Assignment 2

Zoe Zhou

20/04/2020

Task: We will use LDA to describe how countries of different “regimes” differ on a range of measures.

Question 1: Perform the linear discriminant analysis, using numeric variables only (not region or country labels) and use leave-1-out cross validation to evaluate how many discriminant functions are useful for separating the government types.

```
library(MASS)
data = read.csv("governments.csv")
government = data[, -(1:3)]
government$regime = government$regime + 1

predictions = matrix(NA, ncol = 5, nrow = nrow(government))
for (i in 1:nrow(government)){
  model = lda(regime ~., data = government[-i, ])
  for (j in 1:5){
    model.predict = predict(model, newdata = government[i, ], dimen = j)
    predictions[i, j] = model.predict$class
  }
}

# CV Prediction accuracy
apply(predictions, 2, function(x){sum(x == government$regime)/nrow(government)})
```

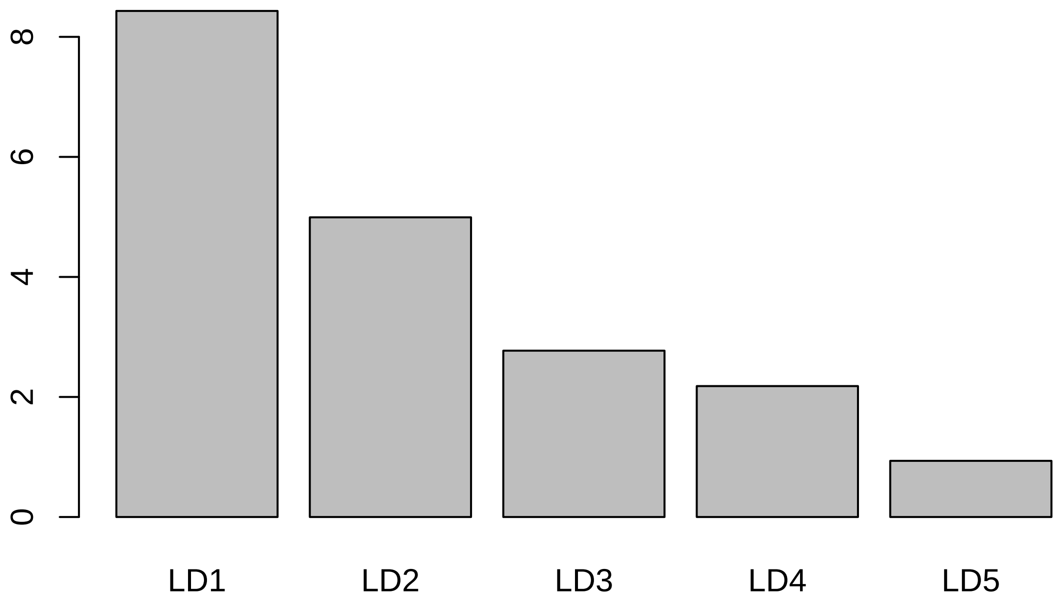
```
## [1] 0.4193548 0.5107527 0.5053763 0.5268817 0.4838710
```

Comment

The CV results showed that using 4 discriminant functions gives the highest accuracy (52.69%). But its just slightly better than using 2 functions (51.07%).

Check eigenvalues

```
# Fit with all data
gov.lda = lda(regime ~., data = government)
barplot(gov.lda$svd, names.arg = paste0("LD", 1:5))
```



```
gov.lda$svd/sum(gov.lda$svd)
```

