# Supervised Machine Learning- Logistic Regression

# Agenda

- Logistic Regression Introduction
- Binomial Logistic Regression
- Model Diagnostic
- Regression Assumptions
- Feature Selection using Woe
- Implementation in R

## What is Logistic Regression ?

Used to estimate the probability that an event will occur as a function of other variables

To predict an outcome variable that is categorical or discrete from one or more continuous or categorical predictor variables

Instead of building a predictive model for Y (Response) directly, the approach models Log Odds(Y) ; hence the name Logistic or Logit.

In Logistic regression, the response variable (Y) is a dichotomous categorical variable

Can be considered as a class label with highest probability

The probability that a borrower will default as a function of his credit score, income, the size of the loan, and his existing debts

## Logistic Regression

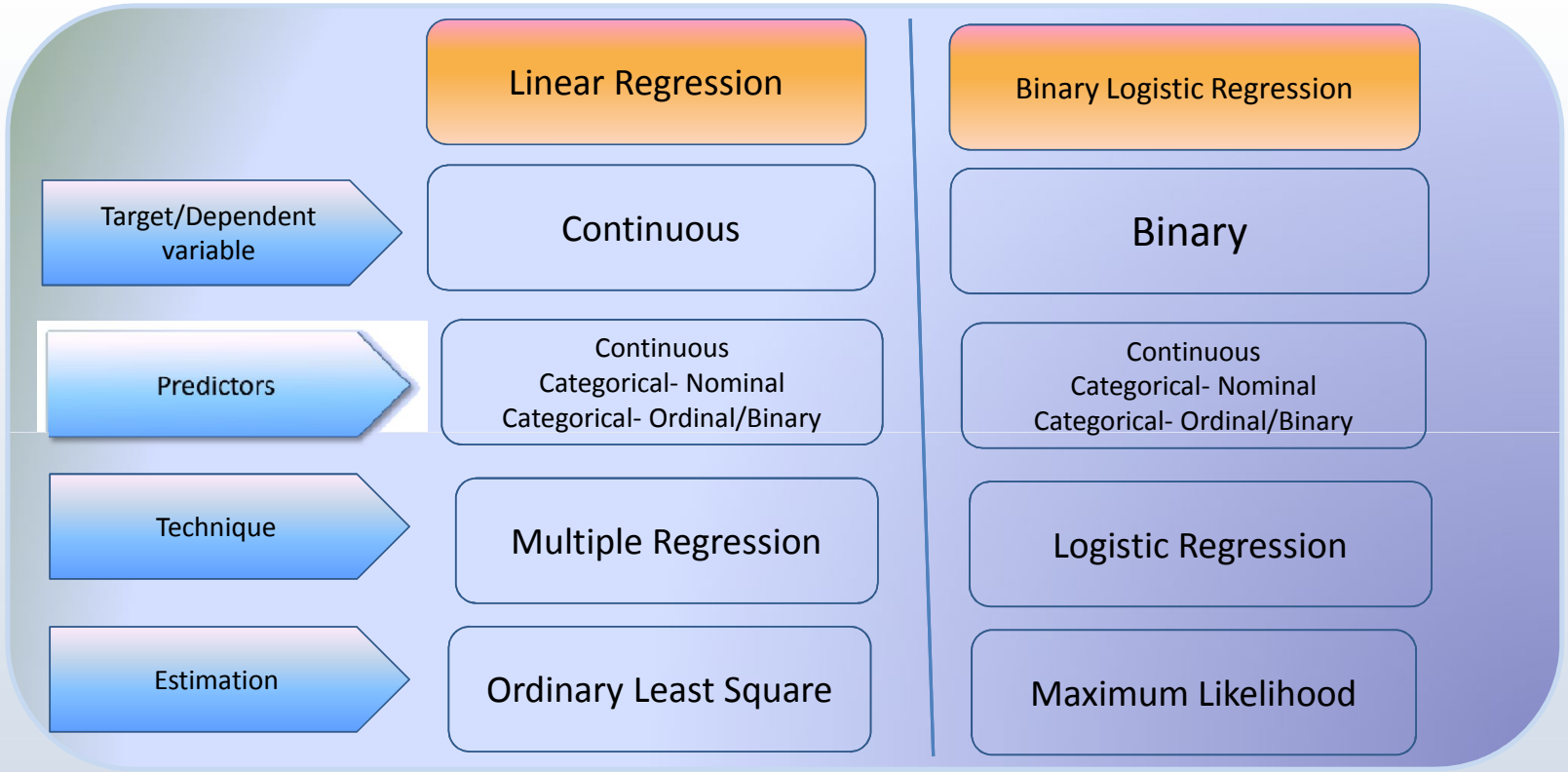
Input = > variables can be continuous or discrete

