Challenge 2

Naiyu Jiang

Question 1: Load and scale data.

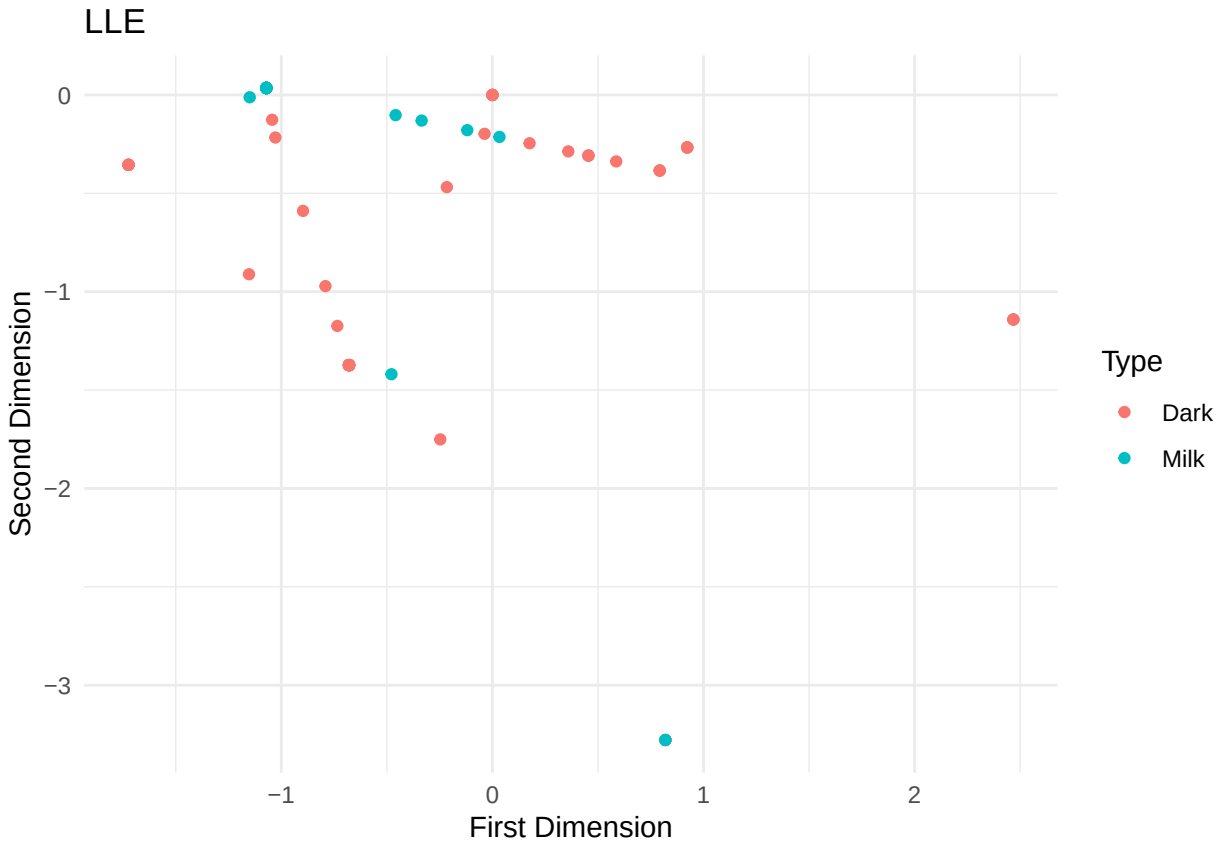
```
# load libraries
library(tidyverse)
library(lle)
library(parallel)
library(ggrepel)
library(tictoc)
library(patchwork)
# load dataset
chocolates <- read_csv("~/Desktop/Challenge 2/data/chocolates.csv")
# subset and scale data
cho_scaled <- chocolates[,4:14] %>%
  as_tibble() %>%
  mutate_at(scale, .vars = vars(-Type)) # scale numeric variables (except Type)
cho_scaled$Type <- as.factor(cho_scaled$Type) # set Type as factor
```

Question 2: Setting k = 2 and m = 2. Plot the results with color varying by chocolate Type.

```
{
  tic()
  lle_fit1 <- lle(cho_scaled[,2:11],
                  m = 2,
                  nnk = TRUE,
                  k = 2)
  toc()
}
```

```
## finding neighbours
## calculating weights
## computing coordinates
## 0.125 sec elapsed
```

```
cho_scaled %>%
  tibble() %>%
  ggplot(aes(x = lle_fit1$Y[,1],
             y = lle_fit1$Y[,2],
             col = Type)) +
  geom_point() +
  labs(x = "First Dimension",
       y = "Second Dimension",
       title = "LLE") +
  theme_minimal()
```



Given k = 2, there is no clear separation between types of chocolate from such a local version of LLE. The `Dark` points and `Milk` points mix together.