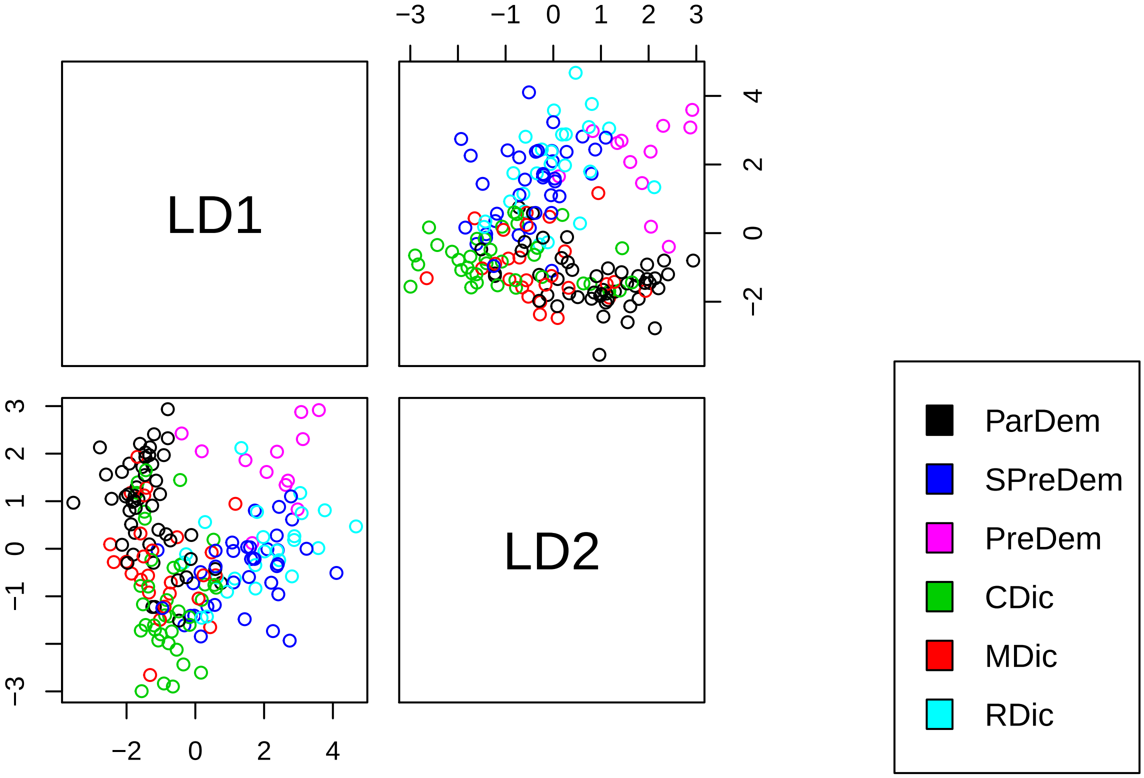
```
## [1] 0.43656388 0.25852465 0.14349660 0.11294519 0.04846968
```

The eigenvalues showed that LD1 and LD2 are contributing the most. We will use 2 LDs since LD3 and LD4 have very small eigenvalues.

Question 1 continued: Provide a pairs plot of the useful discriminant scores. Are the groups well separated? If not, which are hard to distinguish? Are any discriminant scores associated with separating particular government types?

```
gov.predict = predict(gov.lda)

pairs(gov.predict$x[, 1:2], col = government$regime, oma = c(3, 3, 3, 15))
par(xpd = TRUE)
legend("bottomright", fill = unique(government$regime),
       legend = c("ParDem", "SPreDem", "PreDem", "CDic", "MDic", "RDic"))
```



Comment:

1 = Parlimentary Democracy (eg New Zealand)- Black

2 = Semi-presidential democracy (eg France) - Blue

3 = Presidential democracy (eg USA) - Pink

4 = Civilian Dictatorship (eg Iran) - Green

5 = Military Dictatorship (eg Cuba) - red

6 = Royal Dictatorship (eg Qatar) - Baby Blue

The groups are not very well separated as we can see there are a lot of overlapping.

Relatively, only LD1 does a great job at telling the difference between 2 big clusters. Clusters of colors of Black green red on the left, clusters of blue light blue pink on the right.

Within these 2 big clusters, differences are hard to be distinguished. For example, Civilian Dictatorship and Military Dictatorship.

