Case Study

Predict Customer Churn with R

Maxine(Diandian) Yi

September 11, 2019

Summary

- Load libraries and the data
  - Libraries employed: tydiverse,ggplot2,corrplot,caret,corrplot,rattle,ROSE,DMwR,kableExtra
  - Two data set to start with: train/test
- Exploratory data analysis and data preprocess
  - Exploratory data analysis only on train dataset
  - Data structure/Missing value/Variables distribution and correlation
- Train models with different methods and control setting
- Evaluate the performance

A glance at the data

```
dim(train);dim(test)
```

```
## [1] 142403     31
```

```
## [1] 142404     30
```

A glance at the data(2)

```
## 'data.frame':    142403 obs. of  31 variables:
##  $ V0   : num  75604 65147 155927 46798 47785 ...
##  $ V1   : num  1.38 -2.64 2.32 1.24 1.12 ...
##  $ V2   : num  -0.9931 -1.6813 -1.4291 0.2466 -0.0769 ...
##  $ V3   : num  -0.0283 1.6797 -1.2076 0.1731 1.307 ...
##  $ V4   : num  -1.289 -0.33 -1.622 0.506 1.224 ...
##  $ V5   : num  -1.235 -0.191 -1.089 -0.21 -0.929 ...
##  $ V6   : num  -1.242 -1.229 -0.538 -0.573 0.122 ...
##  $ V7   : num  -0.3485 -0.7681 -1.1268 -0.0677 -0.6779 ...
##  $ V8   : num  -0.42766 0.78906 -0.12568 0.00195 0.24698 ...
##  $ V9   : num  -2.274 0.911 -1.027 -0.105 0.652 ...
##  $ V10  : num  1.3915 -1.2272 1.6117 -0.0941 -0.027 ...
##  $ V11  : num  -0.267 -1.528 -1.518 1.222 0.884 ...
##  $ V12  : num  -0.461 -0.463 -1.625 0.464 1.1 ...
##  $ V13  : num  1.087 -1.907 -0.851 -0.458 -0.182 ...
##  $ V14  : num  -0.1768 0.0776 -0.1567 0.0199 -0.1382 ...
##  $ V15  : num  0.2593 -0.7458 -0.0115 0.4901 -0.5081 ...
##  $ V16  : num  -0.643 0.315 -0.453 0.803 0.281 ...
##  $ V17  : num  0.604 0.156 0.466 -0.34 -0.492 ...
##  $ V18  : num  -0.631 -0.193 0.143 0.386 0.449 ...
##  $ V19  : num  0.00402 -0.26885 0.00235 0.22133 0.08131 ...
##  $ V20  : num  -0.126 0.621 -0.482 -0.101 -0.14 ...
##  $ V21  : num  -0.4703 0.0567 -0.207 -0.2653 -0.015 ...
##  $ V22  : num  -1.178 -0.648 -0.178 -0.833 0.13 ...
##  $ V23  : num  0.0923 0.2209 0.2389 0.0834 -0.0115 ...
##  $ V24  : num  0.3487 0.3865 0.4193 -0.0605 0.336 ...
##  $ V25  : num  0.293 -0.1 -0.155 0.208 0.382 ...
##  $ V26  : num  -0.4908 0.7847 -0.1741 0.0998 -0.4256 ...
##  $ V27  : num  -0.00602 0.11646 -0.00602 -0.03166 0.06862 ...
##  $ V28  : num  0.0331 -0.1963 -0.0494 0.0159 0.0271 ...
##  $ V29  : num  98.25 134.56 20 1.98 4.99 ...
##  $ Class: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 2 1 1 ...
```

