LDA_flu_test

LDA

https://tgmstat.wordpress.com/2014/01/15/computing-and-visualizing-lda-in-r/ https://gist.github.com/thigm85/8424654

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

```
cpal <- c('#e41a1c','#377eb8','#4daf4a','#984ea3','#ff7f00','#e6e600',
          '#a65628','#f781bf','#4d4d4d')
```

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
```

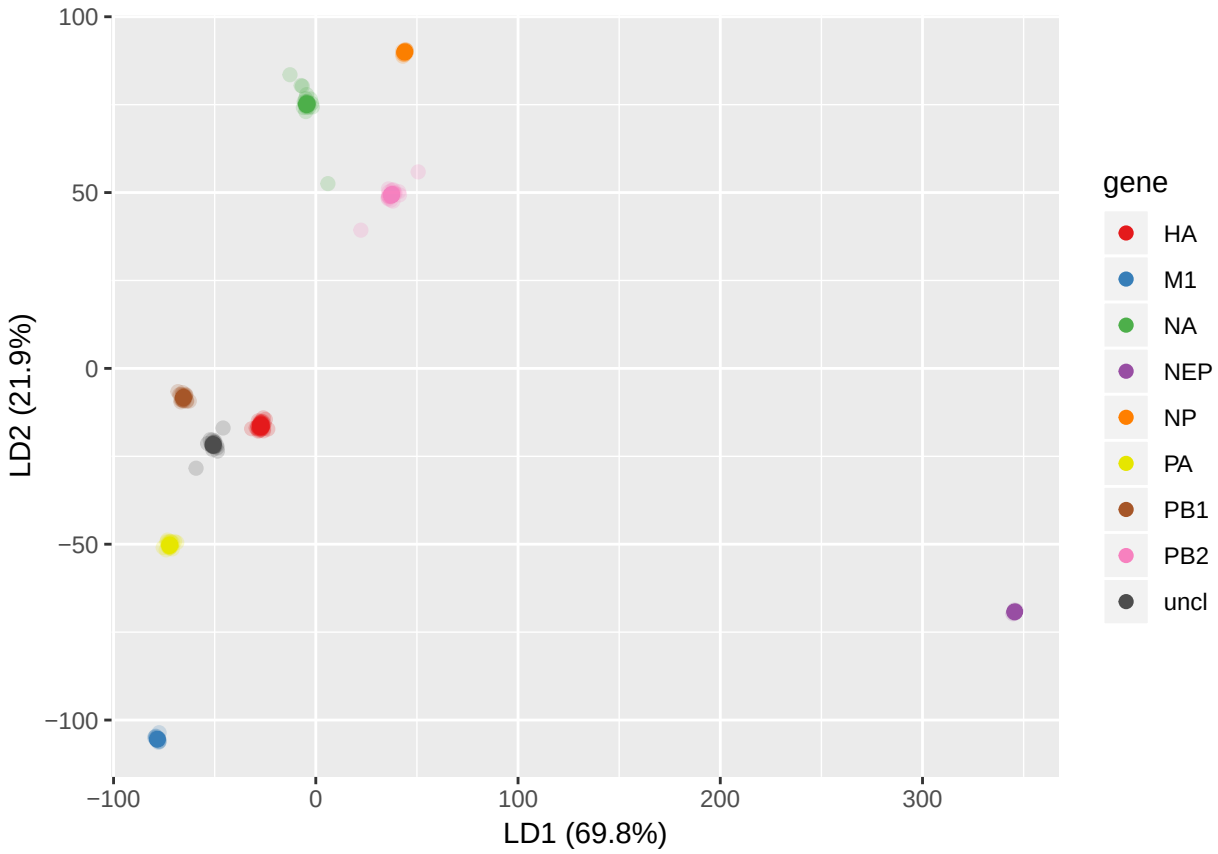
```
prop.lda = lda$svd^2/sum(lda$svd^2)

