Chronic Kidney Disease Prediction in a highly Imbalanced Dataset

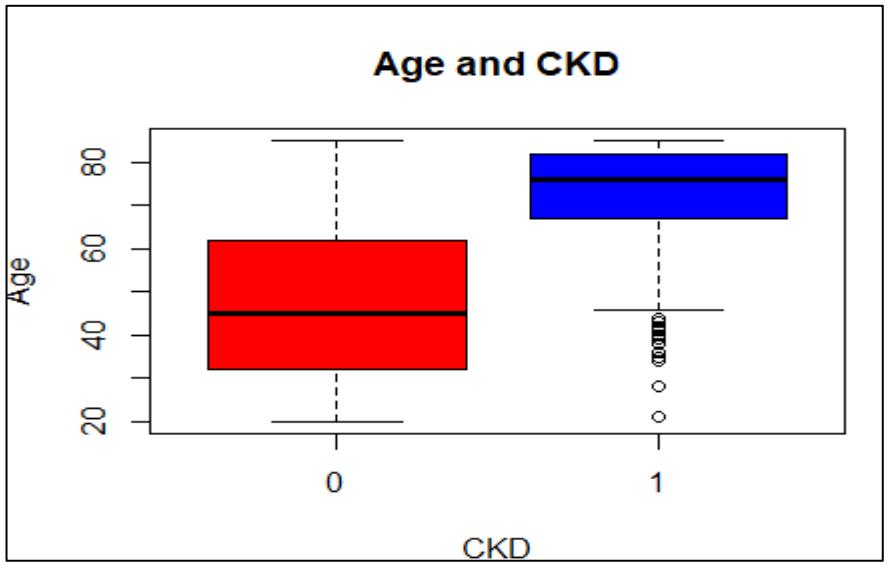
By,

Manthiramoorthy Cheranthian

Descriptive Statistics and Variable Selection:

We perform EDA before data cleansing to check significances before loosing data.

- *Relationship between Age and CKD*:



Hypothesis testing between two groups (Numeric - Target):

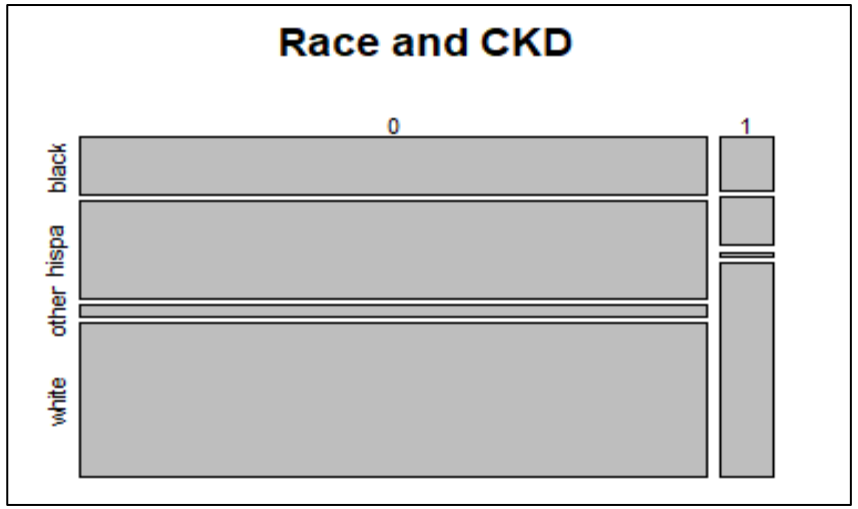
```
> t.test(Age~CKD,data=data)

        Welch Two Sample t-test

data:  Age by CKD
t = -43.564, df = 660.05, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -27.06473 -24.73019
sample estimates:
mean in group 0 mean in group 1
       47.15426        73.05172
```

- Age seems to have higher significance, persons with age>70
  Has higher chance getting affected by CKD.
- Strong P-value (<2.2e-16) also suggests higher significance with the target.
- Hence selecting Age for our model.

