Dimensionality Reduction

Justin Hardy, Fernando Colman, Linus Fackler, Isabelle Villegas

The Data Set

Starting by reading in the data set. The data set we'll use for the assignment consists of data collected by an airline organization, over their customers' submitted satisfaction surveys, as well as relevant information about their flight and demographic.

If you want to see the data set for yourself, you access it here.

```
data <- read.csv("airline_data.csv")
```

Cleaning Up The Data Set

Cleaning up data set for logistic regression, by converting qualitative columns into factors.

```
# Factor columns
data$satisfaction <- factor(data$satisfaction) # satisfaction
data$Gender <- factor(data$Gender) # gender
data$Customer.Type <- factor(data$Customer.Type) # customer
data$Type.of.Travel <- factor(data$Type.of.Travel) # travel
data$Class <- factor(data$Class) # class

# Normalize factor names
levels(data$satisfaction) <- c("Dissatisfied", "Satisfied")
levels(data$Customer.Type) <- c("Disloyal", "Loyal")
levels(data$Type.of.Travel) <- c("Business", "Personal")

# Create new cleaned CustomerData data frame for full factoring (linear regression)
CustomerData_factored <- data

# Continue factoring numeric finite columns
for(i in 8:21) {
CustomerData_factored[,i] <- factor(CustomerData_factored[,i], levels=c(0,1,2,3,4,5)) # out-of-5 rating
}

# Remove na rows
data_complete <- data[complete.cases(data),]
data <- CustomerData_factored[complete.cases(CustomerData_factored),]
```

Dividing Into Train/Test

Dividing the data set into train/test

We are also using the preProcess function to find the principal components from the data

```
# 80/20 split
split <- round(nrow(data)*0.8)
training <- data[1:split, ]
test <- data[(split+1):nrow(data),]
```

```
summary(training)
```

```
##       satisfaction      Gender       Customer.Type         Age
##  Dissatisfied:57750   Female:52764   Disloyal:23515   Min.   : 7.00
##  Satisfied   :45840   Male  :50826   Loyal   :80075   1st Qu.:25.00
##                                                       Median :38.00
##                                                       Mean   :38.36
##                                                       3rd Qu.:50.00
##                                                       Max.   :85.00
##   Type.of.Travel       Class       Flight.Distance Seat.comfort
##  Business:63752   Business:40651   Min.   :  50    0: 4771
##  Personal:39838   Eco     :54657   1st Qu.:1394    1:16727
##                   Eco Plus: 8282   Median :1910    2:24362
##                                    Mean   :1959    3:24722
##                                    3rd Qu.:2473    4:22426
##                                    Max.   :6951    5:10582
##  Departure.Arrival.time.convenient Food.and.drink Gate.location
##  0: 6632                           0: 5904        0:    2
##  1:15544                           1:16337        1:17393
##  2:17546                           2:21795        2:19206
##  3:17890                           3:22724        3:28207
##  4:24331                           4:21696        4:24740
##  5:21647                           5:15134        5:14042
##  Inflight.wifi.service Inflight.entertainment Online.support
##  0:  130               0: 2953                0:    1
##  1:13705               1:10897                1:12980
##  2:22562               2:18204                2:16280
##  3:23151               3:22655                3:20325
##  4:25002               4:31868                4:31527
##  5:19040               5:17013                5:22477
##  Ease.of.Online.booking On.board.service Leg.room.service Baggage.handling
##  0:   18                0:    5          0:  440          0:    0
##  1:13302                1:12202          1:10091          1: 7011
##  2:19770                2:15907          2:20132          2:12481
##  3:21923                3:24925          3:20702          3:23024
##  4:32465                4:33563          4:32780          4:40610
##  5:16112                5:16988          5:19445          5:20464
##  Checkin.service Cleanliness Online.boarding Departure.Delay.in.Minutes
##  0:    1         0:    5     0:   14         Min.   :   0.00
##  1:13394         1: 6845     1:14332         1st Qu.:   0.00
##  2:13466         2:12421     2:17536         Median :   0.00
##  3:28325         3:22497     3:25207         Mean   :  15.05
##  4:28939         4:41143     4:27446         3rd Qu.:  13.00
##  5:19465         5:20679     5:19055         Max.   :1592.00
##  Arrival.Delay.in.Minutes
##  Min.   :   0.00
```

```
##  1st Qu.:   0.00
##  Median :   0.00
##  Mean   :  15.55
##  3rd Qu.:  14.00
##  Max.   :1584.00
```

