````
---
title: "Logistic Reggression"
output: html_notebook
---

Load Data

```{r}
path <- "/Users/pulkitbatra/Desktop/CACSC19/Unit-2 R Programming/Learning
R/Assignment/Placement_Data_Full_Class.csv"



library(dplyr)
library(ggplot2)
location <- "../input/factors-affecting-campus-placement/Placement_Data_Full_Class.csv"
placement.df <- read.csv(path)
# select only relevant columns
placement.lr <- placement.df %>% select(ends_with("_p"), -etest_p, status)
table(placement.lr$status)

placement.lr$status <- ifelse(placement.lr$status == "Not Placed", 1, 0)
table(placement.lr$status)

```

```{r}

library(caTools)
```



```{r}
# Train and Test data
library(caTools) # to split data into train and test
set.seed(101)
sample <- sample.split(placement.lr$status, SplitRatio = 0.80)
train.lr = subset(placement.lr, sample == TRUE)
test.lr = subset(placement.lr, sample == FALSE)
#check the splits
prop.table(table(train.lr$status))
prop.table(table(test.lr$status))
```

```{r}
# Train the model
model.lr <- glm(status ~ degree_p, family = binomial, data = train.lr)
summary(model.lr)
```

```{r}
# prediction
lr.pred <- predict(model.lr, newdata = test.lr, type = "response")
head(lr.pred)
# The probabilities always refer to the class dummy-coded as “1”
head(test.lr$status)
```



```{r}
# Classification Table
# categorize into groups based on the predicted probability
lr.pred.class <- ifelse(lr.pred>=0.5, 1, 0)
head(lr.pred.class)
table(lr.pred.class)
table(test.lr$status)
conf.matrix <- table(test.lr$status, lr.pred.class)
conf.matrix
````
````

```
rownames(conf.matrix) <- c("Placed", "Not Placed")
colnames(conf.matrix) <- c("Placed", "Not Placed")
addmargins(conf.matrix)
```

```{r}
# model accuracy
mean((test.lr$status == lr.pred.class))
```

```{r}
# different cut-off
lr.pred.class1 <- ifelse(lr.pred>=0.35, 1, 0)
conf.matrix1 <- table(test.lr$status, lr.pred.class1)
conf.matrix1
```

Plots

```{r}
ggplot(data = test.lr, aes(x = degree_p, y = status)) +
  geom_point() +
  geom_line(aes(y = lr.pred), color = "blue") +
  labs(title = "Logistic Regression Decision Boundary",
       x = "degree_p",
       y = "Probability of Placement")

```

```{r}
install.packages("pROC")
library(pROC)
roc_curve <- roc(test.lr$status, lr.pred)
plot(roc_curve, main = "ROC Curve", col = "blue", lwd = 2)

```

```{r}
library(ggplot2)

# Convert confusion matrix to a data frame
conf_matrix_df <- as.data.frame.matrix(conf.matrix)
conf_matrix_df <- cbind(Actual = rownames(conf_matrix_df), conf_matrix_df)

# Reshape data for ggplot
conf_matrix_long <- tidyr::gather(conf_matrix_df, key = "Predicted", value =
"Frequency", -Actual)

# Create heatmap using ggplot2
ggplot(data = conf_matrix_long, aes(x = Predicted, y = Actual, fill = Frequency)) +
  geom_tile() +
  labs(title = "Confusion Matrix", x = "Predicted", y = "Actual") +
  scale_fill_gradient(low = "white", high = "blue") +
  theme_minimal()

```