Introduction to Logistic Regression

Still Classification Problem

- Given a set of features (age, gender, BMI, BP, etc.), a person has diabetes (1) or not (0)?

- Given the temperature, the age of equipment, and other features, classify if the equipment will get failure (1) or not (0) in the coming working round?

- Given some features, a student gets admitted into NUS (1) or not (0)?

Notations

- Assume we have a set of n observations used to build a model.
- We have p features, $X_1, ..., X_p$ in general.
- The outcome variable Y is binary with two values, 0 and 1.

Obs	X_1	X_2	$\cdots$	X_p	Y
1	x_{11}	x_{21}	$\cdots$	x_{p1}	y_1
2	x_{21}	x_{22}	$\cdots$	x_{p1}	y_2
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
n	x_{n1}	x_{2n}	$\cdots$	x_{pn}	y_n

Binary Response

- Assume a point with known features, denote

$$P(Y = 1) = p.$$

- We may have a linear regression model for p:

$$p \sim \beta_0 + \beta_1 X_1 + ... + \beta_p X_p.$$

- However, the fitted value might be negative or more than 1, which is not possible for a probability.

- Instead of forming a model for p, we can form a model for a function of p!

Logistic Model

- If we assume $Y = 1$ as a success, then $P(Y = 1) = p$ is the success probability. The *odds of success* is then defined as

$$\frac{p}{1-p}.$$

- We then consider model for the log-odds, or called "logit":

$$\log(\frac{p}{1-p}) = \beta_0 + \beta_1 X_1 + ... + \beta_p X_p.$$

- Keeping other features constant, for each unit increased in X_1, the **log odds** increases by β_1.

- This is a type of **generalized linear model (GLM)**.

Logistic Model

- From the logit equation, we can have the equivalent version:

$$p = \frac{e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}.$$

- Regardless the values of the features, the range for p is always between 0 and 1.

Logistic Model

- Just like linear regression, in logistic regression the parameters $\beta_0, \beta_1, ..., \beta_p$ need to be estimated based on the training data.

- Instead of the method of ordinary least squares (OLS), parameter estimation in logistic regression is based on the method called Maximum Likelihood Estimation (MLE).

- In our course, we'll not introduce the details of MLE.

Example: Customer Churn

- A wireless telecommunications company wants to predict whether a customer will churn (switch to a different company) in the next six months.
- With a reasonably accurate prediction of a person's churning, the sales and marketing groups can attempt to retain the customer by offering various incentives.
- Data on 8,000 current and prior customers was obtained. The variables collected for each customer follow:
  (i) Age (years)
  (ii) Married (true/false)
  (iii) Duration as a customer (years)
  (iv) Churned contacts—Number of the customer's contacts that have churned (count)
  (v) Churned (true/false)—Whether the customer churned

Example: Customer Churn

```
> churn = read.csv("C:/Data/churn.csv")
> churn[1:3,]
  ID Churned Age Married Cust_years Churned_contacts
1  1       0  61       1          3                1
2  2       0  50       1          3                2
3  3       0  47       1          2                0
> churn$Churned = as.factor(churn$Churned)
> churn$Married = as.factor(churn$Married)
> churn= churn[,-1] # Remove ID column
> attach(churn)
> #table(Churned)
> prop.table(table(Churned))
Churned
       0        1
0.782125 0.217875
```

- About 21.8% of the customers churned in the given data.

Example: Customer Churn

- Logistic regression can be performed using the Generalized Linear Model function, **glm()** in R.

- Specify the family to be binomial, the logit link is set as the default.

```
> M1<- glm( Churned ~., data =churn,
+                 family = binomial(link ="logit"))
```

```
> summary(M1)
Call:
glm(formula = Churned ~ ., family = binomial(link = "logit"),
    data = churn)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4542  -0.5206  -0.1971  -0.0728   3.3786

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)       3.415201   0.163734  20.858   <2e-16 ***
Age              -0.156643   0.004088 -38.320   <2e-16 ***
Married1          0.066432   0.068302   0.973    0.331
Cust_years        0.017857   0.030497   0.586    0.558
Churned_contacts  0.382324   0.027313  13.998   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8387.3  on 7999  degrees of freedom
```

p-value of a Coefficient

- In a linear regression model, the column 'Pr(>|Z|)' indicate the p-value for a test to test the significance of the coefficient in the fitted model.

- Similarly, in a logistic model, we'll have p-value for each coefficient in the last column in the table 'Coefficients'.

- A large p-value means the contribution of the coefficient (equivalently, of the feature) to the model is not significant.

- It's optional to drop or to keep an insignificant feature in the model. However, dropping it will simplify the model.