Output => Log-odds ratio easily converted to the probability of the outcome

## Types of Logistic Regression and its Use Cases

- The preferred method for many binary classification problems:
  - Especially if you are interested in the probability of an event, not just predicting the "yes or no“
  - Try this first; if it fails, then try something more complicated
- Binary Classification examples:
  - The probability that a borrower will default
  - The probability that a customer will churn
- Multi-class example
  - The probability that a politician will vote yes/vote no/not show up to vote on a given bill

## Linear Regression Vs Logistic Regression



## Logistic Regression (Binomial)

The Logistic regression has the following mathematical form :

$$\ln \frac{P(y=1)}{1-P(y=1)} = b_0 + b_1 x_1 + b_2 x_2 \ldots$$

y = 1 is the case of interest : 'TRUE'

L.H.S is called logit (p(y=1)) , hence it is called as logistic regression

P is the probability of event

$\beta_0$ is the intercept and $\beta_1$ is the slope

log transformation has a linear relationship with predictors ( a unit change in x will lead to a fixed % change in log Y)

## Probability, Odds , Odds Ratio & Logit

**What is probability ?**

**Probability =** $\frac{\text{Number of favorable outcome}}{\text{Total number of possible outcomes}}$

**What is an odds ?**

It is a term that denotes the probability of success to probability of failure

If probability of success is .5 , then odds ratio = .5/.5 = 1

If the probability of success is .5, then the odds ratio is = .50/.50 = 1

There is an equal chance of success

Odds of response by males = 51/100 = 0.51 Odds of response by females = 43/150 = .29

| Response | No | Yes |
|---|---|---|
| Gender | | |
| Male | 49 | 51 |
| Female | 107 | 43 |

**What is an odds ratio ?**

It is a ratio of two odds odds ratio = $\frac{\text{odds1}}{\text{odds0}}$ $\frac{\text{Odds of response by males}}{\text{Odds of response by Females}}$ **= 1.76 ( if odds of success is greater than 1, it is more likely that males will respond )**

**What is logit ?**

It is the natural log of an odds ratio; often called a log odds

## Parameter Estimation

**Estimation of Logistic:**

The parameters for the logistic regression are estimated using a technique known as Maximum Likelihood Estimation (MLE)

It's a popular method of estimation

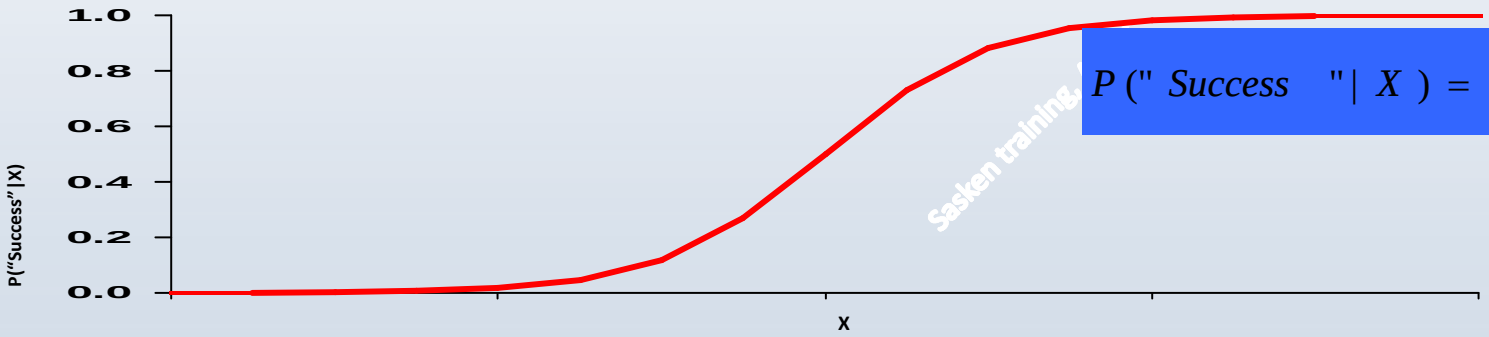
- It doesn't have any underlying assumptions of distribution
- When the underlying distribution of error terms is normal, MLE estimates are similar to OLS estimates

## Logistic Regression

So what we have on the Y axis is a probability

Prob of churners (Y = 1) = β0 + β1 * total.day.minutes + e

What can be the values of Y ?




$$P(\text{"Success"} \mid X) = \frac{e^{\beta_o + \beta_1 X}}{1 + e^{\beta_o + \beta_1 X}}$$

## Why not Linear Regression ?

Why can't we use linear regression model to solve classification problem ?

