Flight Delay prediction

My goal:

1. Understand what’s the driven factors that cause delay and how can we improve
2. Build a prototype model to predict delay

Steps I took

1. read in data and take a quick glimpse

2. Generate new features

3. Regularized mean encode the Categorical variables

4. Split data into train, validation and test set

5. Train a base model and tune Hyper parameter to find the best model

6.derive insights from the best model

1. Read in data and take a glimpse

Time related: MONTH, DAY_OF_WEEK, FL_DATE (flight date), CRS_DEP_TIME (departure time), CRS_ELAPSED_TIME

Flight related: UNIQUE_CARRIER, FL_NUM (flight number)

Geo related: ORIGIN (airport code), ORIGIN_CITY_NAME, DEST (airport code), DEST_CITY_NAME, D ISTANCE (miles between origin and destination)

Response: ARR_DEL15 (arrival delay greater than 15 minutes — the target)

Duration: 2016-04-01 to 2017-02-28

Number of city: 308

1. Read in data and take a glimpse

- 5129354 entries
- 1.3% of the response variable columns (delayed 15 min or not)

  is NA → drop it

- Not delayed: 4147263; delayed: 911071

Thoughts before start

What Model should I use?

- Prediction on structured data
- Trade off between interpretability and flexibility
- Robustness

>> Random Forest

- Built in function for interpretation
- Can model non-linear decision boundaries

Which metric should I use?

- Log loss, AUC, accuracy, precision, recall?
- Is the data balanced?
- What do I care?

>> precision

- I care false negative more than false positive

2. Generate new features

External

- holiday and holiday week: boolean, indicate whether that date or week is holiday or not (it's reasonable to assume that during holidays, there should be more flights to accommodate traveler's demand)

- weather

- population of the city: indirectly indicate the size of the airport

Internal

- day of month: to capture monthly seasonality

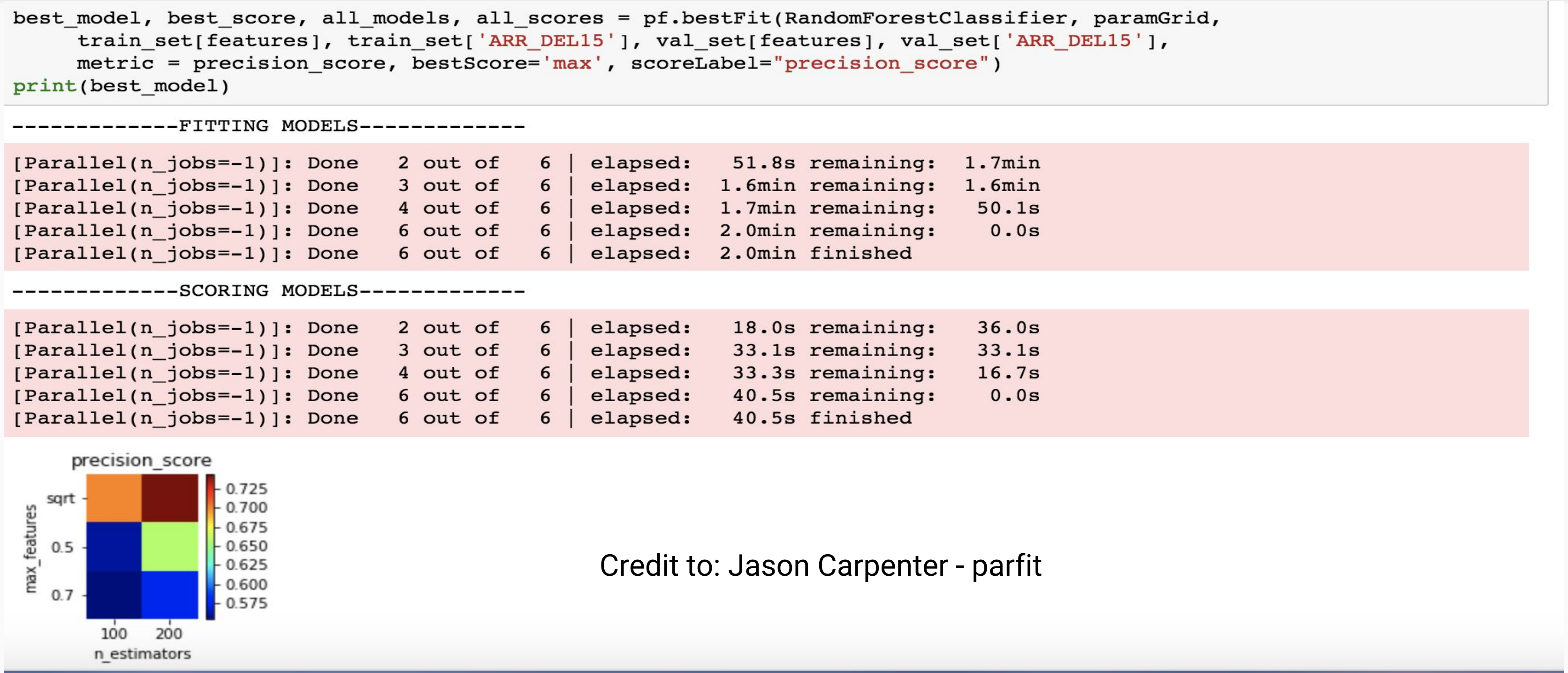
Population data:https://simplemaps.com/data/us-cities