Relationship between Race group and CKD:

Chi-Squared testing between two Factors:

```
> ##Checking for significance of race
> chisq.test(data$CKD,data$Racegrp)

        Pearson's Chi-squared test

data:  data$CKD and data$Racegrp
X-squared = 71.704, df = 3, p-value = 1.843e-15
```

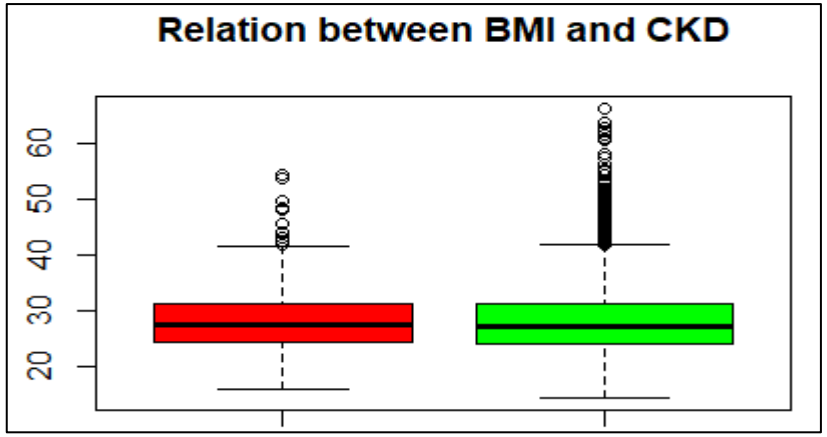


- It is difficult to identify clear significance from Mosaic plot we can test it further by using chi-squared test (Factor-Factor).
- Strong p-value from chi squared test indicates higher significance.
- Hence selecting Race group for our model.

Relationship between BMI and CKD:

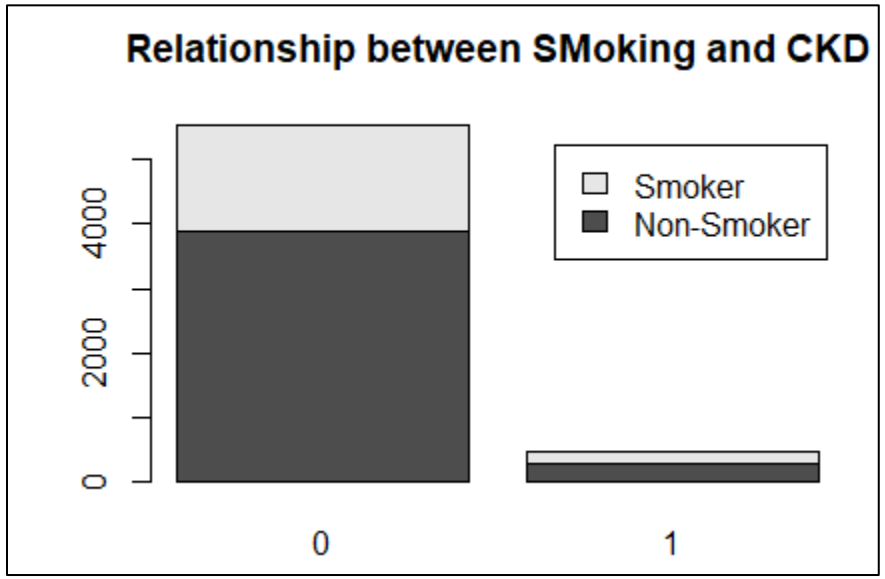
Hypothesis testing between tw2 groups (Numeric - Target)

```
> t.test(ckd,nockd,conf.level = 0.99)

        Welch Two Sample t-test

data:  ckd and nockd
t = 0.35717, df = 488.42, p-value = 0.7211
alternative hypothesis: true difference in means is not equal to 0
99 percent confidence interval:
 -0.6800425  0.8980044
sample estimates:
mean of x mean of y
 28.34544  28.23646
```



- It seems like there is very weak significance between BMI and CKD we can confirming by preforming t test
- Weak and high p-value from t test indicates no significance.
- Hence neglecting BMI from our model.

Descriptive Statistics and Variable Selection:



Hypothesis testing between two Factor - Factor:

```
> chisq.test(data$CKD,data$Smoker)

        Pearson's Chi-squared test with Yates' continuity correction

data:  data$CKD and data$Smoker
X-squared = 26.896, df = 1, p-value = 2.147e-07
```

- Smoking habit seems to have significance over CKD.
- Performing a chi squared statistical test to confirm the relationship.
- Strong P-value (<2.147e-07) also suggests higher significance with the target.
- Hence selecting Smoking variable for our model.

