PREDICTION & ROUTING

Part 1: Prediction

Build a model to predict flight delays in Map-Reduce.

- Your program takes as input 36 historical files. Each file is a month of data. You can use these to build a model.
- You program also take a single test file. This represents all the future flights we want to predict.
- Output a file containing predictions in the format [FL_NUM]_[FL_DATE]_[CRS_DEP_TIME], logical. The first column uniquely identifies a flight and the second is TRUE if the flight will be late.
- The choice of predictive model is open; you will be graded on the accuracy of your method as well as execution time. One possible choice is to use the Random Forest algorithm.
- Input data is to be found in bucked s3://mrclassvitek, in folders a6history and a6test. The folder a6validate contains a file that has the correct answers for most flights, use it to compute the confusion matrix.
- As a measure of accuracy, use the sum of the percentage of on-time flights misclassified as delayed and the percentage of delayed flights misclassified as on-time.
    -
    - Some hints for using random forests:
    - split the data and build models for subsets of the entire data set.
    - Recode the data so that the type of each column has at most 50 different values. In R, they should be factors.
    - Delete columns that are not usable for predictions.
    - Synthesize features that you think make sense. For example you could create a column labelled "Holidays" and it would be true when a flight is close to Christmas, New Year, and Thanksgiving.
- A flight is delayed is ARR_DELAY > 0.
- Your solution should be parallel and efficient.

Prediction report should include:

- Strategy for modeling and prediction (algorithm, etc).
- Execution time to build model and predict on test input.
- Your confusion matrix.

Part 2: Routing

Write a program using Map-Reduce to generate predicted optimal two-flight routes.

- Your program takes 36 history files to build a model, one single test file that contains all scheduled flights for the next year, to create itineraries, and one request file in the following format year, month, day, origin, destination, ignore.
- For each request, your program should propose an itinerary, written flight_num, flight_num, duration.
- The output of the program is scored as follows: sum the durations in hours. Add 100 hours for each missed connection.
- Connections must have at least 30 minutes and no more than one hour on the same carrier.
- The data is in the bucket s3://mrclassvitek, in folders a7history, a7test, a7request and a7validate. The last directory contains a file which lists missed connections, use it for scoring.

The Routing report should include:

- Details about the implementation.
- A a study of the performance and accuracy of the solution.
- A detailed breakdown of what each team member worked on should be added.
- Grading will include the quality of the report and the quality of the code, and will include an extra "code walk" grade.

Your submission should include no binaries except two PDF reports. It should include scripts to build your project and to run it both locally and on EMR.