Probability values are restricted to 0 and 1

p / (1-p) – can take values of 0 to ∞, log(p/1-p) can take values between - ∞ to + ∞

In order to solve this take an odds ratio (p/(1-p)) and then take the log

Why do we need to take log ?

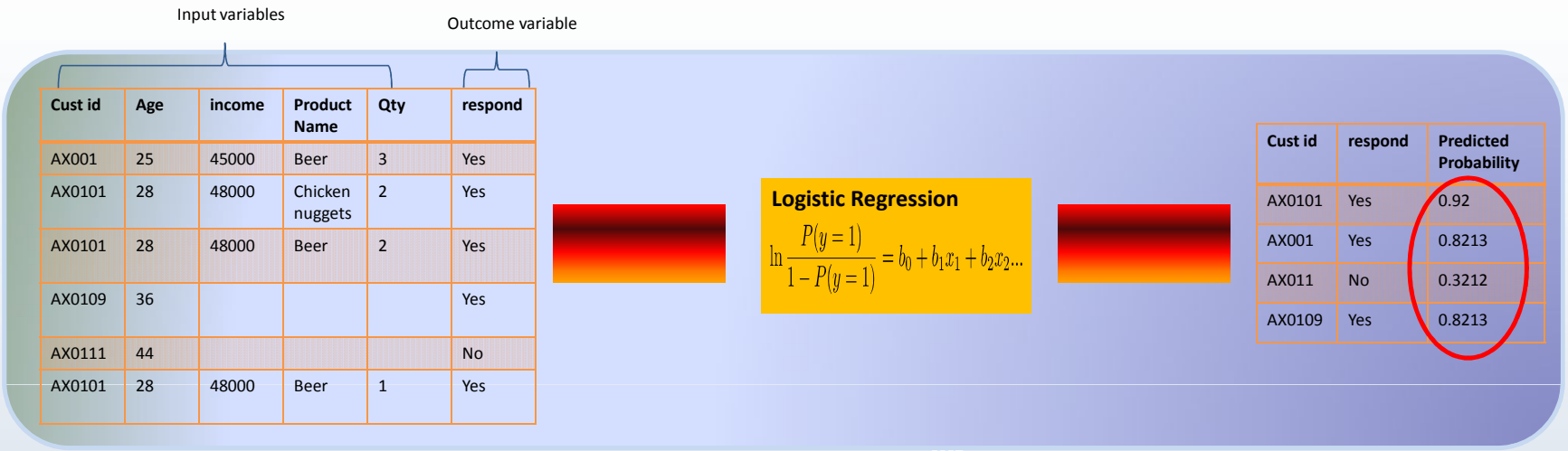
How to Convert log odds ratio to odds ratio ?

exp(logit) = odds ratio

How to convert odds ratio to Probability ?

probability = odd ratio / (1 + odd ratio)

# How exactly Logistic Regression works ?




| Cust id | Age | income | Product Name | Qty | respond |
|---|---|---|---|---|---|
| AX001 | 25 | 45000 | Beer | 3 | Yes |
| AX0101 | 28 | 48000 | Chicken nuggets | 2 | Yes |
| AX0101 | 28 | 48000 | Beer | 2 | Yes |
| AX0109 | 36 | | | | Yes |
| AX0111 | 44 | | | | No |
| AX0101 | 28 | 48000 | Beer | 1 | Yes |

| Cust id | respond | Predicted Probability |
|---|---|---|
| AX0101 | Yes | 0.92 |
| AX001 | Yes | 0.8213 |
| AX011 | No | 0.3212 |
| AX0109 | Yes | 0.8213 |

If you would like to classify which all customers will respond to your promotional campaign , the algorithm can help classifying the customers as either Yes or No

In this case, the algorithm assigned the predicted score (probability) to all the employees

# Logistic Regression - Assumptions

- Binary logistic regression requires the **dependent variable to be binary** and ordinal logit requires the dependent variable to be ordinal

- The error terms needs to be independent. Logit requires each observation to be independent

- It assumes linearity of independent variables and log odds.