Exploratory Data Analysis

Missing value

```
sapply(train,function(x)sum(is.na(x)))
```

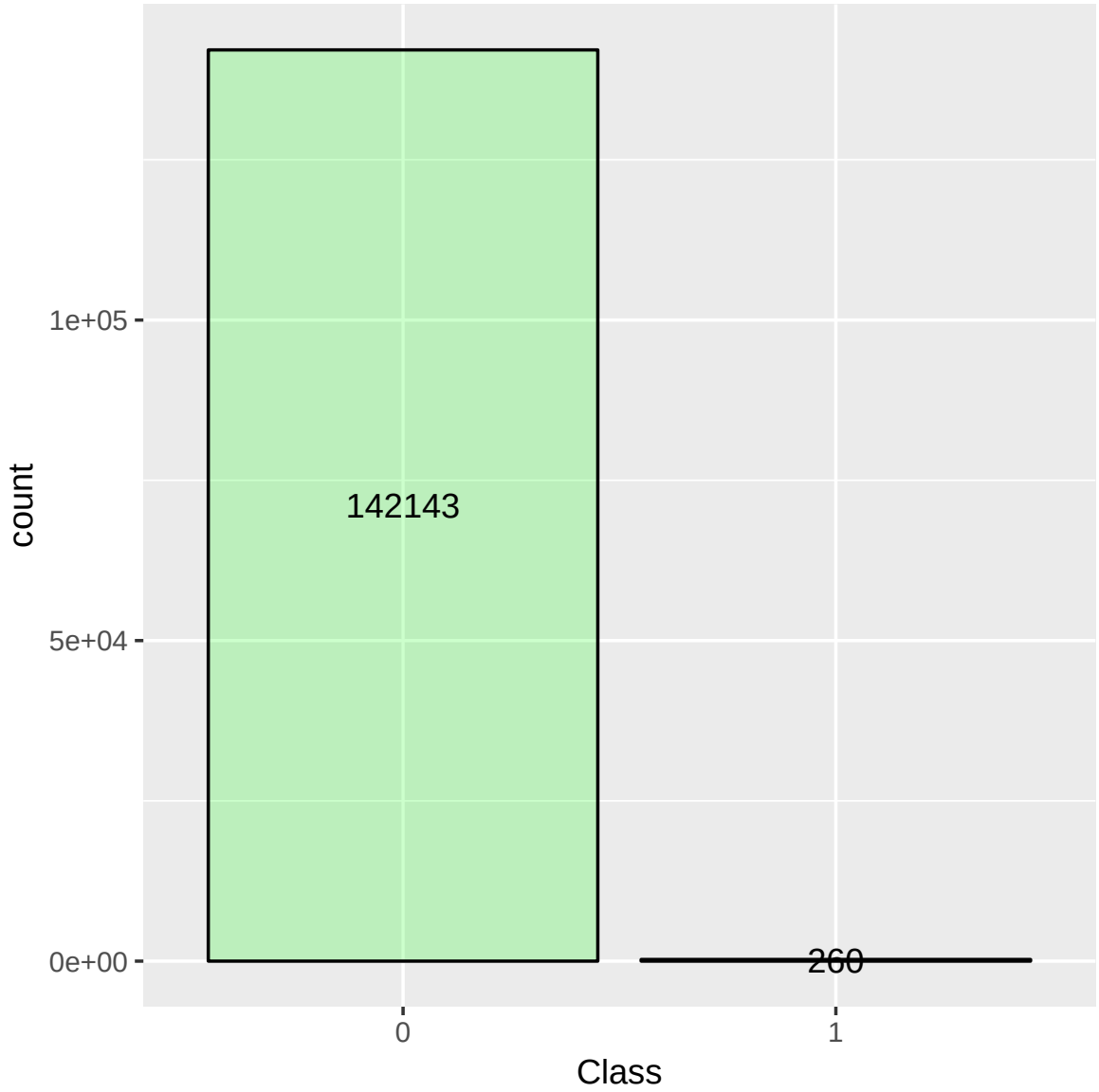
```
##    V0    V1    V2    V3    V4    V5    V6    V7    V8
##     0     0     0     0     0     0     0     0     0
##   V12   V13   V14   V15   V16   V17   V18   V19   V20
##     0     0     0     0     0     0     0     0     0
##   V24   V25   V26   V27   V28   V29 Class
##     0     0     0     0     0     0     0
```