Principal Component Analysis

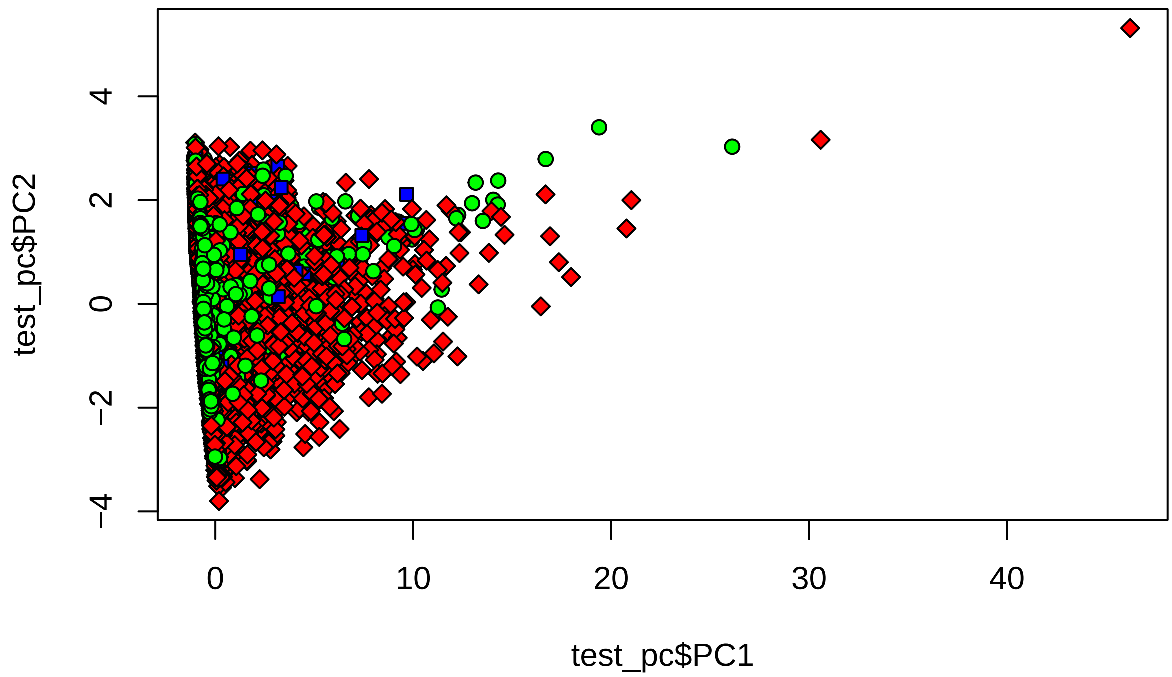
Running PCA on flight data

```
# preProcessing the training data in order to find the principal components
pca_out <- preProcess(training[,1:ncol(training)], method=c("center", "scale", "pca"))
pca_out
```

```
## Created from 103590 samples and 23 variables
##
## Pre-processing:
##   - centered (4)
##   - ignored (19)
##   - principal component signal extraction (4)
##   - scaled (4)
##
## PCA needed 3 components to capture 95 percent of the variance
```

Plotting PC1 and PC2 for PCA Model

```
train_pc <- predict(pca_out, training[, 1:ncol(training)])
test_pc <- predict(pca_out, test[,])
plot(test_pc$PC1, test_pc$PC2, pch=c(23,21,22)[unclass(test_pc$Class)], bg=c("red","green","blue")[uncla
```




```
train_df <- data.frame(train_pc$PC1, train_pc$PC2, training$Class)
test_df <- data.frame(test_pc$PC1, test_pc$PC2, test$Class)

pred <- knn(train=train_df[,1:2], test=test_df[,1:2], cl=train_df[,3], k=3)
mean(pred==test$Class)
```

```
## [1] 0.4663088
```

The accuracy is pretty low for PCA, although I can imagine it may be because of the overlapping data, which would cause the accuracy to be a lot lower.

