FinalExam

Question 1

a

```r
prop_var  <- c(2.90,1.24,0.85,0.72,0.22,0.07)/6
cumsum(prop_var)
```

```
## [1] 0.4833333 0.6900000 0.8316667 0.9516667 0.9883333 1.0000000
```

We would keep the first 3 principal components accounting for 83.2% of the variance of the scaled data

b

The fourth eigenvalue still represents a meaningful amount of information taking explained variance from .83 to .95. It is significantly above the standard cutoff which for 6 eigenvalues would be 1/6 = .17. This woudl provide a much fuller picture and is likely worth the added complexity.

c

The first principal component appears to represent overall team strength. where positive pc1 means a more complete team and negative pc1 is a more struggling team. We can look at loading and see that DefPts and DefRush are both negative and the Offenseive poitns offenseive rush yards and offensive pass yards are positive. The only exception is that the Defensive pass coefficient is flipped from ush and points which indicates some nuance on defense.

d

Principal component 2 contrasts run heavy offesnses against pass dominant teams with strong pass defense. this is seen with the positive loading on offensive rushing and strong negative loads on defensive points, pass defense, and offensive passing. here a positive score indicates a lot of rushing output but weaker passing offense and defense while a strong neagtive score indicates a pass focused offense with a strong pass defense.

e

27 has a slightly above average pc1 indicating that both their offense and defense were above average, and a very strong and high pc2 indicating that they were a rush first team

f

Point 16 representing the chiefs hovers in the middle of the plot for both PC1 and PC2. They are just above average in PC1 indicating that the oferal team is slightly above average, and they also hover right around 0 for PC2 indicating a balanced team between rushing and passing.

g

```r
load("Data/nfl2024.RData")

vars <- c("DefPts","DevRush","DevPass","OffPts","OffRush","OffPass")

X   <- scale(nfl[ , vars])

pca <- prcomp(X, center = FALSE, scale. = FALSE)
scores <- pca$x

team <- "Arizona Cardinals"
row  <- which(nfl$Team == team)
z    <- scores[row, 1:4]

x_std_hat <- as.numeric(z %*% t(pca$rotation[ , 1:4]))

means <- colMeans(nfl[ , vars])
sds   <- apply(nfl[ , vars], 2, sd)

x_hat <- means + x_std_hat * sds
names(x_hat) <- vars

print("Reconstructed Values")
```

```
## [1] "Reconstructed Values"
```

```r
round(x_hat, 2)
```

```
##  DefPts DevRush DevPass  OffPts OffRush OffPass
##   22.97  122.19  215.58   24.47  140.58  209.65
```

```r
actual <- nfl[nfl$Team == team, vars]

#diff <- actual - x_hat
#diff
```

h

```r
library(MASS)

lda_fit <- lda(Playoff ~ ., data = nfl[ , c("Playoff", vars)],
               prior = c(0.5625, 0.4375))

round(lda_fit$scaling, 3)
```

```
##             LD1
## DefPts  -0.186
## DevRush -0.027
## DevPass  0.001
## OffPts   0.423
## OffRush -0.021
## OffPass -0.042
```

i

```
scores <- predict(lda_fit)$x
playoff_teams   <- nfl$Team[nfl$Playoff == 1]
playoff_scores  <- scores[nfl$Playoff == 1, 1]

team_low <- playoff_teams[ which.min(playoff_scores) ]
team_low
```

```
## [1] "Los Angeles Rams"
```

```
playoff_scores[ which.min(playoff_scores) ]
```

```
##        19
## -0.861632
```

j

```
prop.table(table(nfl$Playoff))
```

```
##
##      0      1
## 0.5625 0.4375
```

k

```
xnew <- data.frame(DefPts = 21.89,
                   DevRush = 124.85,
                   DevPass = 208.82,
                   OffPts = 23.69,
                   OffRush = 132.15,
                   OffPass = 189.54)

predict(lda_fit, newdata = xnew)
```

```
## $class
## [1] 1
## Levels: 0 1
##
## $posterior
##            0         1
## 1 0.03855635 0.9614437
##
## $x
##        LD1
## 1 1.297546
```