Exploratory Data Analysis(Variables' variation)

```
nearZeroVar(train,saveMetrics = TRUE)
##        freqRatio percentUnique zeroVar   nzv
## V0      1.142857  62.798536548   FALSE FALSE
## V1      1.051282  96.436170586   FALSE FALSE
## V2      1.051282  97.414380315   FALSE FALSE
## V3      1.051282  97.263400350   FALSE FALSE
## V4      1.051282  97.370139674   FALSE FALSE
## V5      1.051282  97.483199090   FALSE FALSE
## V6      1.051282  97.422104871   FALSE FALSE
## V7      1.051282  97.485305787   FALSE FALSE
## V8      1.051282  97.541484379   FALSE FALSE
## V9      1.051282  97.478283463   FALSE FALSE
## V10     1.051282  97.484603555   FALSE FALSE
## V11     1.051282  97.297107505   FALSE FALSE
## V12     1.051282  97.508479456   FALSE FALSE
## V13     1.051282  97.427020498   FALSE FALSE
## V14     1.051282  97.527439731   FALSE FALSE
## V15     1.051282  97.425616033   FALSE FALSE
## V16     1.051282  97.499350435   FALSE FALSE
## V17     1.051282  97.526035266   FALSE FALSE
## V18     1.051282  97.528844196   FALSE FALSE
## V19     1.051282  97.504266062   FALSE FALSE
## V20     1.051282  97.452300864   FALSE FALSE
## V21     1.051282  97.422104871   FALSE FALSE
## V22     1.051282  97.514097315   FALSE FALSE
## V23     1.051282  97.445980773   FALSE FALSE
## V24     1.051282  97.540079914   FALSE FALSE
## V25     1.051282  97.496541505   FALSE FALSE
## V26     1.051282  97.457216491   FALSE FALSE
## V27     1.051282  97.597662971   FALSE FALSE
## V28     1.051282  97.553422330   FALSE FALSE
## V29     2.319826  16.120446901   FALSE FALSE
## Class 546.703846   0.001404465   FALSE  TRUE
```