  Whilst it does not require the dependent and independent variables to be related linearly, it requires that the independent variables are linearly related to the log odds. Otherwise the test underestimates the strength of the relationship and rejects the relationship too easily, that is being not significant (not rejecting the null hypothesis) where it should be significant. A solution to this problem is the categorization of the independent variables. That is transforming metric variables to ordinal level and then including them in the model.

- Lastly, it requires quite large sample sizes. Because maximum likelihood estimates are less powerful than ordinary least squares (e.g., simple linear regression, multiple linear regression); whilst OLS needs 5 cases per independent variable in the analysis, ML needs at least 10 cases per independent variable, some statisticians recommend at least 30 cases for each parameter to be estimated.

- Dependent variables do not need to be normally distributed

# Model Predictions

**Predictions**

We can predict values using either the “fitted.values” or predict function with type=”response”.

The output we get is predicted probabilities and not discrete values(0/1). For example, if the actual value of churn =1 and we get predicted output as 0.8 , then this can be interpreted as : We are 80% confident that the probability of a customer will churn is 1

head(lft2$fitted.values)

```
> head(lft2$fitted.values)
         1          2          3          4          6          7
0.08061805 0.02713147 0.05455481 0.62843130 0.36010495 0.21640418
```

# Model Diagnostic - Confusion Matrix

| | | Predicted | |
|---|---|---|---|
| | | Good | Bad |
| Actual/Observed | Good | True Positive (d) | False negative (c) |
| | Bad | False Positive (b) | True Negative (a) |

Accuracy = $\frac{\text{True Positive + True Negative}}{\text{TP+TN+FP+FN}}$

TPR (Sensitivity) = $\frac{\text{True Positive}}{\text{True Positive + False Negative}}$

TNR (Specificity) = $\frac{\text{True Negative}}{\text{True Negative + False Positive}}$

FPR (1 – Specificity) = $\frac{\text{False Positive}}{\text{True Negative + False positive}}$

Precision = $\frac{\text{True Positive}}{\text{True Positive + False Positive}}$

# Logistic Regression – Customer Churn Case Study

- Business Problem
- Variable Descriptions
- Exploratory Data Analysis
- Data Pre-processing
- Model Development
- Model Validation

## Churn Analysis Case Study – Business Problem Statement

customer churn refers to a decision made by the customer about ending the business relationship. It is also referred as loss of clients or customers. Customer loyalty and customer churn always add up to 100%. If a firm has a 60% of loyalty rate, then their loss or churn rate of customers is 40%. As per 80/20 customer profitability rule, 20% of customers are generating 80% of revenue. So, it is very important to predict the users likely to churn from business relationship and the factors affecting the customer decisions. In this case study, we are going to demonstrate how logistic regression model using R can be used to identify the customer churn in the telecom dataset.

Some of the available variables are

- Age
- Job
- marital

Predicted variable would be

churn

- 0 : Customer will not churn
- 1 : customer will churn

## Business Context and Problem Statement

- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

```
features= read.csv("D:/dat1/churndata.csv", skip=4, header=FALSE,sep=":",
 colClasses=c("character", "NULL"))[[1]] features
churnDf = read.csv("D:/dat1/churnTel.csv", header=FALSE, col.names=c(features,"churn"))
# Feature engineering
# Exclude unused features
 churnDf = churnDf[,! names(churnDf) %in% c("state","area.code","phone.number") ]
 churnDf <- na.omit(churnDf)
# Perform recoding
 churnflg <- as.numeric(churnDf$churn) churnDf$churn <- ifelse(churnflg==2,0,I
  felse(churnflg==3,1,NA))
```

```
# Split the data into training and test dataset
# set.seed(2000)
library("caret")
ind<-createDataPartition(y = churnDf$churn,times = 1,p = 0.6,list = FALSE)
# ind <- createDataPartition(churnDf$churn,p=0.6,list=FALSE)
train <- churnDf[ind,]
test <- churnDf[-ind,]
dim(train)
```

## Business Context and Problem Statement

- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

- What proportion of customers are leaving?
- Where are the terminations occurring?

