Airport Delays

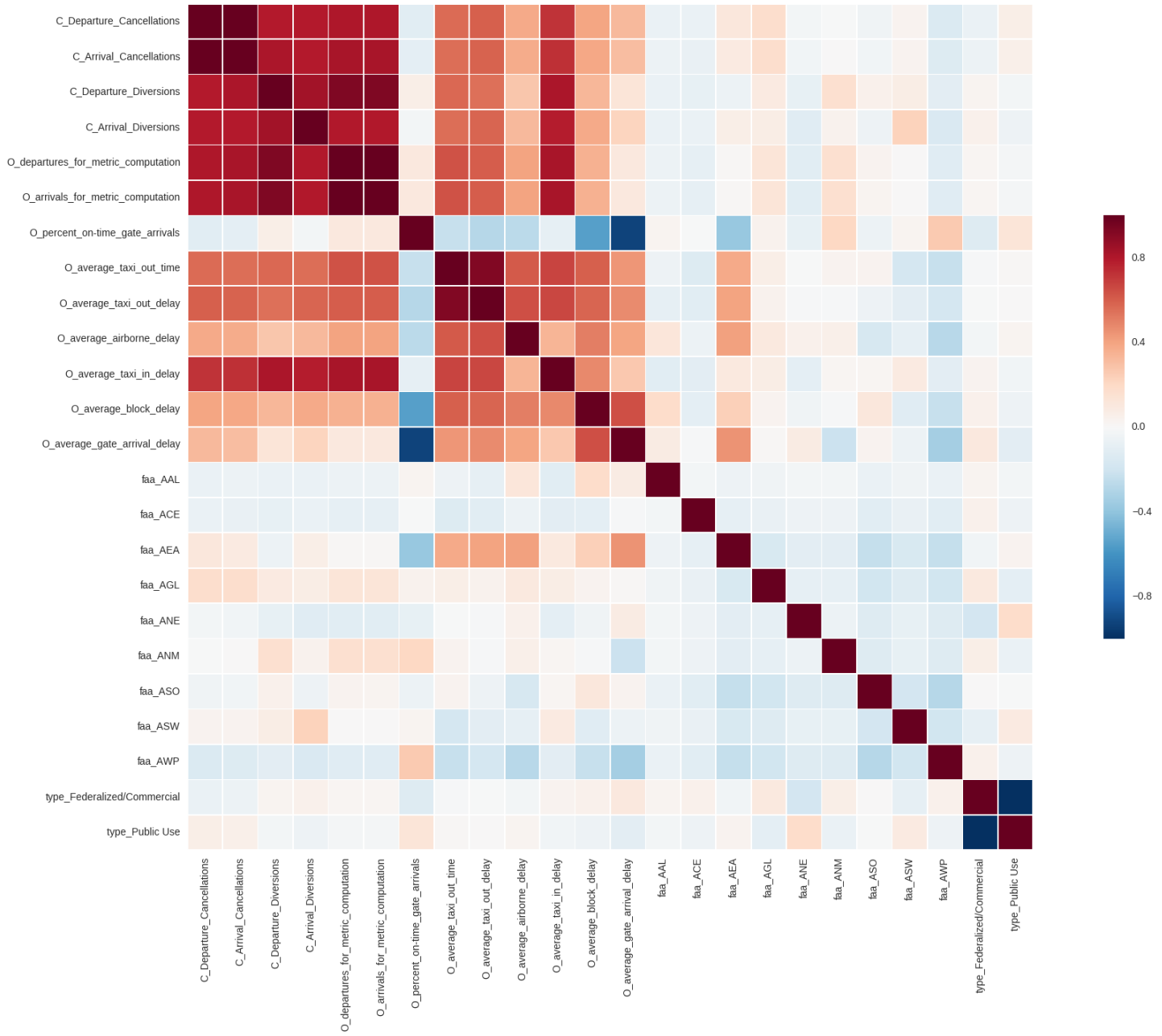
In this project we are commissioned by FAA to analyse the operations and delays of nations airports. We are given three CSVs containing information about airports, airport delays and airport operations. Our first task is to convert these text file to SQL so they can be accessed easier in future. So we import those files with pandas and write those pandas to sql:

```
con = sqlite3.connect("airports.sqlite")
operations=pd.read_csv('../assets/Airport_operations.csv')
cancellations=pd.read_csv('../assets/airport_cancellations.csv')
airports=pd.read_csv('../assets/airports.csv')
airports.to_sql('airports', con, if_exists='replace', index=False)
operations.to_sql('operations', con, if_exists='replace', index=False)
cancellations.to_sql('cancellations', con, if_exists='replace', index=False)
```