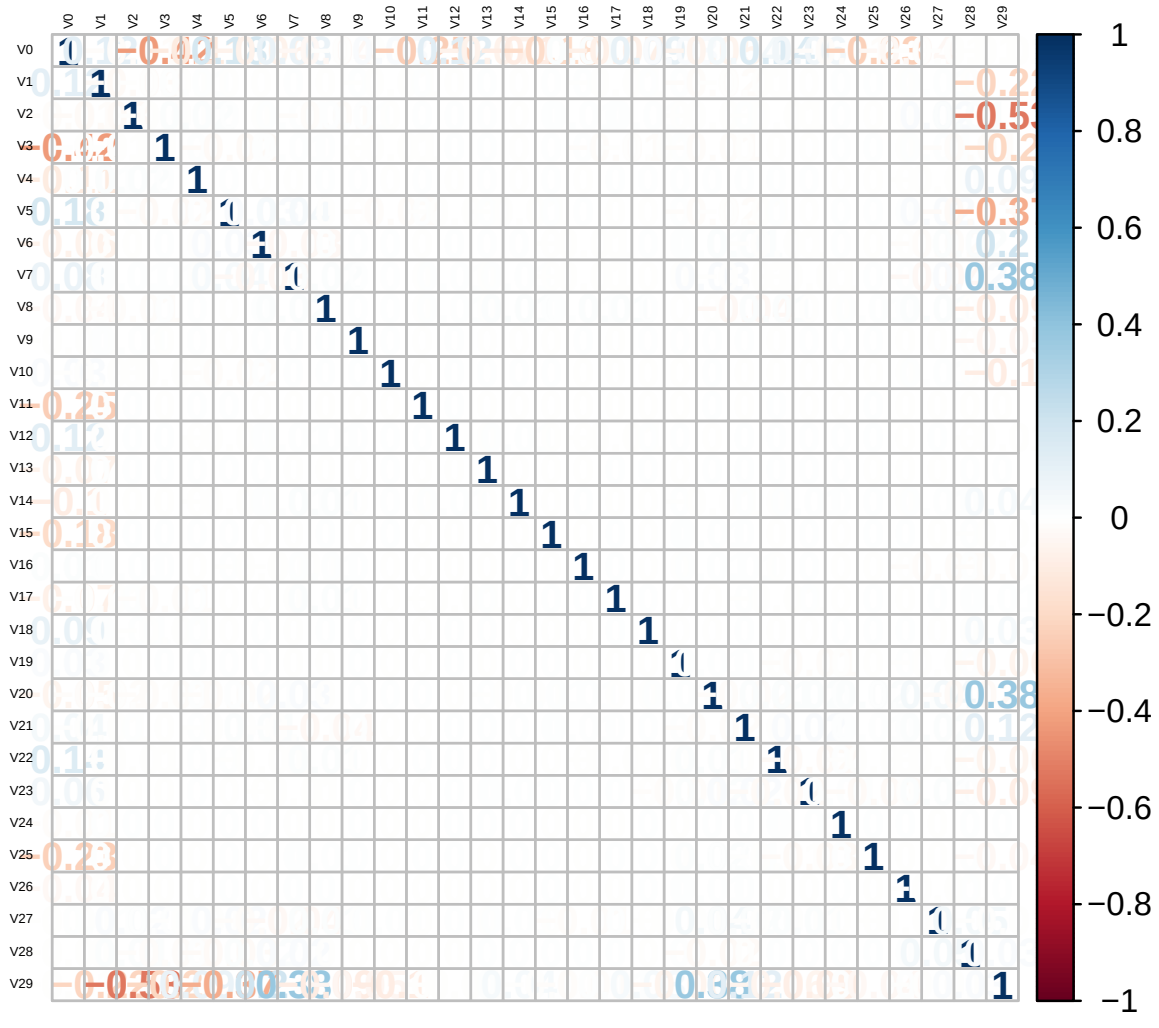
Exploratory Data Analysis

Categorical explained variable



Exploratory Data Analysis

Numeric variables correlation plot



Train models

Prediction Study Design

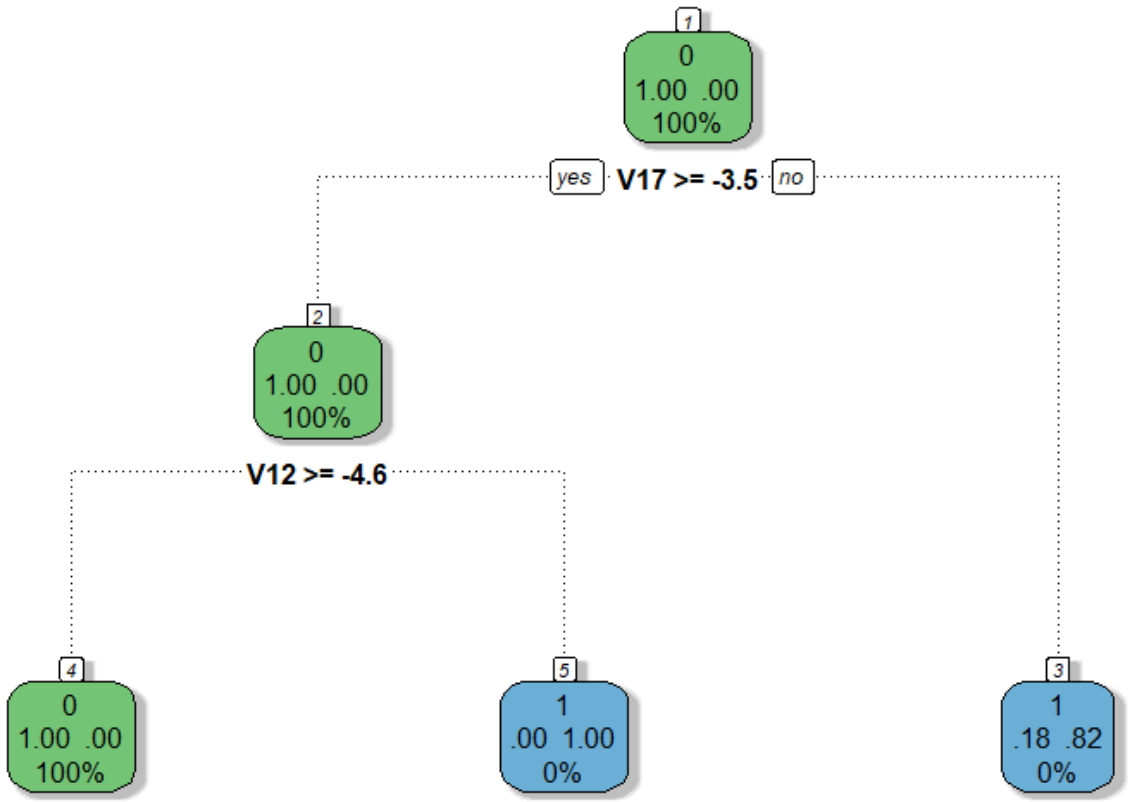
- Define my performance indicators
- Split the data: Training, Validation, Testing

```
set.seed(1234)
index <- createDataPartition(train$Class, p = 0.6, list = FALSE)
train_insample <- train[index, ]
test_insample  <- train[-index, ]
```

