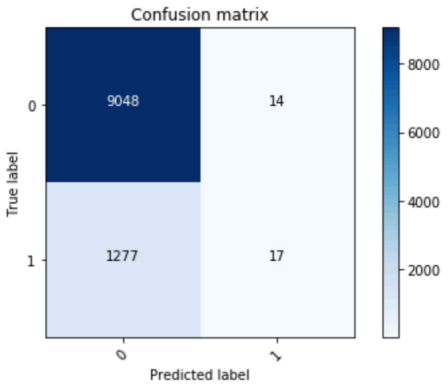
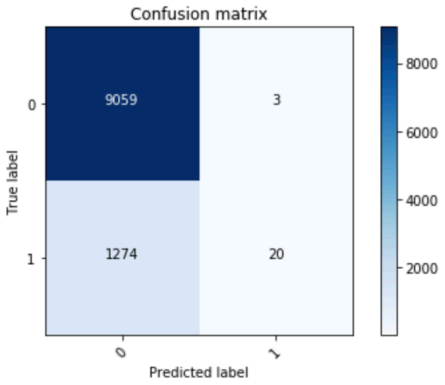
1. **Data Prep**
    a. Check missing data in each column
        i. Exclude columns with > 90% missing data
    b. Check data dictionary and look for duplicated variables in common sense
        i. Exclude WheelType, but keep WheelTypeID
        ii. Exclude VNZIP but keep VNST
        iii. Exclude PurchDate and RefID for prediction
    c. Check label class
        i. Based on the % of each class, we have over 80% label as 0, so considered as imbalance data
2. **Feature selection using Chi_squared Test**
        i. Many of the variables are categorical variables
        ii. Need to test if there is significant relationship between variables
        iii. Pick p_value threshold of 0.05
        iv. Reserve only important features for further analysis
3. **Train model**
    a. It is a classification problem
        i. Test logistic regression
        ii. Test random forest classification (w/o class_weight)
        iii. Test random forest classification (w class_weight)
        iv. Test random forest classification (w/ SMOTE)
    b. Split the train data into train (80%) and test (20%)
        i. Feature hashing: I picked this over one-hot-coding because it will present categories in similar style but with a lower dimension. Also, I have more successful implementation on this method over OHE based on my past experience
        ii. Evaluation metrics: since the data is imbalanced, cannot just use accuracy score, and I decided to use confusion matrix, as well as look at the precision, recall, f1-score.
4. **Results of models**
    As shown in three plots below, three models perform similar for Class 0 prediction, however RF_weighted model out-preforms in Class 1 prediction. The recall values for Class 1 prediction are relatively low. The metrics got better when apply SMOTE before applying RF model. This technique was used in final prediction. This final model makes better prediction for Class 1, with sacrifice on Class 0 prediction.
5. **Further Steps**
    a. Fine tune the model using cross-validation to optimize the parameters in RF model
    b. Try other classification models, e.g. SVM etc.
    c. Try other techniques to address imbalance data issue, e.g. penalized model, anomaly detection, collect more data etc.

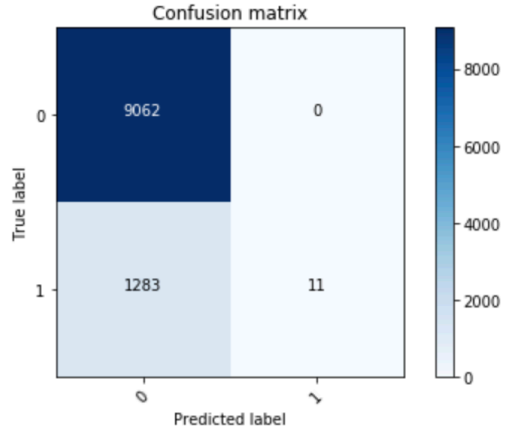
```
Final Report for LR
              precision    recall  f1-score  support     pred
0              0.876320  0.998455  0.933409   9062.0  10325.0
1              0.548387  0.013138  0.025660   1294.0     31.0
avg / total    0.835344  0.875338  0.819984  10356.0  10356.0
Confusion matrix, without normalization
[[9048   14]
 [1277   17]]
```




```
Final Report for RF
              precision    recall  f1-score  support     pred
0              0.876706  0.999669  0.934158   9062.0  10333.0
1              0.869565  0.015456  0.030372   1294.0     23.0
avg / total    0.875813  0.876690  0.821229  10356.0  10356.0
Confusion matrix, without normalization
[[9059    3]
 [1274   20]]
```




```
Final Report for RF_Weighted
              precision    recall  f1-score  support     pred
0              0.875979  1.000000  0.933890   9062.0  10345.0
1              1.000000  0.008501  0.016858   1294.0     11.0
avg / total    0.891475  0.876110  0.819305  10356.0  10356.0
Confusion matrix, without normalization
[[9062    0]
 [1283   11]]
```



```
Final Report for RF w/SMOTE
             precision    recall  f1-score  support     pred
0             0.899694  0.972964  0.934896   9062.0   9800.0
1             0.559353  0.240340  0.336216   1294.0    556.0
avg / total   0.857168  0.881421  0.860090  10356.0  10356.0
Confusion matrix, without normalization
[[8817  245]
 [ 983  311]]
```