From the plot, We can conclude LD1 and LD2 together can roughly separate democracy and Dictatorship.

Question 2: Compute the correlation of the useful discriminant functions with the original variables. Display the output in readable form.

```
correlation = matrix(0, ncol = 2, nrow = 15)
colnames(correlation) = paste("variate", 1:2)
```

```r
rownames(correlation) = colnames(government)[-3]
for (i in 1:15){
  for (j in 1:2){
    correlation[i, j] = cor(gov.predict$x[, j], government[, i])
  }
}
class(correlation) = "loadings"
print(correlation, cutoff = 0.2)
```

```
##
## Loadings:
##            variate 1 variate 2
## bornyear    0.268
## tenure08    0.642
## agereg      0.764
## VAEstimate -0.213     0.464
## VAStdErr   -0.918     0.245
## PSEstimate -0.375     0.574
## PSStdErr   -0.350     0.491
## GEEstimate
## GEStdErr   -0.534     0.551
## RQEstimate -0.203     0.400
## RQStdErr   -0.592     0.539
## RLEstimate -0.311     0.494
## RLStdErr   -0.561     0.656
## CCEstimate            0.307
## CCStdErr   -0.527     0.589
##
##                variate 1 variate 2
## SS loadings        3.613     2.755
## Proportion Var     0.241     0.184
## Cumulative Var     0.241     0.424
```

Q2 continued: What are the most important variables associated with each function?

Answer:

For variate 1 and 2, the most important variables are VAStdErr (Voice and Accountability) and agereg (How long the current style of goverment has been in effect, as of end 2008). For variate 2, the most important variables are RLStdErr (Rule of Law) and CCStdErr (Control of Corruption).

Q2 continued: Are there any variables that don't help discriminate between the government types?

Answer:

GEEstimate (Government Effectiveness) and CCEstimate (Control of Corruption). They have very small loadings with values under 0.2 or around 0.3.

Question 3: Consider the observations in "newdata.csv." What would be the posterior probability of each regime type for these observations? Display in a readable form.

```
newdata = read.csv("newdata2020-update.csv")
gov.newobs = predict(gov.lda, newdata = newdata)
table = round(gov.newobs$posterior, 2)
colnames(table) = c("ParDem", "SemiPreDem", "PreDem", "CivDic", "MiliDic", "RoyDic")

#c("Parlimentary Democracy", " Semi-presidential democracy",
# " Semi-presidential democracy", " Civilian Dictatorship",
# "Military Dictatorship", "Royal Dictatorship")
table
```

