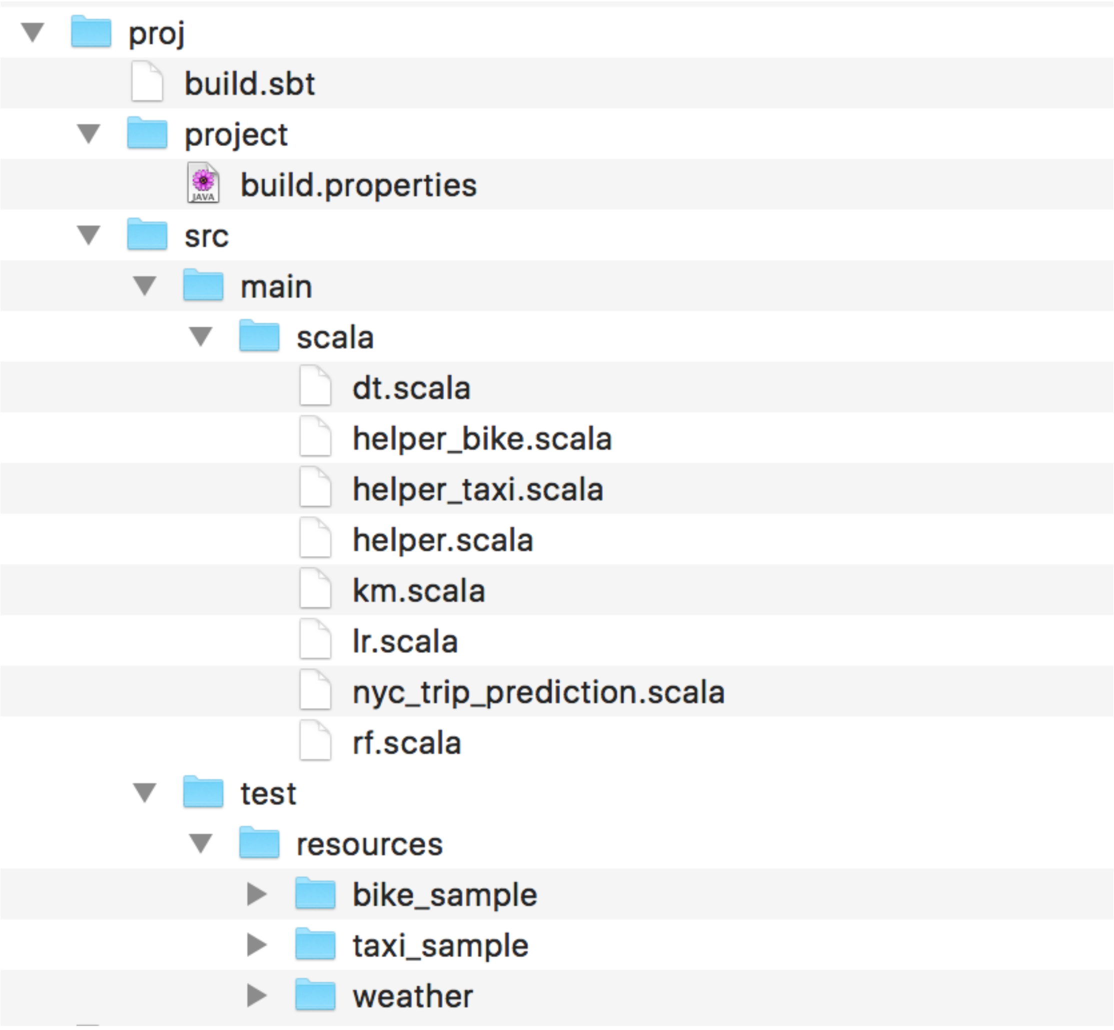
## Hierarchy of the project directory.



```
▼ proj
    build.sbt
  ▼ project
      build.properties
  ▼ src
    ▼ main
      ▼ scala
          dt.scala
          helper_bike.scala
          helper_taxi.scala
          helper.scala
          km.scala
          lr.scala
          nyc_trip_prediction.scala
          rf.scala
    ▼ test
      ▼ resources
        ▶ bike_sample
        ▶ taxi_sample
        ▶ weather
```

## Description of the files:

build script: **build.sbt**

source code:

**nyc_trip_prediction.scala** Main program. Parse the input arguments and determine which mode and method should be used.

**helper.scala** Implementation of some generalized helper functions for data parsing, feature extraction and feature construction.

**helper_bike.scala** Implementation of data parsing, data filtering, feature extraction and feature construction for bike data.

**helper_taxi.scala** Implementation of data parsing, data filtering, feature extraction and feature construction for taxi data.

**dt.scala** Use decision tree regression algorithm to analyze taxi/bike data and save the best model (in training mode), or use the saved model to test the taxi/bike data and output mean squared error (in test mode).

**rf.scala** Use random forest regression algorithm to analyze taxi/bike data and save the best model (in training mode), or use the saved model to test the taxi/bike data and output mean squared error (in test mode).

**lr.scala** Use linear regression algorithm to analyze taxi/bike data and save the best model (in training mode), or use the saved model to test the taxi/bike data and output mean squared error (in test mode).

**km.scala** Use kmeans algorithm to find the best number of centers when clustering the coordinates, and use them in the regression algorithm as features (in training mode), or use the saved model to assign the taxi/bike data coordinates to centers, and use them in the regression algorithm as features (in test mode).

Dataset (sample for the pilot run; can be applied to full profiled dataset):

**weather_sample** Sample weather data which are profiled from full dataset to do a pilot run of the algorithm.

**bike_sample** Sample bike data which are profiled from full dataset to do a pilot run of the algorithm.

**taxi_sample** Sample taxi data which are profiled from full dataset to do a pilot run of the algorithm.

## How to run the code:

```
cd proj
```

```
sbt package
```

```
spark-submit --class NYC_Trip_Prediction <jar> <weather_data_directory> <trip_data_directory> <model_directory> <kmeans_directory> <Taxi|Bike> <DT|RF|LR> <Training|Test>
```

Where <jar> is the location of .jar file, <weather_data_directory> is the location of weather data, <trip_data_directory> is the location of trip data (either taxi or bike), <model_directory> is the location of the regression model (the directory you want to save when in the training mode, or the directory you want to load when in the test mode), <kmeans_directory> is the location of the kmeans model (the directory you want to save when in the training mode, or the directory you want to load when in the test mode), <Taxi|Bike> is the identification of whether you want to use taxi or bike data (ignore the cases), <DT|RF|LR> is the identification of whether you want to use decision tree (DT), random forest (RF), or linear regression (LR) model (ignore the cases), <Training|Test> is the indication of whether this is training mode or test mode (ignore the cases).

A sample run can be like one of these:

```
spark-submit --class NYC_Trip_Prediction target/scala-2.10/nyc-trip-prediction-bdad-final-project_2.10-0.4.jar proj/weather proj/taxi_sample proj/taxi_dt proj/kmeans_taxi_dt taxi dt training
```

```
spark-submit --class NYC_Trip_Prediction target/scala-2.10/nyc-trip-prediction-bdad-final-project_2.10-0.4.jar proj/weather proj/taxi_sample proj/taxi_dt proj/kmeans_taxi_dt taxi dt test
```

The above commands are run in dumbo.

“target/scala-2.10/nyc-trip-prediction-bdad-final-project_2.10-0.4.jar” are generated after command “sbt package” is run.

“proj/weather_sample”, “proj/bike_sample” and “proj/taxi_sample” are the sample data of citibike and taxi and they reside in HDFS.