20, 61.24, 103),
                            Customer(1, 8, 80.64, 23),
                            Customer(0, 4, 100.94, 42),
                            Customer(0, 8, 99.48, 26),
                            Customer(1, 17, 120.56, 47)]).toDF()
```

Creating feature vectors (3)

```
assembler = VectorAssembler().setInputCols(["sessions", "revenue", "recency"]).setOutputCol("features")
```

```
dfWithFeatures = assembler.transform(customers)
```

```
from pyspark.ml.feature import VectorSlicer
slicer = VectorSlicer().setInputCol("features").setOutputCol("some_features")
```

```
slicer.setIndices([0, 1]).transform(dfWithFeatures).show()
```

```
+-----+--------+-------+-------+-----------------+-------------+
|churn|sessions|revenue|recency|         features|some_features|
+-----+--------+-------+-------+-----------------+-------------+
|    1|      20|  61.24|    103|[20.0,61.24,103.0]| [20.0,61.24]|
|    1|       8|  80.64|     23|  [8.0,80.64,23.0]|  [8.0,80.64]|
|    0|       4| 100.94|     42| [4.0,100.94,42.0]| [4.0,100.94]|
|    0|       8|  99.48|     26|  [8.0,99.48,26.0]|  [8.0,99.48]|
|    1|      17| 120.56|     47|[17.0,120.56,47.0]|[17.0,120.56]|
+-----+--------+-------+-------+-----------------+-------------+
```

Lesson Summary

- Having completed this lesson, you should be able to:
  - Understand how feature vectors fit into the APIs for Spark's MLlib and spark.ml libraries
  - Assemble feature vectors
  - Extract specific dimensions from feature vectors