3. Regularized mean encode the Categorical variables

- the point of mean encoding is to derive better meaning from the feature relative to response

- cat_col = ['MONTH', 'DAY_OF_WEEK', 'UNIQUE_CARRIER', 'FL_NUM', 'ORIGIN', 'DEST', 'DEST_CITY_NAME', 'DAY_OF_MONTH', 'DEST_CITY', 'DEST_STATE', 'ORIG_CITY', 'ORIG_STATE']

4. Split data, train a base model, and grid search

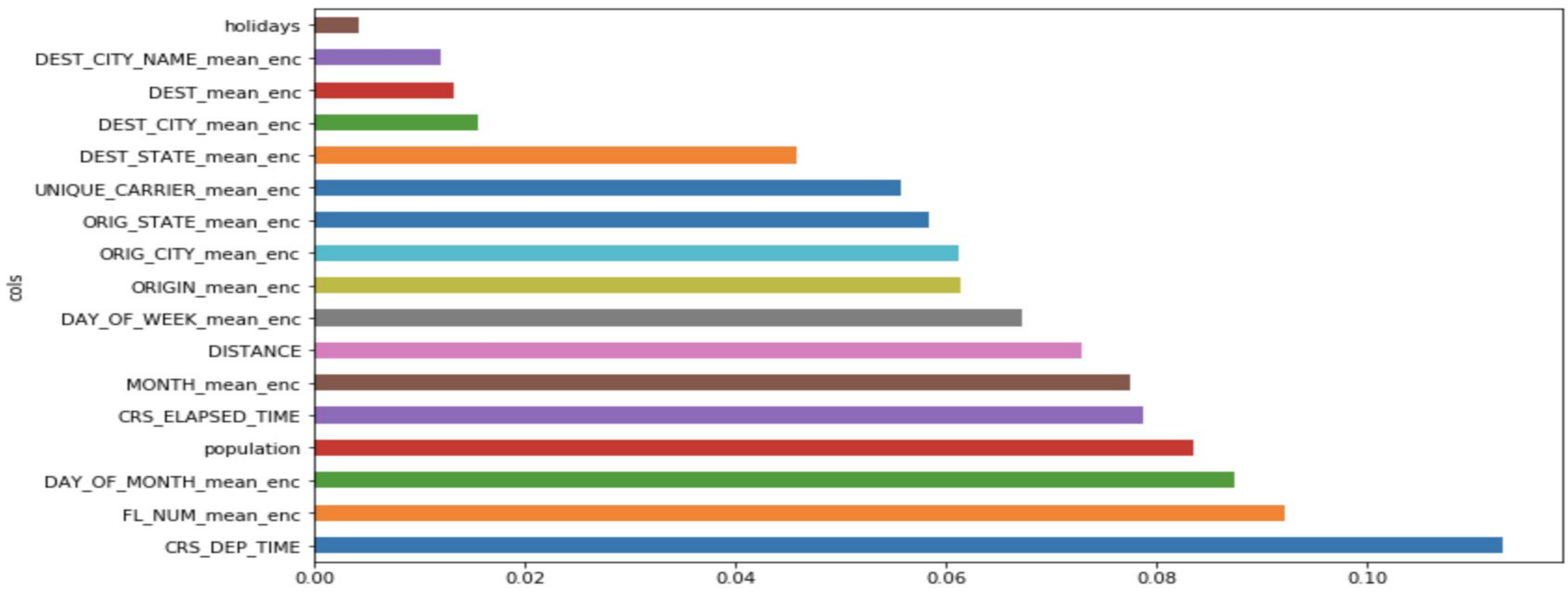


```
best_model, best_score, all_models, all_scores = pf.bestFit(RandomForestClassifier, paramGrid,
      train_set[features], train_set['ARR_DEL15'], val_set[features], val_set['ARR_DEL15'],
      metric = precision_score, bestScore='max', scoreLabel="precision_score")
print(best_model)
```

```
-------------FITTING MODELS-------------
[Parallel(n_jobs=-1)]: Done   2 out of   6 | elapsed:   51.8s remaining:  1.7min
[Parallel(n_jobs=-1)]: Done   3 out of   6 | elapsed:  1.6min remaining:  1.6min
[Parallel(n_jobs=-1)]: Done   4 out of   6 | elapsed:  1.7min remaining:   50.1s
[Parallel(n_jobs=-1)]: Done   6 out of   6 | elapsed:  2.0min remaining:    0.0s
[Parallel(n_jobs=-1)]: Done   6 out of   6 | elapsed:  2.0min finished
-------------SCORING MODELS-------------
[Parallel(n_jobs=-1)]: Done   2 out of   6 | elapsed:   18.0s remaining:   36.0s
[Parallel(n_jobs=-1)]: Done   3 out of   6 | elapsed:   33.1s remaining:   33.1s
[Parallel(n_jobs=-1)]: Done   4 out of   6 | elapsed:   33.3s remaining:   16.7s
[Parallel(n_jobs=-1)]: Done   6 out of   6 | elapsed:   40.5s remaining:    0.0s
[Parallel(n_jobs=-1)]: Done   6 out of   6 | elapsed:   40.5s finished
```