Example: Customer Churn

- From the initial model, 'Cust_years' is most insignificant. We can drop it.
- Re-fit the logistic model without 'Cust_years'. We have model M2.

```
> M2<- glm( Churned ~ Age + Married + Churned_contacts,
+                data = churn, family = binomial)
```

```
> summary(M2)
Call:
glm(formula = Churned ~ Age + Married + Churned_contacts, family = b
    data = churn)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4476  -0.5178  -0.1972  -0.0723   3.3776

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)       3.472062   0.132107  26.282   <2e-16 ***
Age              -0.156635   0.004088 -38.318   <2e-16 ***
Married1          0.066430   0.068299   0.973    0.331
Churned_contacts  0.381909   0.027302  13.988   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8387.3  on 7999  degrees of freedom
Residual deviance: 5358.3  on 7996  degrees of freedom
```

Example: Customer Churn

- The p-value of 'Married' in model M2 is quite large (0.331), it indicates that 'Married' doesn't contribute significantly to the model when predicting the response.

- We consider to drop it and simplify the model to only two features, model M3.

```
> M3<- glm( Churned ~ Age + Churned_contacts,
+                  data = churn, family = binomial)
```

```
> summary(M3)
Call:
glm(formula = Churned ~ Age + Churned_contacts, family = binomial,
    data = churn)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4599  -0.5214  -0.1960  -0.0736   3.3671

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)       3.502716   0.128430   27.27   <2e-16 ***
Age              -0.156551   0.004085  -38.32   <2e-16 ***
Churned_contacts  0.381857   0.027297   13.99   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8387.3  on 7999  degrees of freedom
Residual deviance: 5359.2  on 7997  degrees of freedom
AIC: 5365.2
```

Example: Customer Churn

- The fitted model M3 is then

$$\hat{p} = \frac{e^{3.5-0.157\,\mathsf{A}+0.382\,\mathsf{C}}}{1+e^{3.5-0.157\,\mathsf{A}+0.382\,\mathsf{C}}}$$

where A stands for `Age` and C stands for `Churned_contacts`.

- We then can predict for a customer who is 50 years old with 5 churned contacts, the estimate probability of churning is

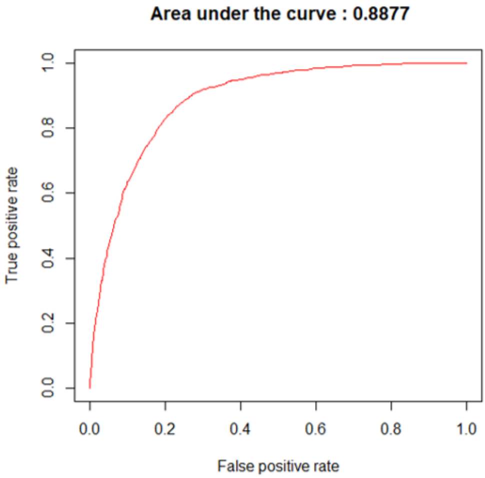
$$\hat{p} = \frac{e^{3.5-0.157\times 50+0.382\times 5}}{1+e^{3.5-0.157++0.382\times 5}} = 0.08.$$

- We predict the outcome Y be 0 or 1 based on a threshold, such as 0.5.

- ROC curve can be constructed for logistic model.

```
> library(ROCR)
> prob = predict(M3, type ="response")
> # above is to predict probability Pr(Y = 1)
> #for each point in the training data set, using M3
>
> pred = prediction(prob , Churned )
> roc = performance(pred , "tpr", "fpr")
> auc = performance(pred , measure ="auc")
> auc@y.values[[1]] # gives value of AUC
[1] 0.8876509
> plot(roc , col = "red",
+      main = paste(" Area under the curve :",
+      round(auc@y.values[[1]] ,4)))
>
```

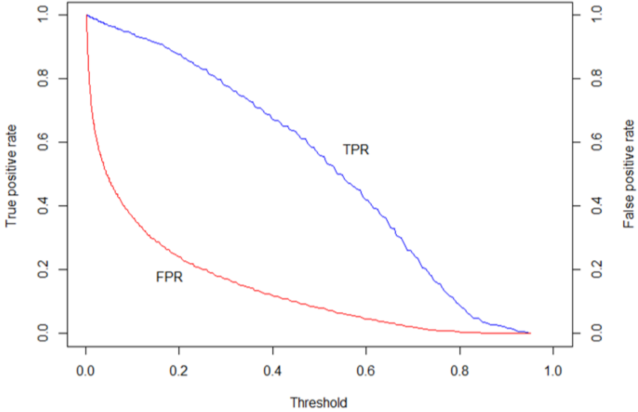
ROC and AUC



This ROC curve is created based on 328 different values of threshold.

How TPR, FPR Change when Threshold Changes?

- We can plot to see how TPR and FPR change along with threshold.



For you to try

- In the example above, we used the whole data set as training data. After that, we evaluated the model (M3) by comparing the real response versus the predict response using M3.

- Can you try to split the full data set into two sets: train set and test set; then build a logistic model on the train set; then check the goodness of the model (using ROC and AUC) using the test set?