- Algorithm: Logistic regression, Decision tree
- Train Controling setting:
  - Use cross validation
  - Dealing with imbalance categorical explained variable

Train models

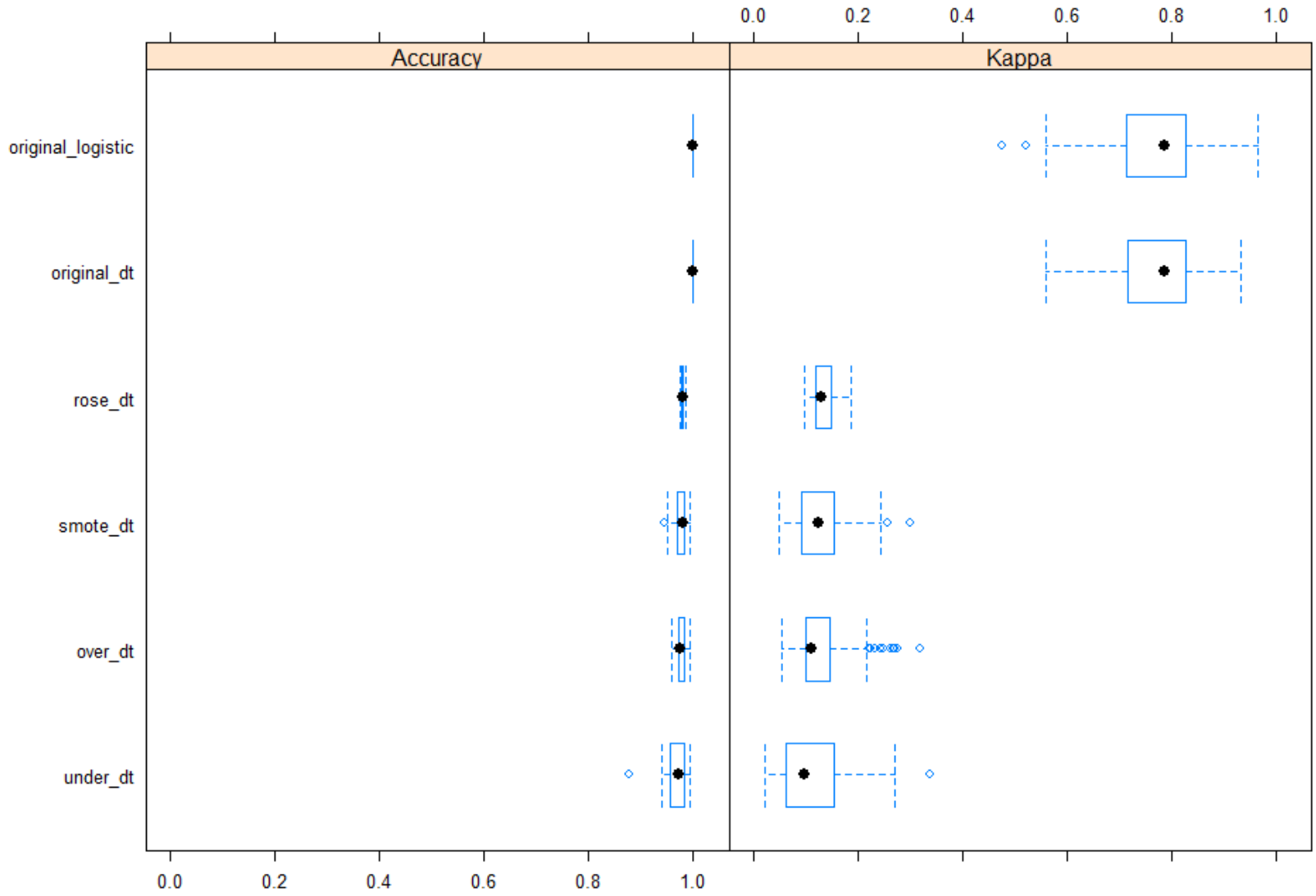
Decision tree



