Problem 3 Section 1

a) From the caret package optimum k was found to be K = 9

```
optimal.KNN <- train(quality ~ .,
                     data       = wine,
                     method     = "knn",
                     tuneLength = 50,
                     trControl  = trainControl(method = "cv", number = 10))
> optimal.KNN
k-Nearest Neighbors
Accuracy was used to select the optimal model using the largest value.
The final value used for the model was k = 9.
```

Misclassification rate	Sensitivity	Specificity	AUC	Estimated test error
0.1776235	0.85	0.6443769	0.745694	0.2139627

b) Logistic regression

Misclassification rate	Sensitivity	Specificity	AUC	Estimated test error
0.1976317	0.8264331	0.5916335	0.6133877	0.2000789

c) LDA

Misclassification rate	Sensitivity	Specificity	AUC	Estimated test error
0.19804	0.831484	0.5786713	0.6247355	0.2013075

d) QDA

Misclassification rate	Sensitivity	Specificity	AUC	Estimated test error
0.2476521	0.911571	0.455237	0.745694	0.2519394

e)

Method	AUC	Overall Misclassification rate	CV Test Error Rate
KNN	0.745694	0.1776235	0.2139627
GLM	0.6133877	0.1976317	0.2000789
LDA	0.6247355	0.19804	0.2013075
QDA	0.745694	0.2476521	0.2519394

AUC is high for both KNN and QDA. Lowest training and test errors are for KNN. Thus I would recommend KNN classifier.

Problem 3 Section 1

```
### problem 3.

library(caret) # for cross-validation
library(MASS) # for LDA and QDA
library(cvAUC) # for calculating AUC
library(class) # for knn

wine <- read.csv("winequality-white.csv", header = T, sep=';')
wine$quality <- ifelse(wine$quality >= 7, 1, 0)
wine$quality <- as.factor(wine$quality)
attach(wine)

### problem 3. a)

set.seed(1234)
optimal.KNN <- train(quality ~ .,
                     data       = wine,
                     method     = "knn",
                     tuneLength = 50,
                     trControl  = trainControl(method = "cv", number = 10))
#> optimal.KNN
# k-Nearest Neighbors
#
# 4898 samples
# 11 predictor
# 2 classes: '0', '1'
#
# No pre-processing
# Resampling: Cross-Validated (10 fold)
# Summary of sample sizes: 4408, 4409, 4408, 4408, 4408, 4409, ...
# Resampling results across tuning parameters:
#
#   k    Accuracy   Kappa
# 5  0.7837912  0.300339072
# 7  0.7807308  0.265270003
# 9  0.7860373  0.256348924
# 11  0.7811331  0.212080442
# ---------------------------
# 97  0.7825650  0.003279178
# 99  0.7829732  0.004089494
# 101  0.7831764  0.003416900
# 103  0.7829732  0.001974173
#
# Accuracy was used to select the optimal model using the largest value.
# The final value used for the model was k = 9.

fit.KNN <- knn(wine[,-12], wine[,-12], quality, k = 9)
mean(quality != fit.KNN)
# [1] 0.1776235

table(quality, fit.KNN)
#        fit.KNN
```

```
# quality    0    1
#       0 3604  234
#       1  636  424

#sensitivity
3604/(3604+636) # [1] 0.85
#specificity
424/(234+424) # [1] 0.6443769

AUC(pred, quality)
# [1] 0.745694

# Estimated test error rate is 0.256348924


### problem 3. b)
set.seed(1234)
fit.full.GLM.CARET <- train(quality ~ . ,
                            data = wine,
                            method ="glm",
                            trControl = trainControl(method = "cv", number = 10))
pred <- as.numeric(predict(fit.full.GLM.CARET, wine)) - 1
mean(quality != pred)
# [1] 0.1976317

table(quality, pred)
#        pred
# quality    1    2
#       0 3633  205
#       1  763  297

#sensitivity
3633/(3633+763) # [1] 0.8264331
#specificity
297/(205+297) # [1] 0.5916335

AUC(pred, quality)
# [1] 0.6133877

# Estimated test error rate is 0.273915

### problem 3. c)
set.seed(1234)
fit.full.LDA.CARET <- train(quality ~ . ,
                            data = wine,
                            method ="lda",
                            trControl = trainControl(method = "cv", number = 10))
pred <- as.numeric(predict(fit.full.LDA.CARET, wine)) - 1
mean(quality != pred)
# [1] 0.19804

table(quality, pred)
#        pred
```

```
# quality    1    2
#       0 3597  241
#       1  729  331

#sensitivity
3597/(3597+729) # [1] 0.831484
#specificity
331/(241+331) # [1] 0.5786713

AUC(pred, quality)
# [1] 0.6247355

# Estimated test error rate is 0.2876861


### problem 3. d)
set.seed(1234)
fit.full.QDA.CARET <- train(quality ~ . ,
                            data = wine,
                            method ="qda",
                            trControl = trainControl(method = "cv", number = 10))
pred <- as.numeric(predict(fit.full.QDA.CARET, wine)) - 1
mean(quality != pred)
# [1] 0.2476521

table(quality, pred)
#        pred
# quality    0    1
#       0 2907  931
#       1  282  778

#sensitivity
2907/(2907+282) # [1] 0.911571
#specificity
778/(931+778) # [1] 0.455237

AUC(pred, quality)
# [1] 0.745694

# Estimated test error rate is 0.3900707
```