Significant Variables:

- Following are the variables selected for our model using EDA and statistical tests.
- Age **To see EDA, tests for all variables please see .R file*
- Racegrp
- SBP
- DBP
- HDL
- LDL
- PVD
- Activity
- Smoker
- Hypertension
- Diabetes
- Stroke
- CVD
- Fam.CVD
- CHF
- Anemia

Target variable distribution:

```
> table(data$CKD)

   0    1
5536  464
```

- We could see that our dataset has very less values for positive cases which could weaken our model and leads to reduced Sensitivity which is very detrimental since we are trying to produce a model that could explain Positive cases.

- Solution for this would be to train our model in both Over sampled and under sampled dataset and select the model with increased F1-score and Sensitivity.

Data Preparation:

```
naval=which(!complete.cases(data))##Gives rows which has NA values
str(naval)
int [1:3904] 2 10 11 24 29 33 41 53 55 61 ...
```

- a.)

```
> prdata=data[-naval,]
> str(prdata)
'data.frame':   4915 obs. of  29 variables:
```

b.)

- We see that totally there are 3904 rows which has at least one NA value in either of their columns.
- We can try to impute NA values using KNN, mean/median imputation but since some of the variables are vital health information, we do not want to impute the values which might affect the meaning of the data.
- Hence, we remove the records with NA values to create a model with cleaned data.
- b.) Our processed data with no NA values has a total of 4915 observations.

Train – Test Split before Sampling:

- Splitting 75% of the data for train and 25% of the data for test.

```
> set.seed(111)
> indx=sample(2,nrow(prdata),replace=TRUE ,prob=c(0.75,0.25) )
> traindata=prdata[indx==1,]
> testdata=prdata[indx==2,]
```

- Above are the distribution of the target variable in both test and train data.

```
> table(traindata$CKD)

   0    1
3486  244
> table(testdata$CKD)

   0    1
1107   78
```

```
> ##Model in Imbalanced dataset
> modl=glm(CKD~Age+Racegrp+Unmarried+CareSource+Waist+SBP+DBP+HDL+LDL+PV
abetes+Stroke+CVD+Fam.CVD+CHF+Anemia,data=prdata,family = "binomial")
> summary(modl)

Call:
glm(formula = CKD ~ Age + Racegrp + Unmarried + CareSource +
    Waist + SBP + DBP + HDL + LDL + PVD + Activity + Smoker +
    Hypertension + Diabetes + Stroke + CVD + Fam.CVD + CHF +
    Anemia, family = "binomial", data = prdata)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8770  -0.3015  -0.1358  -0.0698   3.4038

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)       -14.534817  324.746019  -0.045 0.964301
```

```
> pred=predict(modl,testdata)
> predic=ifelse(pred>=0.5, 1,0)
> predic=as.factor(predic)
> ##confusionMatrix
> library(caret)
> library(e1071)
> confusionMatrix(predic,testdata$CKD,positive='1')
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1100   69
         1    7    9

               Accuracy : 0.9359
                 95% CI : (0.9204, 0.9491)
    No Information Rate : 0.9342
    P-Value [Acc > NIR] : 0.4368

                  Kappa : 0.173

 Mcnemar's Test P-Value : 2.612e-12

            Sensitivity : 0.115385
            Specificity : 0.993677
         Pos Pred Value : 0.562500
         Neg Pred Value : 0.940975
```

- Logistic Regression without sampling.
- We can see from the result that Accuracy is high, but Sensitivity is very less compared to specificity.
- To solve this problem we use sampling techniques.

Developing a simple Logistic Regression:

Oversampling:

- We use ROSE library (Randomly Over Sampling Examples) for over sampling.
- Since the target variable distribution has 3486 negative samples and 244 positive samples.
- Such that we create a sample with 3486*2=6972 samples such that 244 positive samples will be scaled to another 3486 samples.
- *Imbalanced*

```
> table(traindata$CKD)

   0    1
3486  244
```

Over Sampled

```
> over=ovun.sample(CKD~Age+Racegrp+Unmarried+CareSource+Waist+SBP+DBP+HDL+LDL+PVD+Activity+Smoker+Hyperte
nsion+Diabetes+Stroke+CVD+Fam.CVD+CHF+Anemia,data=traindata,method='over',N=6972)$data
> table(over$CKD)

   0    1
3486 3486
```

Under sampling:

- Our target variable distribution has 3486 negative samples and 244 positive samples.
- Now we create a under sample with 244*2=488 samples such that 3486 negative samples will be scaled down to another 244 samples.