l

Im a little confused here so i have recreated the heirarchical clustering appraoch and am using it to produce two groups.

```
hc_cent <- hclust(dist(scale(nfl[ , vars])), method = "centroid")
groups2 <- cutree(hc_cent, k = 2)

table(Cluster = groups2, Playoff = nfl$Playoff)
```

```
##        Playoff
## Cluster  0  1
##       1 17 14
##       2  1  0
```

m

We must standardize because our metrics are on vary different scales. Points range up to maybe 40 and yards range in the hundreds. without standardizing, the euclidean distance used for clustering would be dominated by the variable with larger variance and would not really represent waht we want it to. By putting our information on the common scale of standard deviations, euclidean distance is now more appropriate and considers all variables equally.

n

```
hc_ward <- hclust(dist(scale(nfl[ , vars])), method = "ward.D2")
groups2w<- cutree(hc_ward, 2)
table(Cluster = groups2w, Playoff = nfl$Playoff)
```

```
##        Playoff
## Cluster  0  1
##       1 18  3
##       2  0 11
```

```
mean(groups2w == nfl$Playoff)
```

```
## [1] 0.09375
```

Hierarchical clustering at level 2 performs quite well only getting three teams incorrect.

o

It seems group 2 is a strong playoff team as they are teh most positive on pc1. becasue PC2 talks more about play style, it is subjective and we focus on PC1 which leads us to cluster 2.

p

```
library(Hotelling)
```

```
## Loading required package: corpcor
```

```
hot <- hotelling.test(
        x = nfl[nfl$Playoff == 1 , vars],
        y = nfl[nfl$Playoff == 0 , vars])

hot
```

```
## Test stat:  78.168 
## Numerator df:  6 
## Denominator df:  25 
## P-value:  5.901e-06
```

q

```
pvals <- sapply(vars, \(v)
           t.test(nfl[[v]] ~ nfl$Playoff)$p.value)

alpha <- 0.05 / length(vars)

data.frame(Variable = vars,
           p.value  = round(pvals, 5),
           Significant = pvals <= alpha)
```

```
##         Variable p.value Significant
## DefPts    DefPts 0.00000        TRUE
## DevRush  DevRush 0.00011        TRUE
## DevPass  DevPass 0.58277       FALSE
## OffPts    OffPts 0.00004        TRUE
## OffRush  OffRush 0.00674        TRUE
## OffPass  OffPass 0.26419       FALSE
```

r

LDA and the means test would have been invalidated if we have inequal variance problems.

Question 2

a

PCA and Factor analysis may help identify outliers via dimension reduction making it more easily to visually identify them on a plot

Clustering especially heirarchical will give us an indication of how “difficult” it is for some of these things to cluster together which can also help us identify outliers

Hotelling test is designed for comparing groups so is not useful for identifying individual outliers.

b

PCA will be great for exploration here, but is likely not used directly for modeling.

Clustering is the primary useful approach for the traditional customer segmentation.

If labels are provided which is uncommon for segmentation but common for other similar use cases for this type of analysis, LDA could be used. this would be relevant for target variables like customer churn.

Question 3

```r
mu      <- c(1,2,3)

Sigma   <- matrix(c(3,1,-1, 1,2,0, -1,0,3), 3,3)

A       <- matrix(c(2,-1,0, 0,3,1), nrow = 2, byrow = TRUE)

mu_Y    <- A %*% mu
Sigma_Y<- A %*% Sigma %*% t(A)

mu_Y
```

```
##      [,1]
## [1,]    0
## [2,]    9
```

```r
Sigma_Y
```

```
##      [,1] [,2]
## [1,]   10   -2
## [2,]   -2   21
```

$$\mathbf{Y} = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} \sim \mathcal{N}_2\left(\begin{pmatrix} 0 \\ 9 \end{pmatrix}, \begin{pmatrix} 10 & -2 \\ -2 & 21 \end{pmatrix} \right).$$