Credit to: Jason Carpenter - parfit

5. Derive insights from the model - Feature importance



5. Derive insights from the model
- Feature importance

Lessons learnt:

- holiday is not the most important reasons - it didn't add pressure enough to collapse the traffic

- departure time ranks highest - rush hour is an important factor

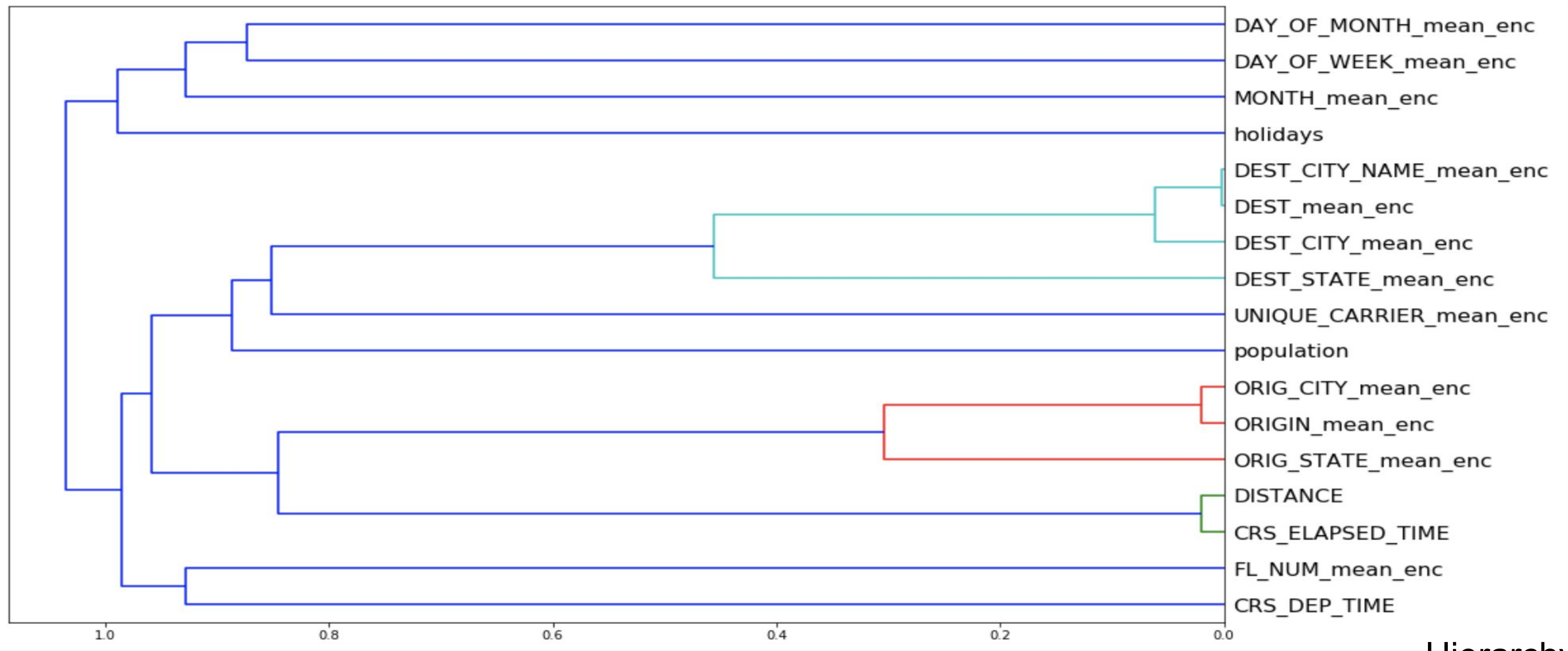
- flight number also ranks high - flight route is an important reasons

- population indicates the popularity and potential volume of the airport

To sum up, 3 type of factors matter:

1. time related: hour of the day
2. geo related: population
3. Flight related: flight route

5. Derive insights from the model
- Similar features



Hierarchy correlation

5. Derive insights from the model
- Similar features

Lessons learnt:

- important time or geo related features tend to correlate with each other

- The interpretability of the model is high

What can we do to improve delay?

1. As airport operator
   - Consider a new runway? (be careful about the huge cost and marginal benefit)
   - Command tower operation efficiency during rush hour
   - Productionize individual model for individual flights as a data product to inform traveler -- information is money!!

2. As a traveler

   - Skip rush hour for “hot” flight route