Imbalanced

```
> table(traindata$CKD)

   0    1
3486  244
```

Under Sampled

```
> under=ovun.sample(CKD~Age+Racegrp+Unmarried+CareSource+Waist+SBP+DBP+HDL+LDL+PVD+Activity+Smoker+Hypert
ension+Diabetes+Stroke+CVD+Fam.CVD+CHF+Anemia,data=traindata,method='under',N=488)$data
> table(under$CKD)

  0   1
244 244
```

Building Models in Balanced data:

```
> ##Logistic Regression built using Over Sampled data
> overmodl=glm(CKD~Age+Racegrp+Unmarried+CareSource+Waist+SBP+DBP+HDL+LDL+PVD+Activity+Smoker+Hyp
n+Diabetes+Stroke+CVD+Fam.CVD+CHF+Anemia,data=over,family = "binomial")
> summary(overmodl)

Call:
glm(formula = CKD ~ Age + Racegrp + Unmarried + CareSource +
    Waist + SBP + DBP + HDL + LDL + PVD + Activity + Smoker +
    Hypertension + Diabetes + Stroke + CVD + Fam.CVD + CHF +
    Anemia, family = "binomial", data = over)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1635  -0.5338   0.0667   0.6260   2.5384

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)       -1.217e+01  1.970e+02  -0.062 0.950725
Age                8.289e-02  3.066e-03  27.037  < 2e-16 ***
Racegrphispa      -7.024e-01  1.192e-01  -5.891 3.83e-09 ***
Racegrpother       3.352e-01  2.202e-01   1.523 0.127871
Racegrpwhite       4.122e-01  1.015e-01   4.063 4.85e-05 ***
Unmarried1         1.634e-01  7.322e-02   2.232 0.025622 *
CareSourceclinic   8.085e+00  1.970e+02   0.041 0.967258
CareSourceDrHMO    8.353e+00  1.970e+02   0.042 0.966172
CareSourcenoplace  8.231e+00  1.970e+02   0.042 0.966669
CareSourceother    8.480e+00  1.970e+02   0.043 0.965661
Waist             -9.274e-03  2.732e-03  -3.395 0.000686 ***
SBP               -1.852e-03  1.948e-03  -0.951 0.341644
DBP               -3.978e-03  2.939e-03  -1.354 0.175893
HDL               -1.490e-02  2.367e-03  -6.293 3.11e-10 ***
LDL                1.956e-03  8.214e-04   2.382 0.017223 *
PVD1               3.297e-01  1.382e-01   2.386 0.017048 *
Activity2         -1.674e-01  8.031e-02  -2.085 0.037087 *
Activity3         -7.463e-01  1.195e-01  -6.247 4.17e-10 ***
Activity4         -8.314e-01  2.324e-01  -3.578 0.000346 ***
```

```
> pr=predict(overmodl,testdata)
> pre=ifelse(pr>=0.5, 1,0)
> pre=as.factor(pre)
> confusionMatrix(pre,testdata$CKD,positive='1')
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 955  22
         1 152  56

               Accuracy : 0.8532
                 95% CI : (0.8317, 0.8728)
    No Information Rate : 0.9342
    P-Value [Acc > NIR] : 1

                  Kappa : 0.3272

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.71795
            Specificity : 0.86269
         Pos Pred Value : 0.26923
         Neg Pred Value : 0.97748
             Prevalence : 0.06582
         Detection Rate : 0.04726
   Detection Prevalence : 0.17553
      Balanced Accuracy : 0.79032

       'Positive' Class : 1
```

- We can see a clear significant increase in Sensitivity from 0.11 to 0.85 with a little decrease in Specificity and Accuracy