Linear Discriminant Analysis

```
lda1 <- lda(Class~satisfaction+Gender+Customer.Type+Type.of.Travel, data=training)
coef(lda1)
```

```
##                               LD1        LD2
## satisfactionSatisfied   0.7822817 -1.0676272
## GenderMale              0.2408862 -0.6331460
```

```
## Customer.TypeLoyal      1.1851552  2.5161674
## Type.of.TravelPersonal -2.7104777 -0.3787878
```

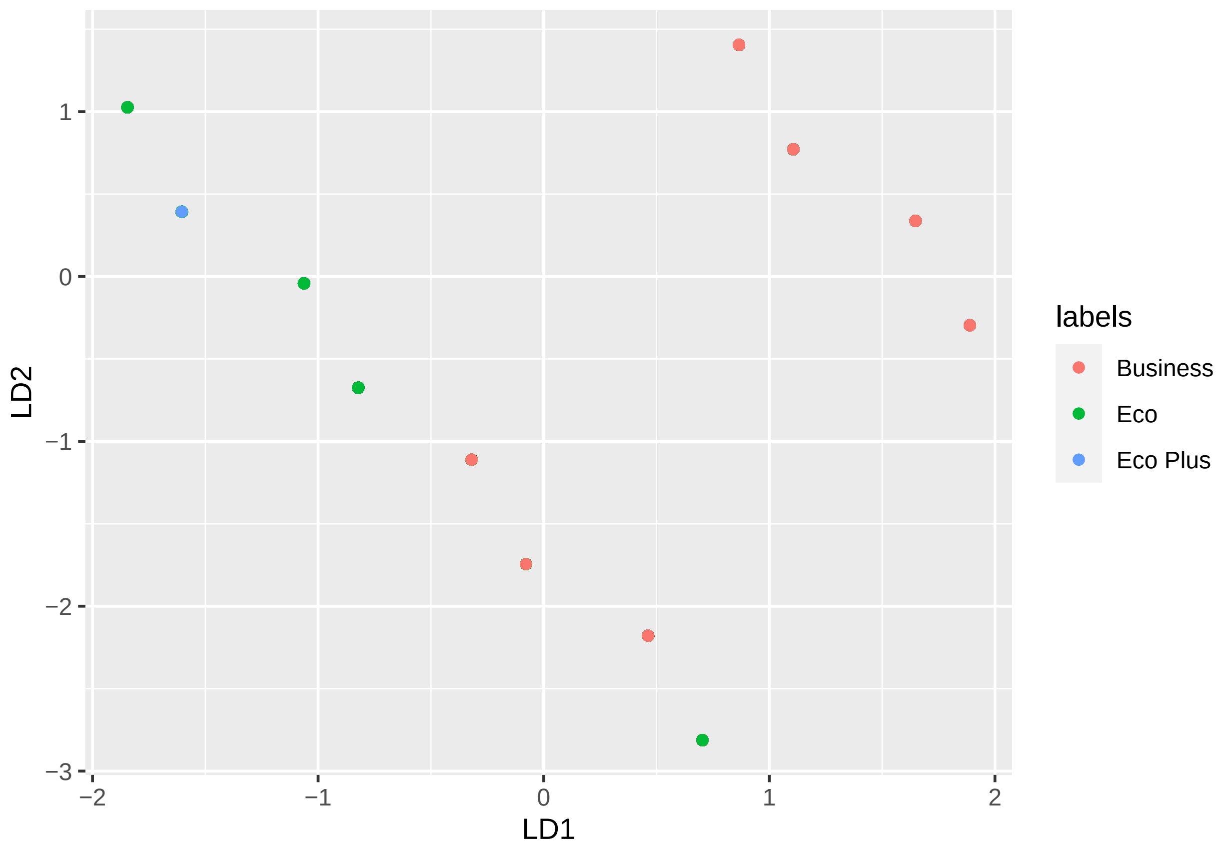
```r
lda1$means
```

```
##          satisfactionSatisfied GenderMale Customer.TypeLoyal
## Business             0.5667757  0.4957320          0.7737571
## Eco                  0.3625519  0.4906782          0.7546517
## Eco Plus             0.3602994  0.4654673          0.8903646
##          Type.of.TravelPersonal
## Business             0.06523825
## Eco                  0.59827652
## Eco Plus             0.54165660
```

```r
## Plotting LD1 and LD2

ggplotLDAPrep <- function(x){
  if (!is.null(Terms <- x$terms)) {
    data <- model.frame(x)
    X <- model.matrix(delete.response(Terms), data)
    g <- model.response(data)
    xint <- match("(Intercept)", colnames(X), nomatch = 0L)
    if (xint > 0L)
      X <- X[, -xint, drop = FALSE]
  }
  means <- colMeans(x$means)
  X <- scale(X, center = means, scale = FALSE) %*% x$scaling
  rtrn <- as.data.frame(cbind(X,labels=as.character(g)))
  rtrn <- data.frame(X,labels=as.character(g))
  return(rtrn)
}

# Plotting LD1 and LS2
fitGraph <- ggplotLDAPrep(lda1)
ggplot(fitGraph, aes(LD1,LD2, color=labels))+geom_point()
```




```
lda1$means
```

```
##          satisfactionSatisfied GenderMale Customer.TypeLoyal
## Business             0.5667757  0.4957320          0.7737571
## Eco                  0.3625519  0.4906782          0.7546517
## Eco Plus             0.3602994  0.4654673          0.8903646
##          Type.of.TravelPersonal
## Business             0.06523825
## Eco                  0.59827652
## Eco Plus             0.54165660
```