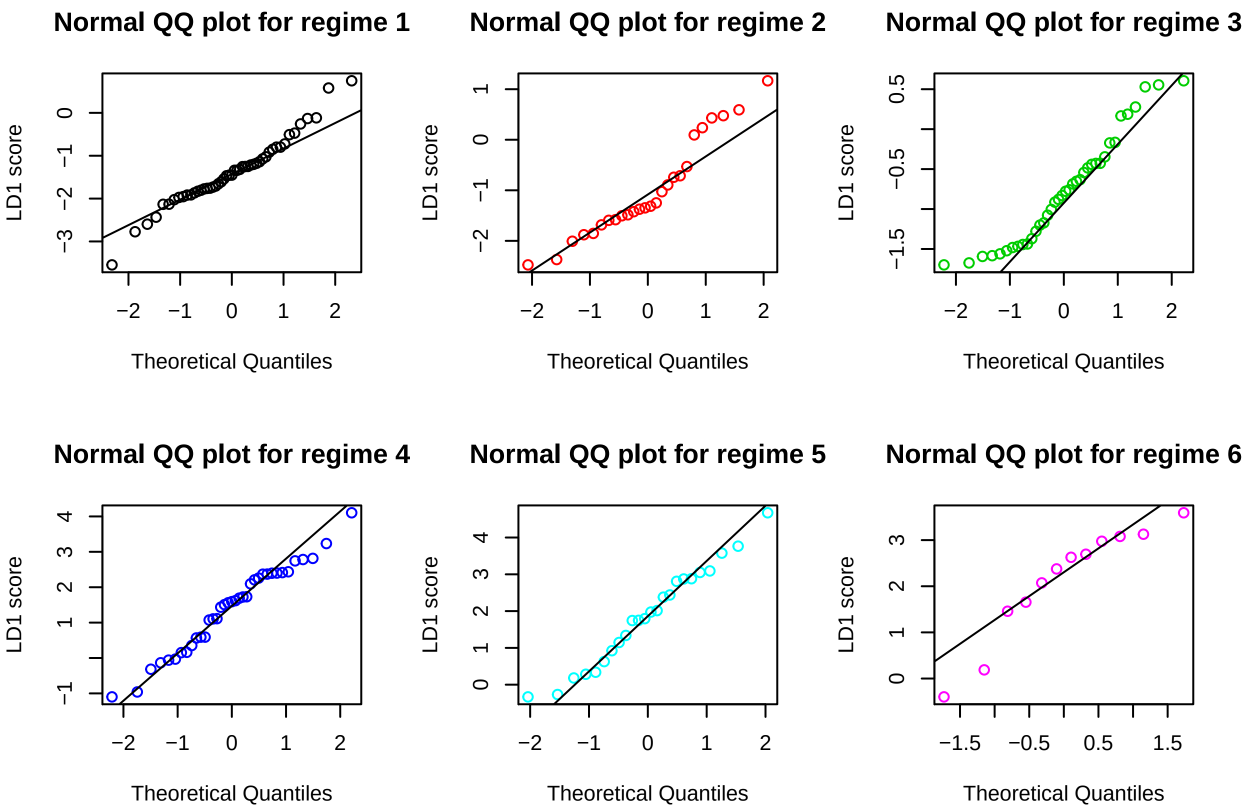
```
##   ParDem SemiPreDem PreDem CivDic MiliDic RoyDic
## 1   0.00       0.00   0.00   0.01    0.80   0.19
## 2   0.03       0.45   0.19   0.01    0.30   0.02
## 3   0.00       0.30   0.01   0.00    0.59   0.10
```

Q3 continued: These are actually data from 1996. What cautions do you have about interpreting these posterior probabilities?

The data we have was from 2008 and 2010. So when we predict for data from 1996, we might have biased prediction results since the data was not from the same population. The political situations might be different too. So we should not rely on the posterior probabilities.

Question 4: Assess the normality of the LDA scores within each regime for your "useful" components. My suggestion for this would be to use qqplots.

```
par(mfrow = c(2, 3))
for (LDA in 1:2){
  for (i in 1:6){
    qqnorm(y = gov.predict$x[, LDA][government$regime == i],
           ylab = paste0("LD", LDA, " score"), col = i,
           main = paste("Normal QQ plot for regime", i))
    qqline(gov.predict$x[government$regime == i, LDA])
  }
}
```