Model in Under sampled data:

```
> undermod1=glm(CKD~Age+Racegrp+Unmarried+CareSource+Waist+SBP+DBP+HDL+LDL+PVD+Act
on+Diabetes+Stroke+CVD+Fam.CVD+CHF+Anemia,data=under,family = "binomial")
> summary(undermod1)

Call:
glm(formula = CKD ~ Age + Racegrp + Unmarried + CareSource +
    Waist + SBP + DBP + HDL + LDL + PVD + Activity + Smoker +
    Hypertension + Diabetes + Stroke + CVD + Fam.CVD + CHF +
    Anemia, family = "binomial", data = under)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.47896  -0.45330  -0.01418   0.59003   2.48895

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)     -4.288614   1.790344  -2.395  0.01660 *
Age              0.087071   0.012242   7.112 1.14e-12 ***
Racegrphispa    -0.595722   0.481892  -1.236  0.21638
Racegrpother     0.364264   0.832309   0.438  0.66164
Racegrpwhite     0.318885   0.405525   0.786  0.43166
Unmarried1       0.180078   0.289950   0.621  0.53456
CareSourceDrHMO  0.514420   0.329622   1.561  0.11861
CareSourcenoplace 0.228665  0.592304   0.386  0.69945
CareSourceother  0.762513   0.733773   1.039  0.29873
Waist           -0.018459   0.010685  -1.728  0.08407 .
SBP              0.007087   0.008026   0.883  0.37724
DBP             -0.005574   0.011718  -0.476  0.63429
HDL             -0.010867   0.009105  -1.194  0.23265
LDL              0.001660   0.003275   0.507  0.61228
PVD1            -0.348611   0.494817  -0.705  0.48111
Activity2       -0.757607   0.331425  -2.286  0.02226 *
Activity3       -1.239399   0.492049  -2.519  0.01177 *
Activity4       -0.571479   0.914851  -0.625  0.53219
Smoker1         -0.387524   0.293697  -1.319  0.18701
Hypertension1    0.610412   0.333791   1.829  0.06744 .
Diabetes1        1.059263   0.396088   2.674  0.00749 **
Stroke1          0.722049   0.890528   0.811  0.41747
CVD1             0.611157   0.623742   0.980  0.32717
```

```
> pr=predict(undermod1,testdata)
> pre=ifelse(pr>=0.5, 1,0)
> pre=as.factor(pre)
> confusionMatrix(pre,testdata$CKD,positive='1')
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 936  22
         1 171  56

               Accuracy : 0.8371
                 95% CI : (0.8149, 0.8577)
    No Information Rate : 0.9342
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2985

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.71795
            Specificity : 0.84553
         Pos Pred Value : 0.24670
         Neg Pred Value : 0.97704
             Prevalence : 0.06582
         Detection Rate : 0.04726
   Detection Prevalence : 0.19156
      Balanced Accuracy : 0.78174

       'Positive' Class : 1
```

- Even though the model in Under sample gives a high sensitivity and a reasonable accuracy performance metrics for model in over sample is higher than the model in under sample.

Comparing performance for different models in different balanced data:

	Over Sampling	Under Sampling
Logistic Regression	Accuracy= 0.85, Sensitivity=0.71, Specificity=0.86, F1- Score=0.77	Accuracy= 0.83, Sensitivity=0.71, Specificity=0.84, F1- Score=0.76
Random Forests	Accuracy= 0.92, Sensitivity=0.17, Specificity=0.98, F1- Score=0.28	Accuracy= 0.77, Sensitivity=0.85, Specificity=0.77, F1- Score=0.80
SVM	Accuracy= 0.82, Sensitivity=0.76, Specificity=0.83, F1- Score=0.78	Accuracy= 0.75, Sensitivity=0.91, Specificity=0.74, F1- Score=0.81

Conclusion:

- We could see that we have increased our Sensitivity to a greater extent using various sampling techniques.
- Even though we give more importance to a model with higher Sensitivity which is our requirement, using F-1 score would be a perfect parameter to select our final model.
- F1 score rates the performance of a model by generalizing both Sensitivity and Specificity.
- Hence, we select a model with high F1 score and Sensitivity.
- Hence, we finally choose **SVM** and **Random forest** trained with under sampled data as our final model which has higher F1 score of 0.80 and 0.81.