Rattle 2019-Sep-12 11:18:54 Diandian

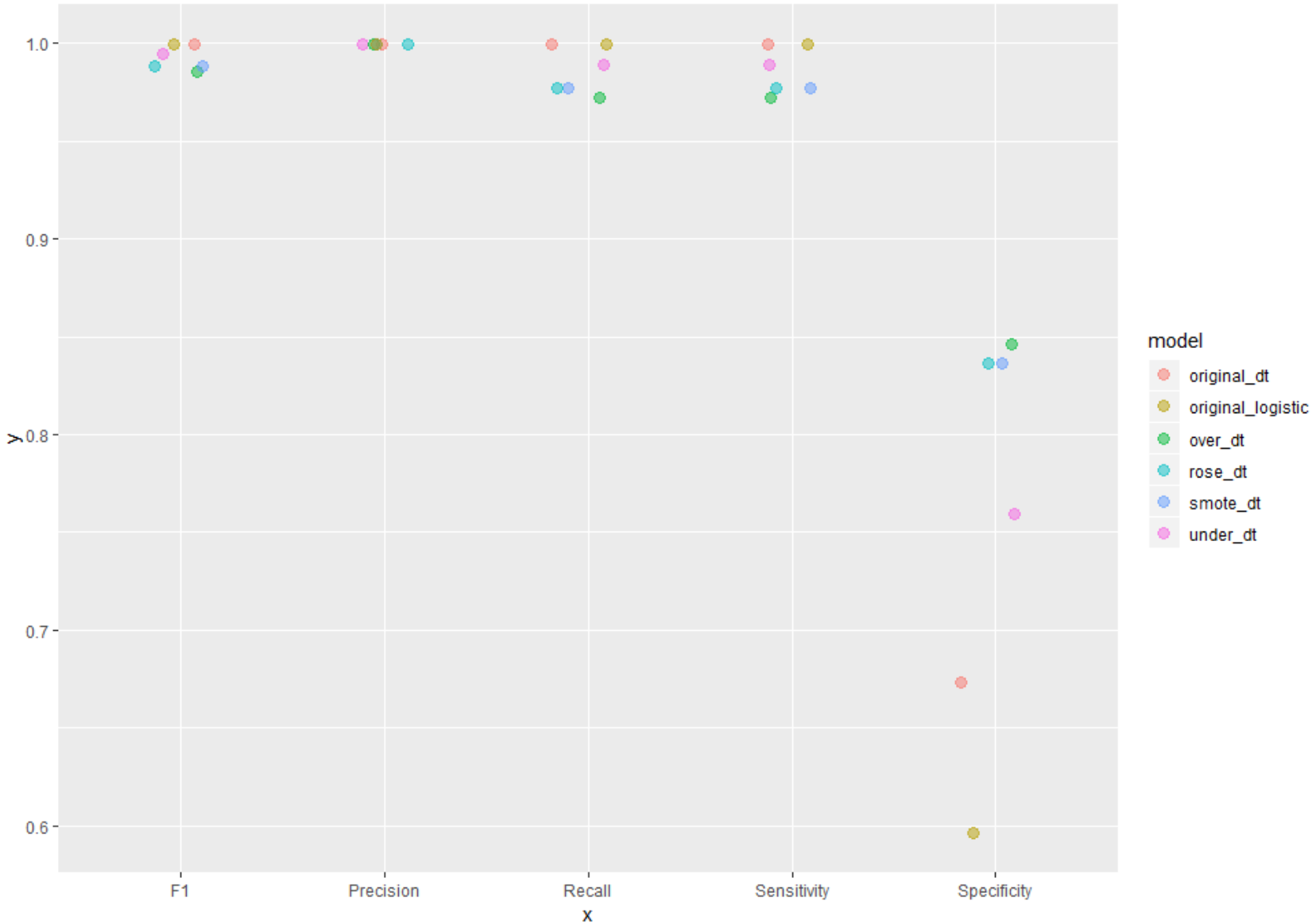
Model performance

Accuracy and Kappa



Model performance

Other performance



Model prediction

Use decision tree algorithm with SMOTE

Class	Count
0	139142
1	3262