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="")) +
  scale_color_manual(values=cpal) +
  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])
```

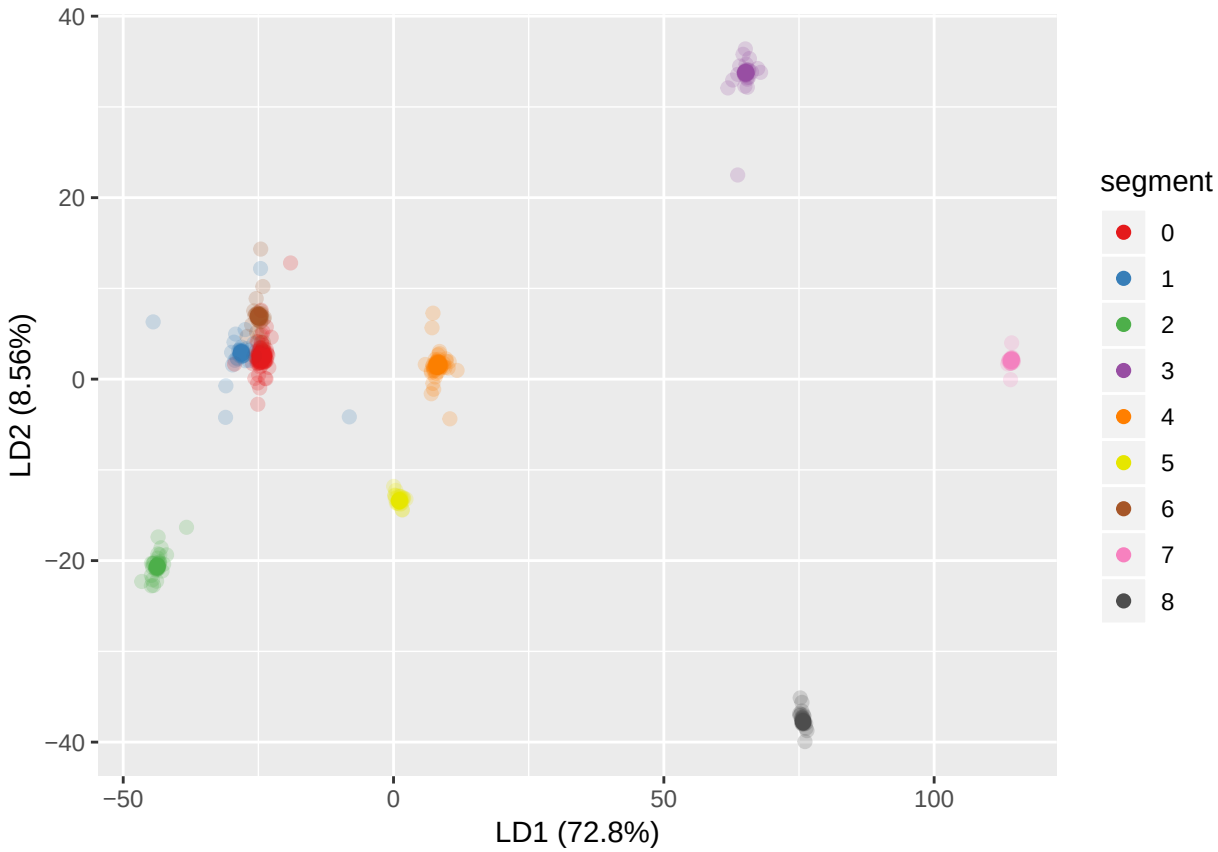
```
## Warning in lda.default(x, grouping, ...): variables are collinear
```

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

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

segment labels