Normal QQ plot for regime 1

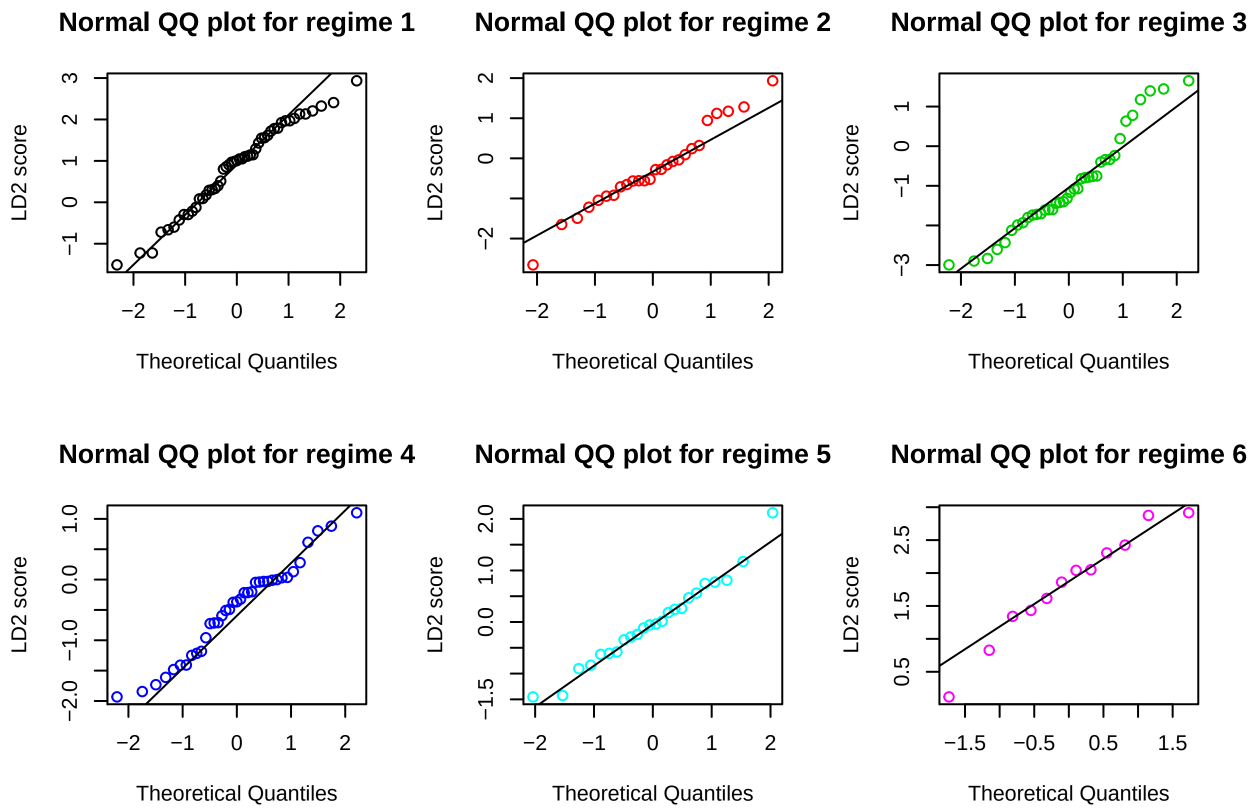
Normal QQ plot for regime 2

Normal QQ plot for regime 3

Normal QQ plot for regime 4

Normal QQ plot for regime 5

Normal QQ plot for regime 6



Comment:

Most of the qq plots showed normal LD scores. Except some small outliners for LD1 regime 2 and 6. We can say we roughly have normality for our LDA scores.

Q4 continued: How do these results influence your interpretation of your results above?

Since we have met normality, we can trust the posterior probabilities we obtained in Q3.

Question 5: Try QDA for this dataset. You will need to exclude "royal dictatorship" — explain why.

```
gov1 = government[!government$regime == 6, ]
gov.qda = qda(regime ~., data = gov1, CV = TRUE)
```

Q5 continued: Why we exclude "Royal Dictatorship" ?