## Business Context and Problem Statement

- Import the data
- Split the dataset
- Exploratory Analysis
- **Perform Model Selection & Fit the model**
- Model Validation
- Perform Prediction

**Model Selection**

```
churn_logit_fit <- glm(churn~.,data=train,family = "binomial")
# Perform step wise regression to select the features
step(churn_logit_fit)
```

**# Refit the model with the features recommended by step wise regression**

```
churn_logit_fit2 <- glm(formula = churn ~ total.day.charge + total.eve.minutes +
total.night.charge + total.intl.minutes + total.intl.charge + number.customer.service.calls +
international.plan.ip + voice.mail.plan.ip, family = "binomial", data = train)
summary(churn_logit_fit2)
```

## Business Context and Problem Statement

- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

pred <- ifelse(predict_logit >.7,1,0) # install.packages("caret",dependencies=T) library(caret) confusionMatrix(data=factor(pred), reference=factor(test$churn), positive='1')

```
##  Confusion Matrix and Statistics
##
##                Reference
##  Prediction     0     1
##              0 638    86
##              1   6     6
##
##                    Accuracy : 0.875
##                      95% CI : (0.8489,  0.898
##         No Information Rate : 0.875
##         P-Value [Acc > NIR] : 0.5278
##
##                     Kappa : 0.0891
##   Mcnemar's Test P-Value : <2e-16
##
##                Sensitivity : 0.065217
##                Specificity : 0.990683
##             Pos Pred Value : 0.500000
##             Neg Pred Value : 0.881215
##                 Prevalence : 0.125000
##             Detection Rate : 0.008152
##       Detection Prevalence : 0.016304
##          Balanced Accuracy : 0.527950
##
##           'Positive' Class : 1
##
```

## Business Context and Problem Statement

- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

```
predict_logit <- predict(churn_logit_fit,test,type="response")
class = predict_logit > .5
summary(class)

predict_logit_fit2 <- predict(churn_logit_fit2,test,type="response")
library(ROCR)
pred  <- prediction(predict_logit_fit2,test$churn)
perf  <- performance(pred,"tpr","fpr")
plot(perf)
```

## Model Summary & Interpretation

- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

```
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                 -7.450e+00  1.198e+00  -6.220 4.96e-10 ***
account.length              -1.005e-03  2.374e-03  -0.423   0.6722
number.vmail.messages        5.463e-02  2.990e-02   1.827   0.0677 .
total.day.minutes            5.769e-01  5.528e+00   0.104   0.9169
total.day.calls             -9.959e-04  4.692e-03  -0.212   0.8319
total.day.charge            -3.327e+00  3.252e+01  -0.102   0.9185
total.eve.minutes            3.002e+00  2.871e+00   1.046   0.2957
total.eve.calls              5.606e-03  4.790e-03   1.170   0.2418
total.eve.charge            -3.523e+01  3.377e+01  -1.043   0.2969
total.night.minutes          2.025e-01  1.475e+00   0.137   0.8908
total.night.calls           -9.119e-03  4.831e-03  -1.887   0.0591 .
total.night.charge          -4.417e+00  3.279e+01  -0.135   0.8928
total.intl.minutes          -9.923e+00  8.848e+00  -1.122   0.2621
total.intl.calls            -7.519e-02  4.147e-02  -1.813   0.0698 .
total.intl.charge            3.706e+01  3.276e+01   1.131   0.2580
number.customer.service.calls 5.217e-01  6.523e-02   7.997 1.28e-15 ***
international.plan.ip        1.967e+00  2.318e-01   8.485  < 2e-16 ***
voice.mail.plan.ip          -2.523e+00  9.834e-01  -2.566   0.0103 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1005.02  on 1287  degrees of freedom
Residual deviance:  793.76  on 1270  degrees of freedom
AIC: 829.76
```

- intercept is the odds of a success when x = 0
- For every unit increase of total.day.minutes, ln(p/1-p) of being churned to increases by .57
- For a one unit increase in international plan, the log odds of being churned to international plan by 1.96 (exp(1.96) => 7.09 / (7.09+1) => .87 probability ), that means the chances of increased international plan leads to higher probability of churn
- For a one unit increase in voice.mail.plan changes the log odds of churn by - .02523 .