Now we must select those airports which we have their cancellations and operations which we can get from sql like this:

```
SELECT
    *
FROM
    (SELECT
        airports.'A_LocID',
        airports.'A_FAA_REGION',
        airports.'A_AP_Type',
        cancellations.*
    FROM
        airports
    JOIN cancellations ON airports.A_LocID = cancellations.C_Airport) AS airport_cancellations
        JOIN
    operations ON airport_cancellations.A_LocID = operations.O_airport
        AND airport_cancellations.C_Year = operations.O_year
```

Now that we have our data we can do some explorations and remove some redundant and useless fields. For this project I decided to choose airport departure delays as my Y. Usually people dont care about any other delays as long as the fly on time and airport can enhance their operations to reduce this delay. Also gate departure has very high correlation with aiport departure so I decided to remove that column which can also be used as its own result.



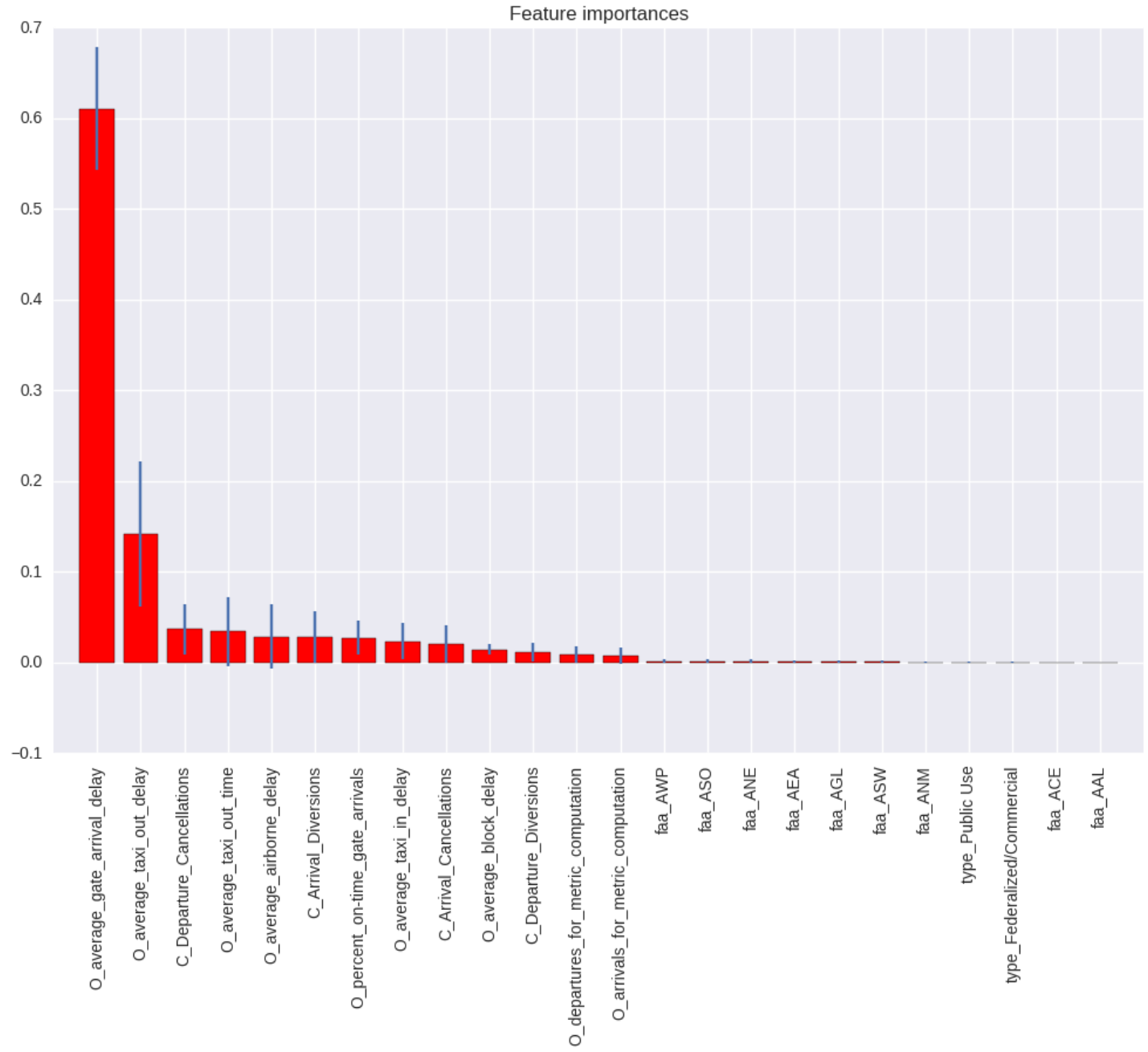
Now that we have our X we can do some modeling to see If we can predict the results and see which feature are the most important.

