```r
# Week 3, Modeling the Expert


# Video 4

# Read in dataset
quality = read.csv("quality.csv")

# Look at structure
str(quality)

# Table outcome
table(quality$PoorCare)

# Baseline accuracy
98/131

# Install and load caTools package
install.packages("caTools")
library(caTools)

# Randomly split data
set.seed(88)
split = sample.split(quality$PoorCare, SplitRatio = 0.75)
split

# Create training and testing sets
qualityTrain = subset(quality, split == TRUE)
qualityTest = subset(quality, split == FALSE)

# Logistic Regression Model
QualityLog = glm(PoorCare ~ OfficeVisits + Narcotics,
 data=qualityTrain, family=binomial)
summary(QualityLog)

# Make predictions on training set
predictTrain = predict(QualityLog, type="response")

# Analyze predictions
summary(predictTrain)
tapply(predictTrain, qualityTrain$PoorCare, mean)



# Video 5

# Confusion matrix for threshold of 0.5
table(qualityTrain$PoorCare, predictTrain > 0.5)
```

```
# Sensitivity and specificity
10/25
70/74

# Confusion matrix for threshold of 0.7
table(qualityTrain$PoorCare, predictTrain > 0.7)

# Sensitivity and specificity
8/25
73/74

# Confusion matrix for threshold of 0.2
table(qualityTrain$PoorCare, predictTrain > 0.2)

# Sensitivity and specificity
16/25
54/74



# Video 6

# Install and load ROCR package
install.packages("ROCR")
library(ROCR)

# Prediction function
ROCRpred = prediction(predictTrain, qualityTrain$PoorCare)

# Performance function
ROCRperf = performance(ROCRpred, "tpr", "fpr")

# Plot ROC curve
plot(ROCRperf)

# Add colors
plot(ROCRperf, colorize=TRUE)

# Add threshold labels
plot(ROCRperf, colorize=TRUE, print.cutoffs.at=seq(0,1,by=0.1),
 text.adj=c(-0.2,1.7))
```