STA6703 SML HW4

Christopher Marais

```
# load data
setwd(getwd())
data <- read.csv("SML.NN.data.csv")
train_data = data[data$set == 'train' | data$set == 'valid',]
test_data = data[data$set == 'test',]

# load MASS
library(MASS)
```

Import data and load libraries

Chapter 4

Question 5

5.a

Problem 1

1.a

Problem 2

Train models

```
L1 = glm(Y ~ 1 + X1 + X2,
        data=train_data,
        family=binomial)

summary(L1)
```

L1

```
##
## Call:
## glm(formula = Y ~ 1 + X1 + X2, family = binomial, data = train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0497  -0.9987  -0.9498   1.3715   1.4516
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.46905    0.08403  -5.582 2.38e-08 ***
## X1          -0.12055    0.14660  -0.822    0.411
## X2           0.05727    0.14406   0.398    0.691
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 799.75  on 599  degrees of freedom
## Residual deviance: 798.96  on 597  degrees of freedom
## AIC: 804.96
##
## Number of Fisher Scoring iterations: 4
```

```r
L2 = glm(Y ~ 1 + X1 + X2 + X1^2 + X2^2 + X1*X2,
        data=train_data,
        family=binomial)

summary(L2)
```

L2

```
##
## Call:
## glm(formula = Y ~ 1 + X1 + X2 + X1^2 + X2^2 + X1 * X2, family = binomial,
##     data = train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2289  -0.9986  -0.8985   1.3708   1.5306
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.45827    0.08440  -5.430 5.65e-08 ***
## X1          -0.12868    0.14745  -0.873    0.383
## X2           0.05857    0.14474   0.405    0.686
## X1:X2       -0.49418    0.25100  -1.969    0.049 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 799.75  on 599  degrees of freedom
## Residual deviance: 795.04  on 596  degrees of freedom
## AIC: 803.04
##
## Number of Fisher Scoring iterations: 4
```

```
D1 = lda(Y ~ X1 + X2,
        data=train_data)

D1
```

LDA

```
## Call:
## lda(Y ~ X1 + X2, data = train_data)
##
## Prior probabilities of groups:
##     0     1
## 0.615 0.385
##
## Group means:
##              X1          X2
## 0  0.004605339 -0.02707597
## 1 -0.033643963 -0.01087806
##
## Coefficients of linear discriminants:
##           LD1
## X1 -1.6168003
## X2  0.7681857
```

```
D2 = qda(Y ~ X1 + X2,
         data=train_data)

D2
```

QDA

```
## Call:
## qda(Y ~ X1 + X2, data = train_data)
```

```
##
## Prior probabilities of groups:
##     0     1
## 0.615 0.385
##
## Group means:
##             X1          X2
## 0  0.004605339 -0.02707597
## 1 -0.033643963 -0.01087806
```

Test models

```
MCR <- function(true_vals, pred_probs, threshold=0.5){
  if(length(true_vals)!=length(pred_probs)){
    print("ERROR: predictions and true values not of same shape")
  }else{
    pred_vals = as.integer((pred_probs > threshold))
    mcr = sum(pred_vals != true_vals)/length(true_vals)
    return(mcr)
  }
}
```

Misclassification rate function

```
L1_probs = data.frame(
              predict(L1,
                    test_data,
                    type ="response"
                    )
              )

MCR(
  true_vals=test_data$Y,
  pred_probs=L1_probs[,1],
  threshold=0.5)
```

L1

```
## [1] 0.325
```

```
L2_probs = data.frame(
              predict(L2,
                    test_data,
                    type ="response"
                    )
              )

MCR(
  true_vals=test_data$Y,
  pred_probs=L2_probs[,1],
  threshold=0.5)
```

L2

```
## [1] 0.335
```

```


D1_probs = data.frame(
                predict(D1,
                    test_data)$posterior[,2]
              )

MCR(
  true_vals=test_data$Y,
  pred_probs=D1_probs[,1],
  threshold=0.5)
```

LDA

```
## [1] 0.325
```

```


D2_probs = data.frame(
                predict(D2,
                    test_data)$posterior[,2]
              )



MCR(
  true_vals=test_data$Y,
  pred_probs=D2_probs[,1],
  threshold=0.5)
```

QDA

```
## [1] 0.09
```

Visualize decision boubndaries

L1

L2

LDA

QDA