After Running Random forest in grid search and cross validating the results we have:

```
Best Model after Grid Search:
RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=None,
           max_features='auto', max_leaf_nodes=None, min_samples_leaf=1,
           min_samples_split=2, min_weight_fraction_leaf=0.0,
           n_estimators=100, n_jobs=-1, oob_score=False, random_state=None,
           verbose=0, warm_start=False)
Mean score of the model is: 0.827876425941
Explained variance regression score: 0.88452915113944242
Mean Absolute Error: 1.33042857143
```

And if we run the test data set we have:

```
Test Data score:  0.980106022086
Explained variance regression score: 0.98013698987102305
Mean Absolute Error: 0.450162083333
```



Here we can see most important features which have the most effect on airport departure.

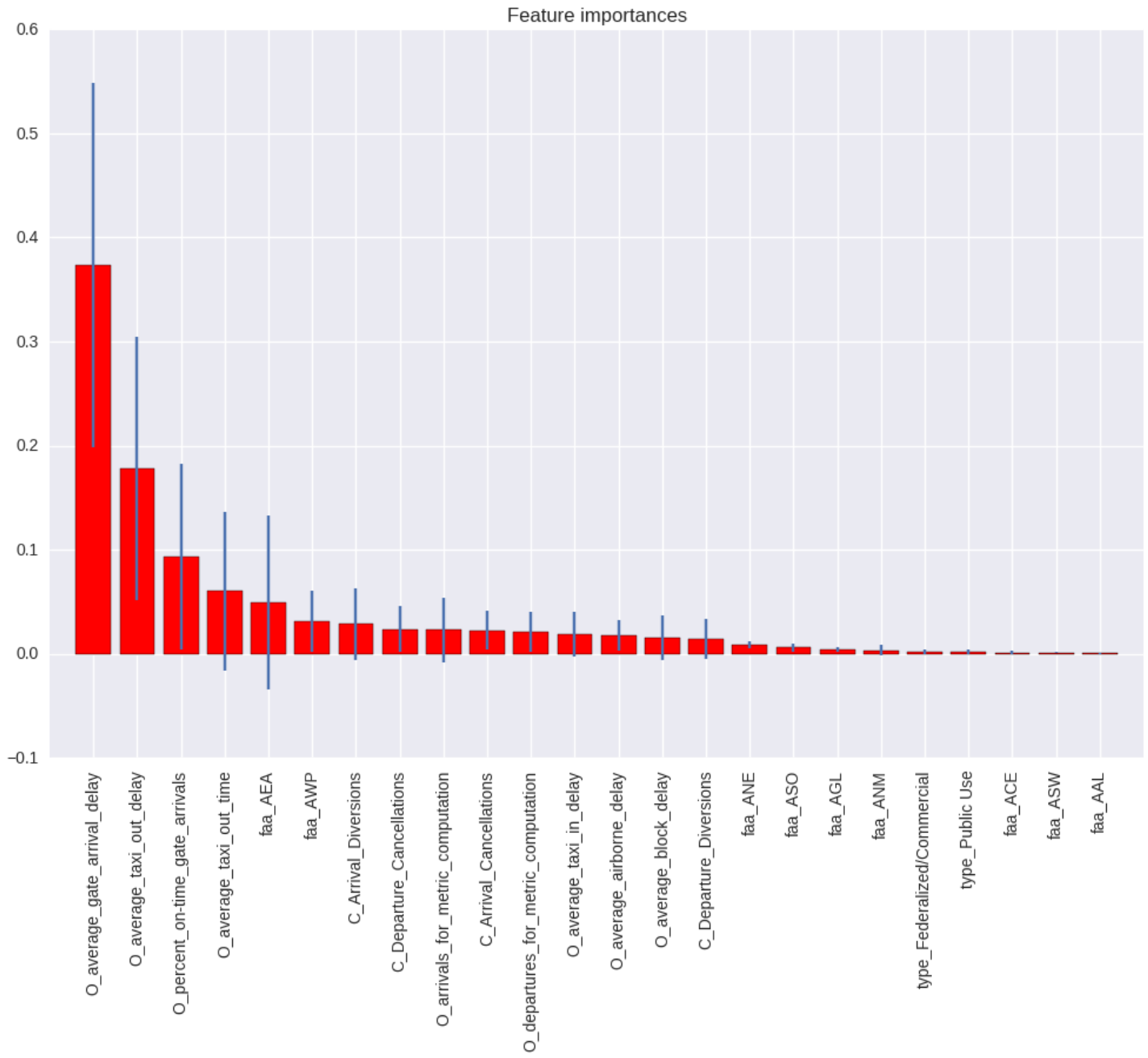
We can repet the above process with Extra tree to confirm our results:

```
Best Model after Grid Search:
ExtraTreesRegressor(bootstrap=False, criterion='mse', max_depth=None,
          max_features='auto', max_leaf_nodes=None, min_samples_leaf=1,
          min_samples_split=2, min_weight_fraction_leaf=0.0,
          n_estimators=100, n_jobs=-1, oob_score=False, random_state=None,
          verbose=0, warm_start=False)
```

```
Mean score of the model is: 0.874522425218
('Explained variance regression score: ', 0.90260288920631238)
Mean Absolute Error: 1.08093642857
```