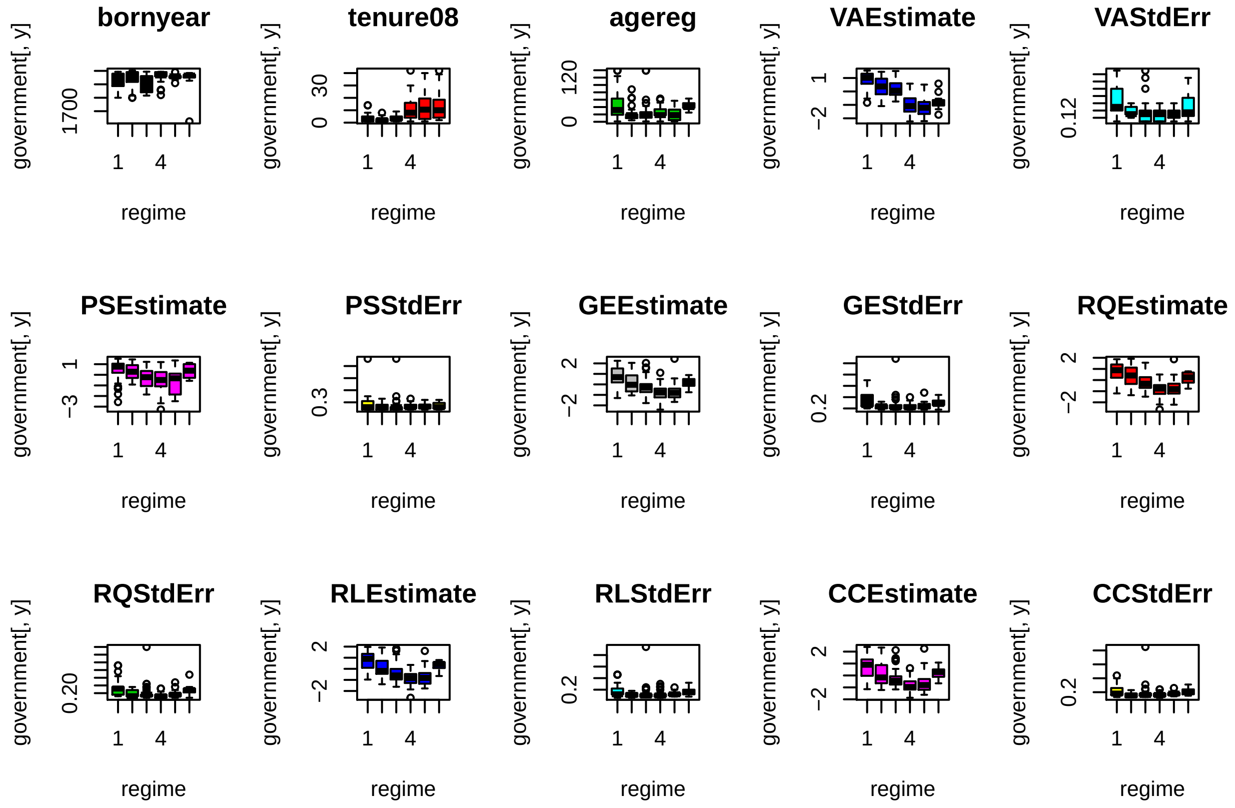
QDA estimates the covariance separately for each group to accommodate covariance differences between groups. The royal dictatorship group has only 12 observations which is not enough to compute an invertible covariance matrix for 15 variables.

Q5 continued: What situation is QDA designed for?

When the covariance matrices are the different between groups.

Q5 continued: Is there evidence that our data fall into this situation?

```
par(mfrow = c(3, 5))
for (i in 1:15){
  y = colnames(government)[-3][i]
  boxplot(government[, y] ~ government$regime, col = i,
          xlab = "regime", main = as.character(y))
}
```



##### Comment: The boxplots show us that the length of each plot clearly differs. This is an indication for non-equal variances. So QDA might be a better choice.

Q5 continued: How does the performance compare to LDA?

```
# QDA
table(paste("predicted ", gov.qda$class), gov1$regime)
```

```
##
##                  1  2  3  4  5
##   predicted  1 35  8  9  4  3
##   predicted  2  4  9  2  1  0
##   predicted  3  6  5 17  3  3
##   predicted  4  4  3  9 25 15
##   predicted  5  0  1  1  4  3
```

```r
# compute % correct
sum(gov.qda$class == gov1$regime) / length(gov1$regime)
```

```
## [1] 0.5114943
```

Recall from our leave-one-out cross validation loop in Q1. We chose to use a model with 2 variates with a prediction accuracy of 0.5107527.

QDA has a very similar result of 0.5114943. It seems QDA improves the performance just by a small margin. But QDA used a smaller sample size (exclude Royal Dictatorship), so it might be affected.