```
dataset2 = data.frame(segment = uniflna_test1_wgs[,1002],
                      lda2 = plda2$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="")) +
  scale_color_manual(values=cpal) +
  guides(colour = guide_legend(override.aes = list(alpha = 1)))
p2
```



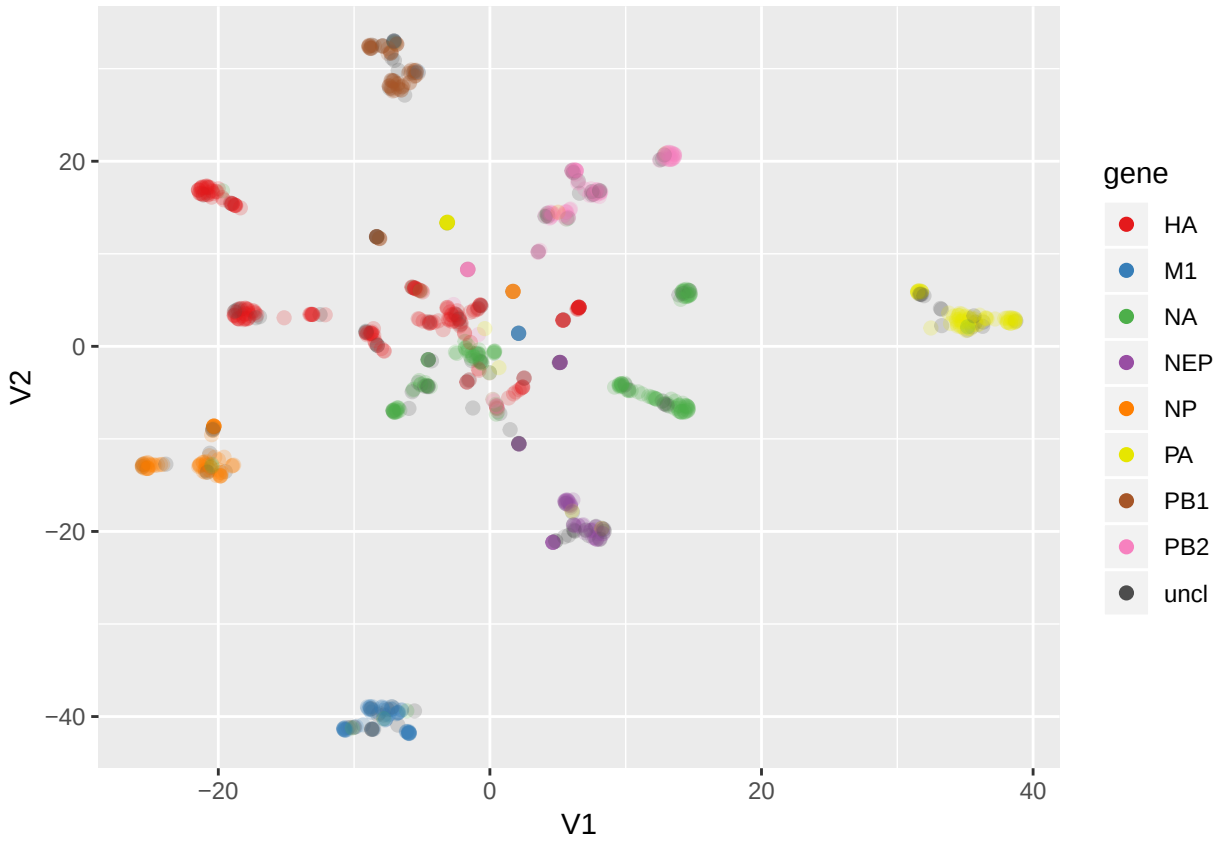
segment labels not as reliable.

Tsne for fun

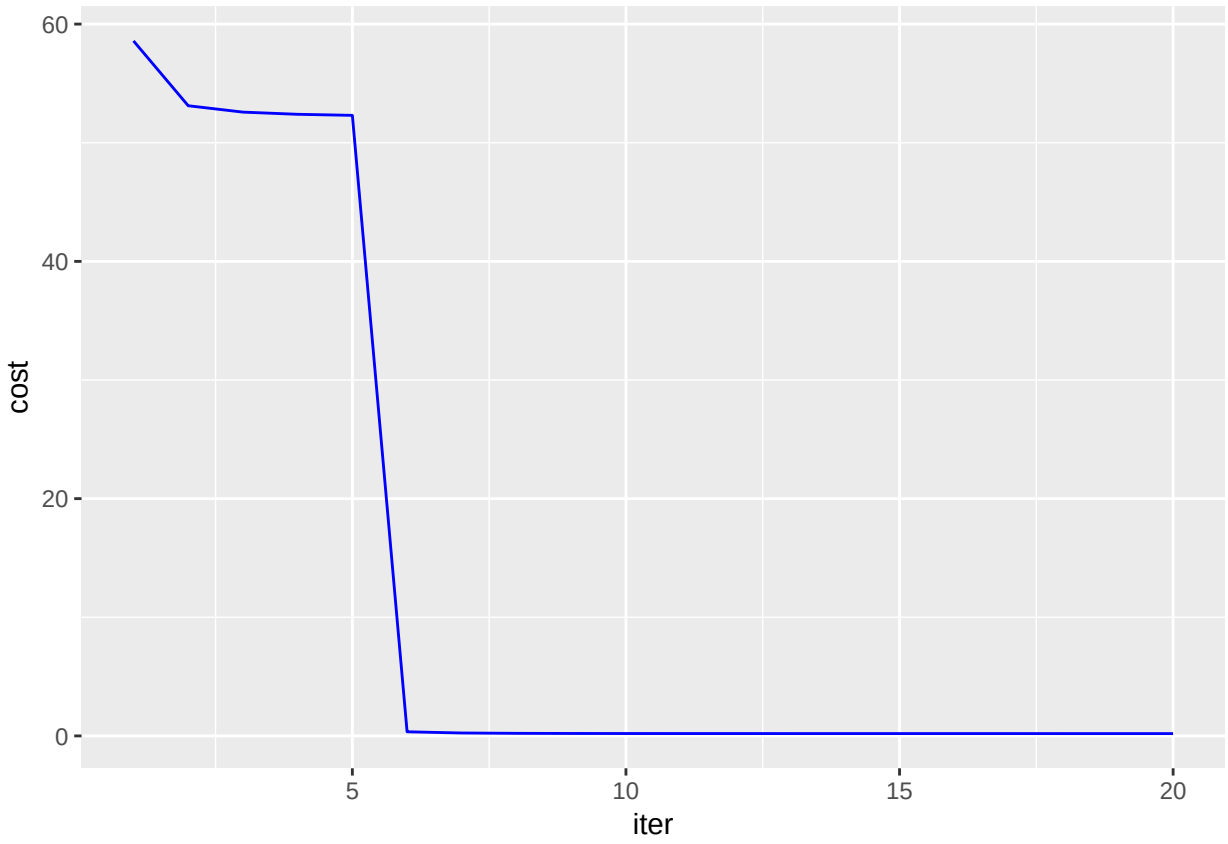
```
library(Rtsne)
```