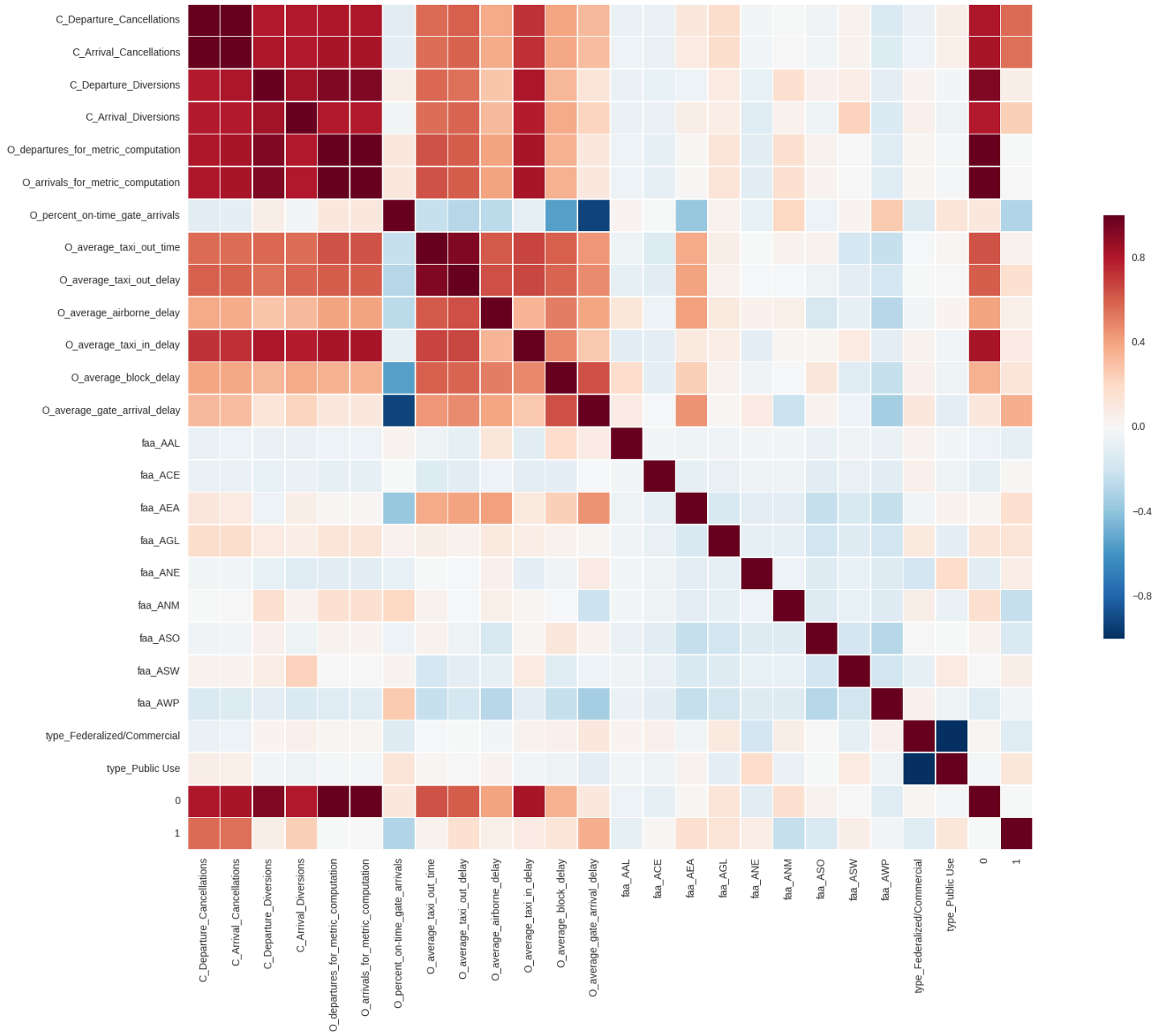
And test data results:

```
Test Data score:  1.0
Explained variance regression score: 1.0
Mean Absolute Error: 1.77043564994e-14
```