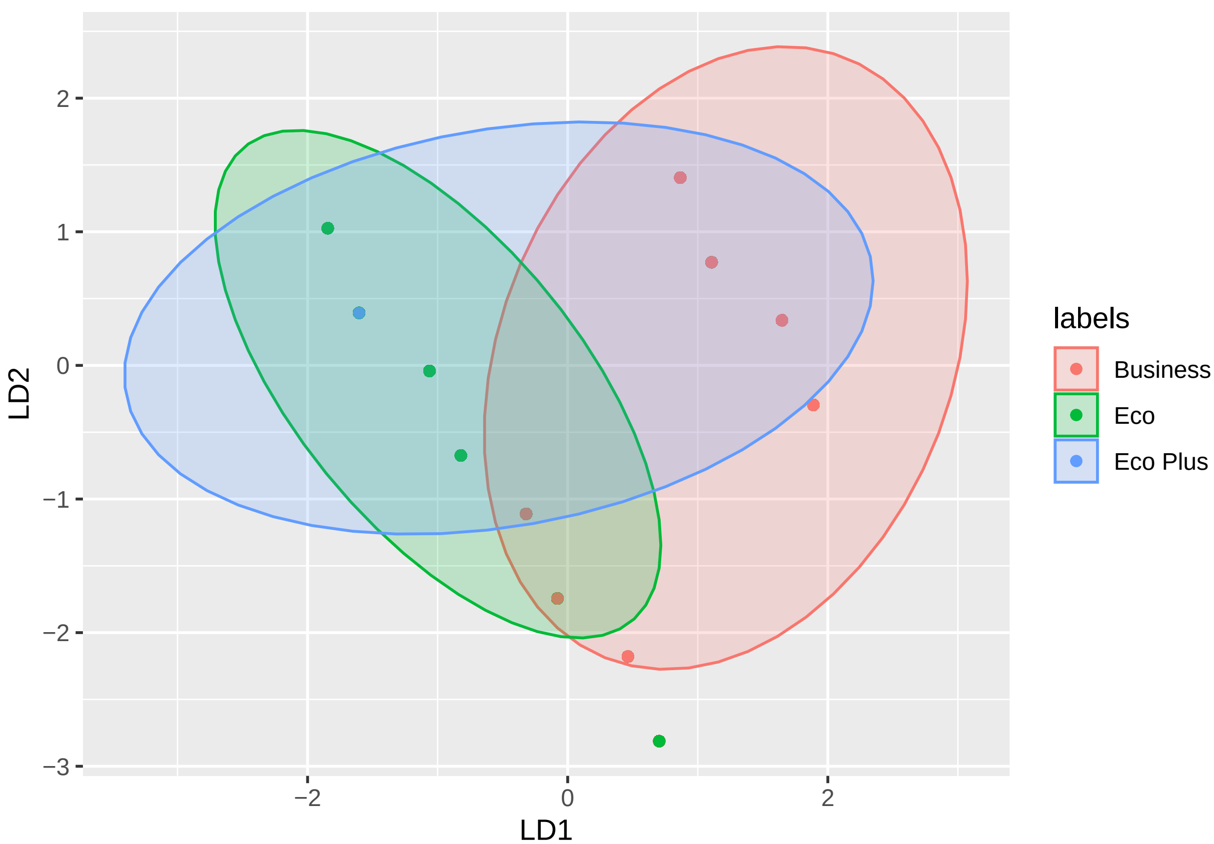
```
lda1
```

```
## Call:
## lda(Class ~ satisfaction + Gender + Customer.Type + Type.of.Travel,
##     data = training)
##
## Prior probabilities of groups:
##  Business       Eco  Eco Plus
## 0.3924220 0.5276281 0.0799498
##
## Group means:
##          satisfactionSatisfied GenderMale Customer.TypeLoyal
## Business             0.5667757  0.4957320          0.7737571
## Eco                  0.3625519  0.4906782          0.7546517
```

```
## Eco Plus              0.3602994  0.4654673          0.8903646
##          Type.of.TravelPersonal
## Business              0.06523825
## Eco                   0.59827652
## Eco Plus              0.54165660
##
## Coefficients of linear discriminants:
##                              LD1        LD2
## satisfactionSatisfied   0.7822817 -1.0676272
## GenderMale              0.2408862 -0.6331460
## Customer.TypeLoyal      1.1851552  2.5161674
## Type.of.TravelPersonal -2.7104777 -0.3787878
##
## Proportion of trace:
##    LD1    LD2
## 0.9835 0.0165
```

```
ggplot(fitGraph, aes(LD1,LD2, color=labels))+geom_point() +
    stat_ellipse(aes(x=LD1, y=LD2, fill = labels), alpha = 0.2, geom = "polygon")
```

```
## Warning in MASS::cov.trob(data[, vars]): Probable convergence failure
```