```
tsne_model <- Rtsne(uniflna_test1_cmp_mat, check_duplicates=FALSE, pca=TRUE, perplexity=50, theta=0.25,
d_tsne = as.data.frame(tsne_model$Y)
d_tsne$gene <- uniflna_test1_wgs[,1001]
#plot(d_tsne$V1, d_tsne$V2)
```

```
ggplot(d_tsne, aes(V1, V2, colour = gene)) + geom_point(size=2, alpha=0.2) +
  scale_color_manual(values=cpal) +
  guides(colour = guide_legend(override.aes = list(alpha = 1)))
```




```
itercosts <- data.frame(iter = 1:20, cost = tsne_model$itercosts)

ggplot(itercosts, aes(iter,cost)) + geom_line(color='blue')
```



second sample with 2500 sequences, k=7 uniflna_test2_cmp.csv

```
# label and seq data
uniflna_test2 <-
  read.csv("/home/brian/Documents/data_mining/flu_project/uniflna_test2.csv", na.strings="")

# comparisons from sourmash
uniflna_test2_cmp <-
  read.csv("/home/brian/Documents/data_mining/flu_project/uniflna_test2_cmp.csv")

# Label the rows
rownames(uniflna_test2_cmp) <- colnames(uniflna_test2_cmp)

# add gene column
uniflna_test2_wgs <- uniflna_test2_cmp
uniflna_test2_wgs$gene <- uniflna_test2$gene
#uniflna_test2_wgs$segment <- factor(uniflna_test1$segment)

# Transform for plotting
uniflna_test2_cmp_mat <- as.matrix(uniflna_test2_cmp)
```

```
lda3 <- lda(gene ~ .,
            uniflna_test2_wgs)
```

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

```
prop.lda3 = lda3$svd^2/sum(lda3$svd^2)

plda3 <- predict(object = lda3,
```

```
                newdata = uniflna_test2_wgs)
```

```
dataset3 = data.frame(gene = uniflna_test2_wgs[,2501],
                      lda3 = plda3$x)
```

```
p3 <- ggplot(dataset3) + geom_point(aes(lda3.LD1, lda3.LD2, colour = gene), size=2, alpha=0.2) +
  #theme_minimal() +3
  labs(x = paste("LD1 (", percent(prop.lda3[1]), ")", sep=""),
       y = paste("LD2 (", percent(prop.lda3[2]), ")", sep="")) +
  scale_color_manual(values=cpal) +
  guides(colour = guide_legend(override.aes = list(alpha = 1)))
# https://stackoverflow.com/questions/5290003/how-to-set-legend-alpha-with-ggplot2

# ggsave("gene2500_LDA_plot.png", plot = p3, width = 7, height = 4, dpi = 200)

p3
```