We can use PCA to reduce the features. In this partcular case since we dont have a lot of features PCA is not going to help us that much but if we had hundreds or thousands of feature PCA would reduce processing time significantly.

If we use 2 as number of components and compare it to the rest of features we can see they have a high correlation with important features:

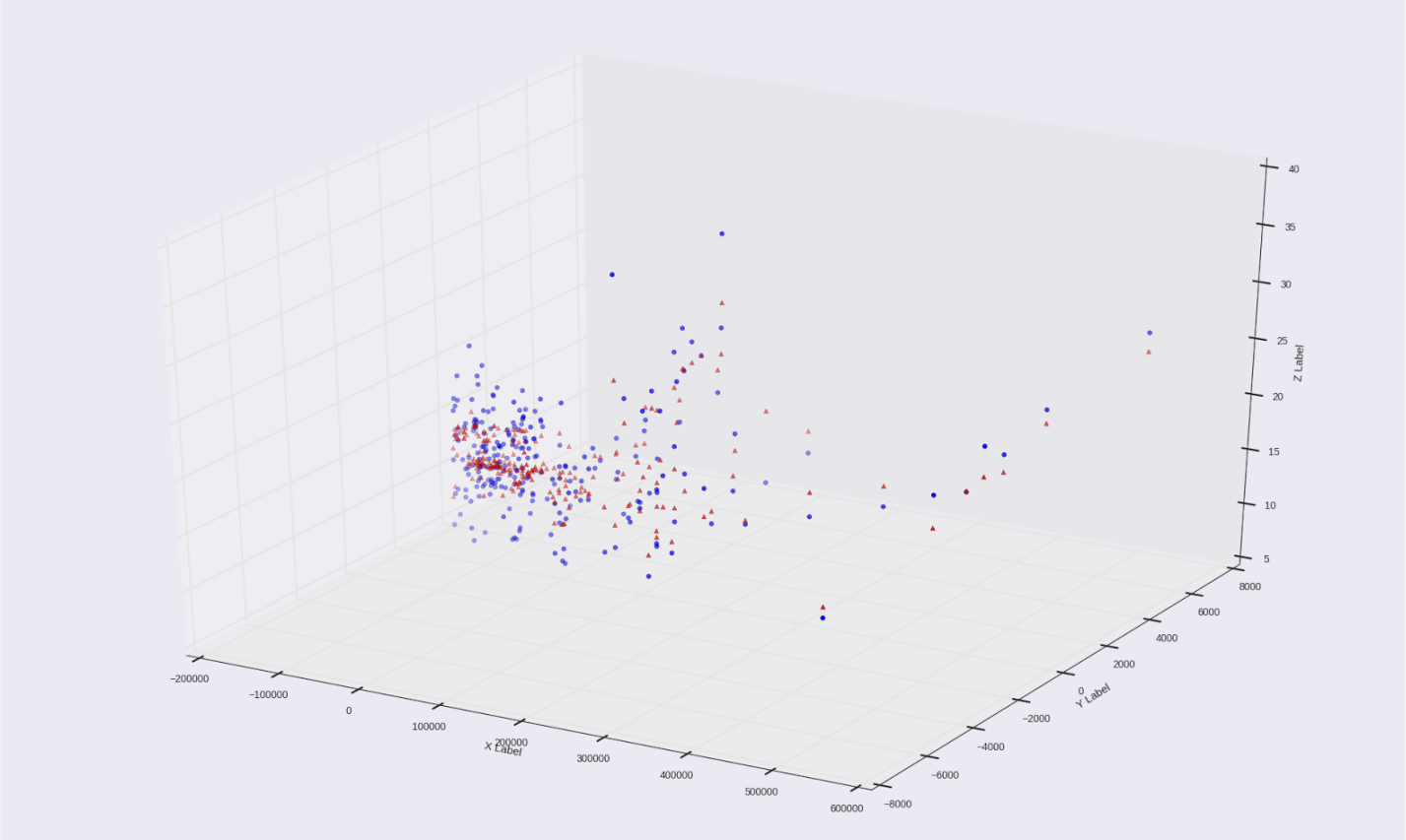


If we run Random forest with this PCA we can see we have terrible results which suggest 2 is not the best number of PCA components

```
Best Model after Grid Search:
RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=5,
           max_features='auto', max_leaf_nodes=None, min_samples_leaf=1,
           min_samples_split=2, min_weight_fraction_leaf=0.0,
           n_estimators=20, n_jobs=-1, oob_score=False, random_state=None,
           verbose=0, warm_start=False)
Mean score of the model is: 0.30217350972
('Explained variance regression score: ', 0.42413532637721252)
Mean Absolute Error: 2.28654098767
This operation took 198.532234907 seconds
```

And test data set scores:

```
Test Data score:  0.576482066441
Explained variance regression score: 0.57701358278581738
Mean Absolute Error: 2.23448689265
```



So we have to choose another number of components. We can use Eigenvectors to help us find the best number of components which results in:

```
array([  32.58072997,   45.25107368,   54.22811336,   60.10633639,
         65.54253378,   70.56703126,   75.36767034,   79.88855899,
         84.13182103,   88.31124643,   91.4089346 ,   94.0390597 ,
         95.72230263,   96.94703491,   98.02010808,   98.76231629,
         99.37037665,   99.64247374,   99.87029901,   99.99241877,
         99.99933537,  100.        ,  100.        ,  100.        ])
```

As we can see from the results above we can choose something between 10 and 12 to get the best results. After using random forest using 10 and 12 as number of components we can see results have improved significantly:

With PCA 10:

```
Best Model after Grid Search:
RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=None,
           max_features='auto', max_leaf_nodes=None, min_samples_leaf=1,
           min_samples_split=2, min_weight_fraction_leaf=0.0,
           n_estimators=200, n_jobs=-1, oob_score=False, random_state=None,
           verbose=0, warm_start=False)
Mean score of the model is: 0.801629721779
('Explained variance regression score: ', 0.79344672821629281)
```

```
Mean Absolute Error: 1.37042
This operation took 196.964634895 seconds
```

And test dataset:

```
Test Data score:  0.971699335985
Explained variance regression score: 0.97183308119432898
Mean Absolute Error: 0.571407708333
```

With PCA 12:

```
Best Model after Grid Search:
RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=None,
           max_features='auto', max_leaf_nodes=None, min_samples_leaf=1,
           min_samples_split=2, min_weight_fraction_leaf=0.0,
           n_estimators=200, n_jobs=-1, oob_score=False, random_state=None,
           verbose=0, warm_start=False)
Mean score of the model is: 0.722701821406
('Explained variance regression score: ', 0.83293312845885337)
Mean Absolute Error: 1.3938125
This operation took 197.029016972 seconds
```

And test dataset:

```
Test Data score:  0.970315075984
('Explained variance regression score: ' 0.97041780039229575
Mean Absolute Error: 0.569477083333
```

We can Also run a loop to run random forest on different number of components. We may get a different results than Eigenvector but it is in the same area.

As we saw in our findings most important features which FAA can focus on are Average gate arrival delays average taxi out delay, departure cancellations and FAA regions AEA and AWP have highest impact on delays which suggest they need the most attention.