```
glm <- glm(Class~satisfaction+Gender+Customer.Type+Type.of.Travel, data=training, family=binomial)
```

```r
# summary
summary(glm)
```

```
##
## Call:
## glm(formula = Class ~ satisfaction + Gender + Customer.Type +
##     Type.of.Travel, family = binomial, data = training)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6182  -0.6469   0.2798   0.4928   1.9026
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)             0.82143    0.01666   49.30   <2e-16 ***
## satisfactionSatisfied  -1.17384    0.01734  -67.69   <2e-16 ***
## GenderMale             -0.17359    0.01640  -10.59   <2e-16 ***
## Customer.TypeLoyal     -1.10531    0.01814  -60.95   <2e-16 ***
## Type.of.TravelPersonal  3.67842    0.02380  154.53   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 138773  on 103589  degrees of freedom
## Residual deviance:  94149  on 103585  degrees of freedom
## AIC: 94159
##
## Number of Fisher Scoring iterations: 5
```

```r
lda_pred <- predict(lda1, newdata=test, type="class")
#lda_pred$class
```

Calculating accuracy for LDA

```r
mean(lda_pred$class==test$Class)
```

```
## [1] 0.8306754
```

The accuracy for LDA is a lot better, though that is probably because the graph for the LDA was a lot better with more defined classes.