# Model Summary

- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

```
                               Estimate Std. Error z value Pr(>|z|)
(Intercept)                  -7.450e+00  1.198e+00  -6.220 4.96e-10 ***
account.length               -1.005e-03  2.374e-03  -0.423   0.6722
number.vmail.messages         5.463e-02  2.990e-02   1.827   0.0677 .
total.day.minutes             5.769e-01  5.528e+00   0.104   0.9169
total.day.calls              -9.959e-04  4.692e-03  -0.212   0.8319
total.day.charge             -3.327e+00  3.252e+01  -0.102   0.9185
total.eve.minutes             3.002e+00  2.871e+00   1.046   0.2957
total.eve.calls               5.606e-03  4.790e-03   1.170   0.2418
total.eve.charge             -3.523e+01  3.377e+01  -1.043   0.2969
total.night.minutes           2.025e-01  1.475e+00   0.137   0.8908
total.night.calls            -9.119e-03  4.831e-03  -1.887   0.0591 .
total.night.charge           -4.417e+00  3.279e+01  -0.135   0.8928
total.intl.minutes           -9.923e+00  8.848e+00  -1.122   0.2621
total.intl.calls             -7.519e-02  4.147e-02  -1.813   0.0698 .
total.intl.charge             3.706e+01  3.276e+01   1.131   0.2580
number.customer.service.calls 5.217e-01  6.523e-02   7.997 1.28e-15 ***
international.plan.ip         1.967e+00  2.318e-01   8.485  < 2e-16 ***
voice.mail.plan.ip           -2.523e+00  9.834e-01  -2.566   0.0103 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1005.02  on 1287  degrees of freedom
Residual deviance:  793.76  on 1270  degrees of freedom
AIC: 829.76
```

**Null Deviance and Residual Deviance :**
Null Deviance indicates the response predicted by a model with nothing but an intercept. Lower the value, better the model. Residual deviance indicates the response predicted by a model on adding independent variables. Lower the value, better the model.

**AIC (Akaike Information Criteria) –** The analogous metric of adjusted $R^2$ in (linear regression) is AIC. AIC is a measure of fit which penalizes model for the number of model coefficients. Therefore, we always prefer model with minimum AIC value.

Fisher Scoring iterations is a derivative of Newton-Raphson algorithm which proposes how the model was estimated. It just confirms the model convergence. It can be interpreted as it took 6 iterations to perform the fit

# Model Validation

- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

```
    Min      1Q  Median      3Q     Max
-1.7200  -0.5032  -0.3412  -0.2223  3.0370

Coefficients:
                                Estimate Std. Error z value Pr(>|z|)
(Intercept)                    -6.678502   0.692879  -9.639  < 2e-16 ***
total.day.minutes               0.011266   0.001721   6.547 5.87e-11 ***
total.eve.minutes               0.004865   0.001807   2.693  0.00709 **
total.night.charge              0.067525   0.039619   1.704  0.08831 .
number.customer.service.calls   0.518902   0.065541   7.917 2.43e-15 ***
international.plan.ip           1.893722   0.233962   8.094 5.77e-16 ***
voice.mail.plan.ip             -0.964759   0.238057  -4.053 5.06e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1001.25  on 1287  degrees of freedom
Residual deviance:  811.74  on 1281  degrees of freedom
AIC: 825.74
```

- Here the DV= churn (whether a customer will churn or not)
- Following parameters we get from the summary of “myresult”
- **Estimate** :
  - Tells us by how the IDV is influencing the DV(churn)
  - For Example : When daily minutes increases by 1 unit then the log of odds increases by 11% .This is because – we are predicting the log of odds ratio and not the DV directly
- **Pr**: This is the p value. Tells us whether the variable is significant or not.
- **Deviances**
  - **Null deviance**: Gives the model Error when No predictor variables (IDV’s ) are involved
  - **Residual deviance**: The model Error when predictors are taken into account
- **AIC**:It’s the model performance metric. Lower the value, the better the model is and also it’s a less complex model (with optimum no of variables/right number of variables)

# Area under the curve (AUC)

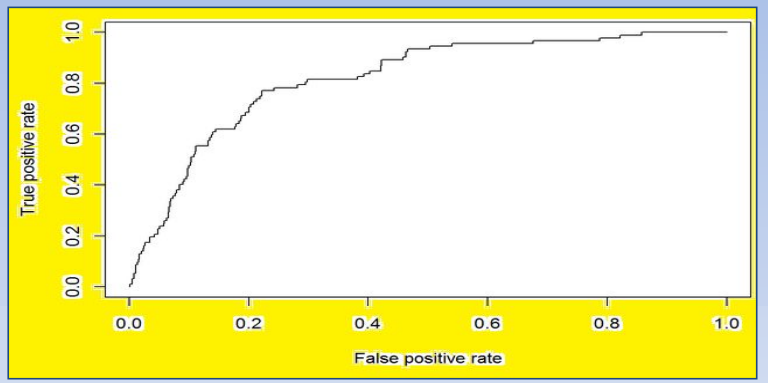
- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

