LDA_flu_test

```
library(MASS)
library(ggplot2)
library(scales)
```

flu virus all test set 1

```
# label and seq data
uniflna_test1 <-
  read.csv("/home/brian/Documents/data_mining/flu_project/uniflna_test1.csv", na.strings="")

# comparisons from sourmash
uniflna_test1_cmp <-
  read.csv("/home/brian/Documents/data_mining/flu_project/uniflna_test1_cmp.csv")

# Label the rows
rownames(uniflna_test1_cmp) <- colnames(uniflna_test1_cmp)

# add gene column
uniflna_test1_wgs <- uniflna_test1_cmp
uniflna_test1_wgs$gene <- uniflna_test1$gene
uniflna_test1_wgs$segment <- factor(uniflna_test1$segment)

# Transform for plotting
uniflna_test1_cmp_mat <- as.matrix(uniflna_test1_cmp)
```

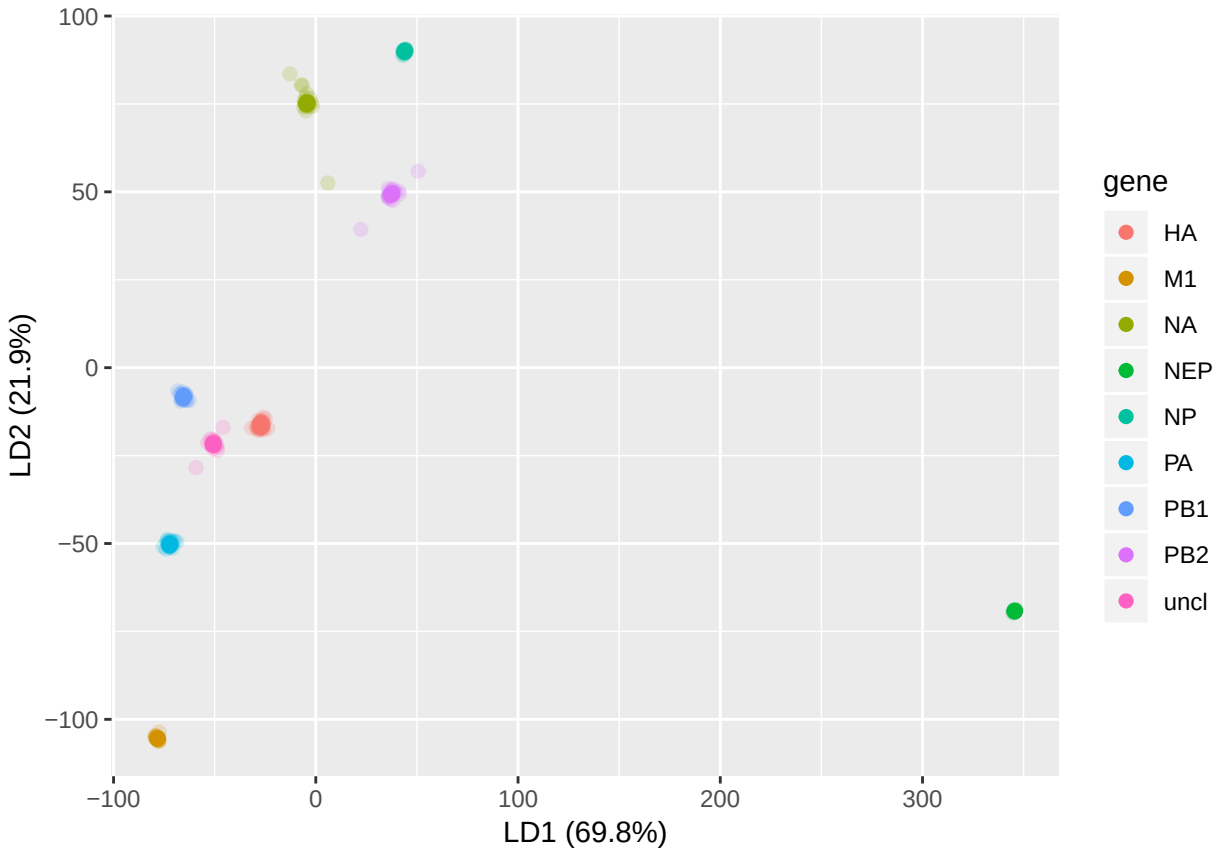
```
lda <- lda(gene ~ .,
           uniflna_test1_wgs[,-1002])
```

```
## Warning in lda.default(x, grouping, ...): variables are collinear
```

```
prop.lda = lda$svd^2/sum(lda$svd^2)

plda <- predict(object = lda,
                newdata = uniflna_test1_wgs)
```

```
dataset = data.frame(gene = uniflna_test1_wgs[,1001],
                     lda = plda$x)
```

```
p1 <- ggplot(dataset) + geom_point(aes(lda.LD1, lda.LD2, colour = gene), size=2, alpha=0.2) +
  #theme_minimal() +
  labs(x = paste("LD1 (", percent(prop.lda[1]), ")", sep=""),
       y = paste("LD2 (", percent(prop.lda[2]), ")", sep="")) +
  guides(colour = guide_legend(override.aes = list(alpha = 1)))
# https://stackoverflow.com/questions/5290003/how-to-set-legend-alpha-with-ggplot2

# ggsave("gene_LDA_plot.png", plot = p1, width = 7, height = 4, dpi = 200)

p1
```