- Plot the ROC curve
- Choose a point which has high TPR and low FPR.
- Choose the model with highest AUC
- Check the accuracy using confusionMatrix which is part of caret package

```
pred <- ifelse(predict_logit >.7,1,0)

# install.packages("caret",dependencies=T)
library(caret)
confusionMatrix(data=factor(pred),
                reference=factor(test$churn),
                positive='1')
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction   0   1
##          0 638  86
##          1   6   6
##
##                Accuracy : 0.875
##                  95% CI : (0.8489, 0.898)
##     No Information Rate : 0.875
##     P-Value [Acc > NIR] : 0.5278
##
##                   Kappa : 0.0891
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.065217
##             Specificity : 0.990683
##          Pos Pred Value : 0.500000
##          Neg Pred Value : 0.881215
##              Prevalence : 0.125000
##          Detection Rate : 0.008152
##    Detection Prevalence : 0.016304
##       Balanced Accuracy : 0.527950
##
##        'Positive' Class : 1
##
```




**Higher the AUC , the better the model**

**If the AUC value is closer to 1 is better, in this case we could see that the auc is .89**

# Confusion Matrix - Interpretation

- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

| | | Predicted | |
|---|---|---|---|
| | | Good | Bad |
| Actual/Observed | Good | True Positive (d) | False negative (c) |
| | Bad | False Positive (b) | True Negative (a) |

Accuracy = $\frac{\text{True Positive + True Negative}}{\text{TP+TN+FP+FN}}$

TPR (Sensitivity) = $\frac{\text{True Positive}}{\text{True Positive + False Negative}}$

TNR (Specificity) = $\frac{\text{True Negative}}{\text{True Negative + False Positive}}$

FPR (1 – Specificity) = $\frac{\text{False Positive}}{\text{True Negative + False positive}}$

Precision = $\frac{\text{True Positive}}{\text{True Positive + False Positive}}$

# Kappa Statistic & Regression Assumptions

- Import the data
- Split the dataset
- Exploratory Analysis
- Perform Model Selection & Fit the model
- Model Validation
- Perform Prediction

The **kappa statistic (labeled Kappa in the previous output) adjusts accuracy by**

- Accounting for the possibility of a correct prediction by chance alone. Kappa values range to a maximum value of 1, which indicates perfect agreement between the model's predictions and the true values—a rare occurrence. Values less than one indicate imperfect agreement.

**interpretation of Kappa**

- Poor agreement = Less than 0.20
- Fair agreement = 0.20 to 0.40
- Moderate agreement = 0.40 to 0.60
- Good agreement = 0.60 to 0.80
- Very good agreement = 0.80 to 1.00

Formula to obtain Kappa Statistic

$$k = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}$$

**Validating Assumptions**

Linearity – The regression equation should have a linear relationship with the logit form of the Dependent variables

Absence of multicollinearity

## Logistic Regression – Takeaways

- Linear regression wouldn't be an appropriate model to predict binary variables as the predictor variables can range from -infinity to +infinity, while the binary variable would be 0 or 1.
- The odds can range from 0 to infinity.
- The variable coefficients are calculated using the maximum Log-likelihood estimate. The roots of the equation are often calculated using the Newton- Raphson method.
- Each coefficient estimate has a Wald statistic and p-value associated to it. The smaller the p-value, the more significant the variable coefficient is to the model.
- The model can be validated using the k-fold cross validation technique, wherein the logistic regression model is run k-times using the testing and training data derived from the overall dataset.
- The model predicts the probability for each observation. A threshold probability value is defined to categorize the probability values as 0 (failures) and 1 (successes).
- Sensitivity measures what proportion of successes were actually identified as successes, while Specificity measures what proportion of failures were actually identified as failures.

# Feature (Variable) Selection
# (Dimension Reduction Techniques)

## Feature selection – Stepwise regression for both (Linear and Logistic Regression)

- There are two main types of stepwise procedures in regression:
- Backward elimination: eliminate the least important variable from the selected ones.
- Forward selection: add the most important variable from the remaining ones.
- A hybrid version that incorporates ideas from both main types: alternates backwards and forwards steps, and stops when all variables have either been retained for inclusion or removed.
- An exhaustive search for the subset may not be feasible if *p* is very large. There are two main alternatives:

**Forward stepwise selection:**

- First we approximate the response variable y with a constant (i.e., an intercept-only regression model).
- Then we gradually add one more variable at a time (or add main effects first, then interactions).
- Every time we always choose from the rest of the variables the one that yields the best accuracy in prediction when added to the pool of already selected variables. This accuracy can be measured by the F-statistic, LRT, AIC, BIC, etc.
- For example, if we have 10 predictor variables, first we would approximate *y* with a constant, and then use one variable out of the 10 (I would perform 10 regressions, each time using a different predictor variable; for every regression I have a residual sum of squares; the variable that yields the minimum residual sum of squares is chosen and put in the pool of selected variables). We then proceed to choose the next variable from the 9 left, etc.

**Backward stepwise selection:**

- This is similar to forward stepwise selection, except that we start with the full model using all the predictors and gradually delete variables one at a time.
- There are various methods developed to choose the number of predictors, for instance the *F*-ratio test. We stop forward or backward stepwise selection when no predictor produces an *F*-ratio statistic greater than some threshold.

## Feature selection – Stepwise regression for both (Linear and Logistic Regression)

**Limitations of Step wise regression**

- There is no guarantee that the subsets obtained from stepwise procedures will contain the same variables or even be the "best" subset.
- When there are more variables than observations ($p > n$), backward elimination is typically not a feasible procedure.
- The maximum or minimum of a set of correlated *F* statistics is not itself an *F* statistic.
- It produces a single answer (a very specific subset) to the variable selection problem, although several different subsets may be equally good for regression purposes.
- The computing is easy by the use of R function step() or regsubsets(). However, to specify a practically good answer, you must know the practical context in which your inference will be used.
- Scott Zeger on 'how to pick the wrong model': Turn your scientific problem over to a computer that, knowing nothing about your science or your question, is very good at optimizing AIC, BIC, .

## WOE and Information Value

One of our goals when binning variables is to maximize Information Value. Weight of Evidence (WoE) for single bin is defined as:

The *weight of evidence* (WOE) and *information value* (IV) provide a great framework for exploratory analysis and variable screening for binary classifiers.
WOE and IV have been used extensively in the credit risk world for several decades, and the underlying theory dates back to the 1950s. However, it is still not widely used outside the credit risk world and it is a somewhat underserved area in R.

## WOE and IV

**WOE and IV enable one to:**

- Consider each variable's independent contribution to the outcome.
- Detect linear and non-linear relationships.
- Rank variables in terms of "univariate" predictive strength.
- Visualize the correlations between the predictive variables and the binary outcome.
- Seamlessly compare the strength of continuous and categorical variables without creating dummy variables.
- Seamlessly handle missing values without imputation.
- Assess the predictive power of missing values.

## WOE and IV

**Steps of Calculating WOE -** For a continuous variable,

- split data into 10 parts (or lesser depending on the distribution).
- Calculate the number of events and non-events in each group (bin)
- Calculate the % of events and % of non-events in each group.
- Calculate WOE by taking natural log of division of % of non-events and % of events

**Steps of Calculating WOE -** For a categorical independent variable,

- Combine categories with similar WOE and then create new categories of an independent variable with continuous WOE values
- In other words, use WOE values rather than raw categories in your model. The transformed variable will be a continuous variable with WOE values. It is same as any continuous variable.Calculate the % of events and % of non-events in each group.

## WOE and IV

If the IV statistic is:

- Less than 0.02, then the predictor is not useful for modeling (separating the Goods from the Bads)
- 0.02 to 0.1, then the predictor has only a weak relationship to the Goods/Bads odds ratio
- 0.1 to 0.3, then the predictor has a medium strength relationship to the Goods/Bads odds ratio
- 0.3 to 0.5, then the predictor has a strong relationship to the Goods/Bads odds ratio.
- > 0.5, suspicious relationship (Check once)

## Model Validation

While validating, we use estimates obtained from the development sample to check robustness in any other sample. We aggregate our findings at the decile level for ease of interpretation

- ROC curve - ROCR curve means Receiver Operator Characteristics Curve
  This is a model performance metric. It is the area under the curve.
  This is obtained from the "performance" function where you can choose the option "AUC".
  In the ROCR the higher the value of TPR, the greater will the area under the curve (it is obvious if you observe the graph).
  Generally a good logistic model has an AUC>70%

- Gain curve – It is the most popular technique among marketers for evaluating and comparing the model fit. At each decile, it demonstrates the mode's power. Our built model i.e., development and validation sample model with predictors is compared with a Random model