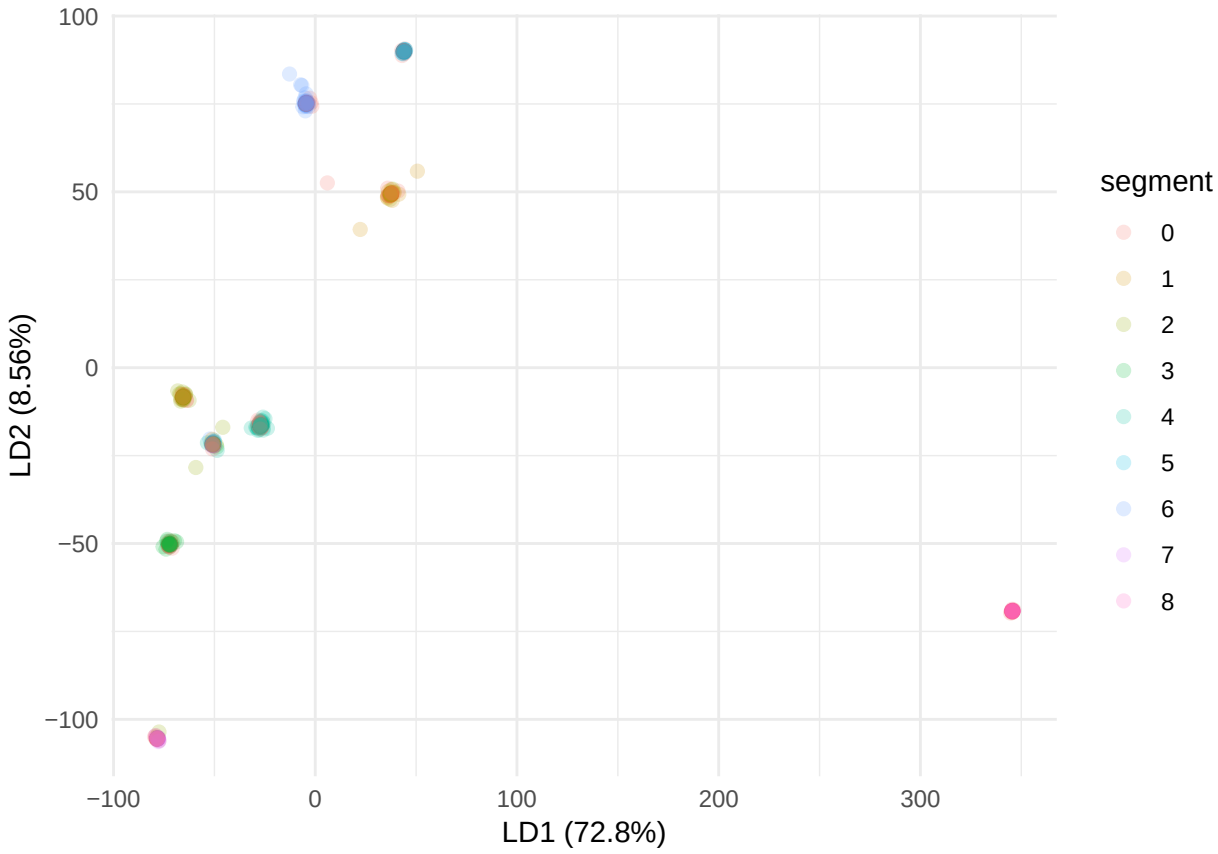

```
lda2 <- lda(segment ~ .,
            uniflna_test1_wgs[,-1001])
```

```
## Warning in lda.default(x, grouping, ...): variables are collinear
```

```
prop.lda2 = lda2$svd^2/sum(lda2$svd^2)

plda2 <- predict(object = lda2,
                 newdata = uniflna_test1_wgs)
```

```
dataset2 = data.frame(segment = uniflna_test1_wgs[,1002],
                      lda2 = plda$x)
```

```
p2 <- ggplot(dataset2) + geom_point(aes(lda2.LD1, lda2.LD2, colour = segment), size=2, alpha=0.2) +
  theme_minimal() +
  labs(x = paste("LD1 (", percent(prop.lda2[1]), ")", sep=""),
       y = paste("LD2 (", percent(prop.lda2[2]), ")", sep=""))
p2
```


100
50
0
−50
−100
LD2 (8.56%)
−100
0
100
200
300
LD1 (72.8%)
segment
0
1
2
3
4
5
6
7
8