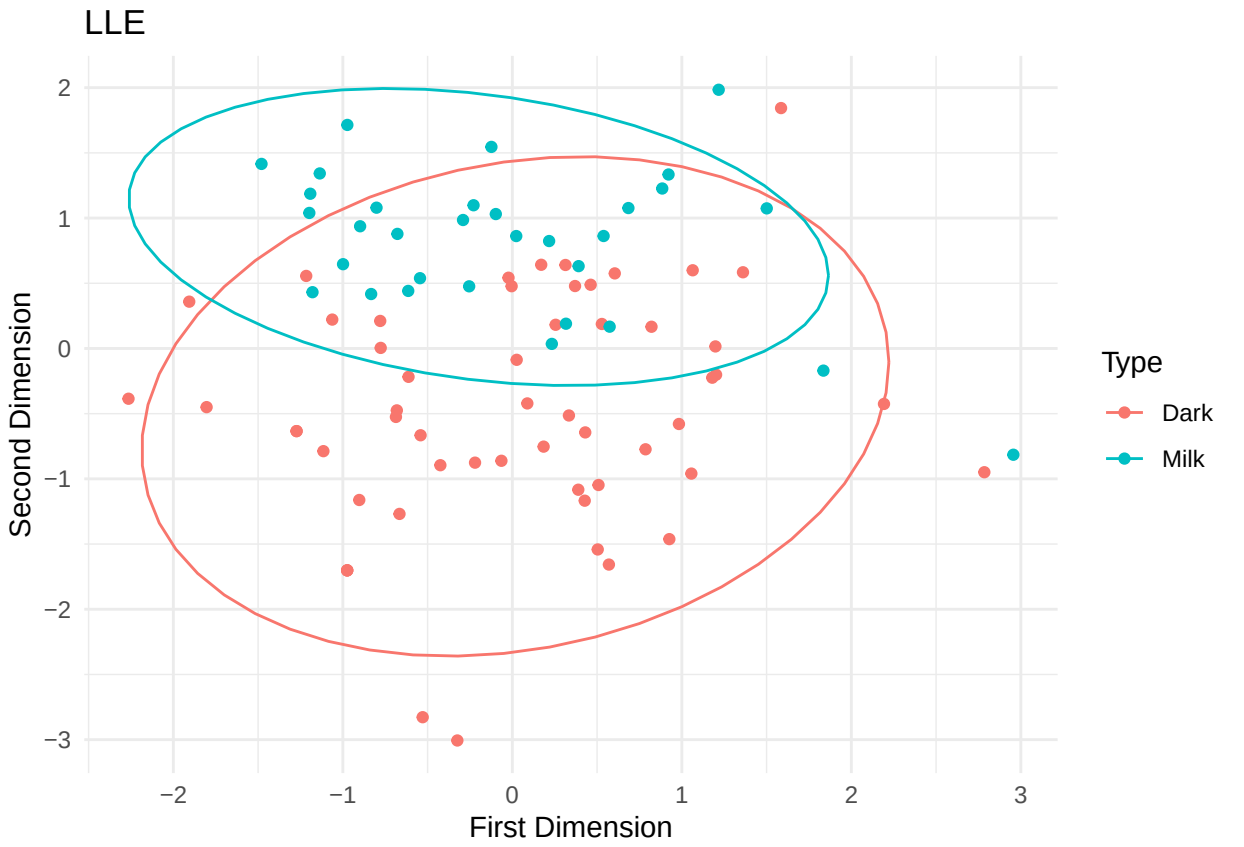
Question 3: Setting k = 10 and m = 2. Plot the results with color varying by chocolate Type.

```
{
  tic()
  lle_fit2 <- lle(cho_scaled[,2:11],
                  m = 2,
                  nnk = TRUE,
                  k = 10)
  toc()
}
```

```
## finding neighbours
## calculating weights
## computing coordinates
## 0.086 sec elapsed
```

```
cho_scaled %>%
  tibble() %>%
  ggplot(aes(x = lle_fit2$Y[,1],
             y = lle_fit2$Y[,2],
             col = Type)) +
  geom_point() +
  stat_ellipse() +
```

```
  labs(x = "First Dimension",
       y = "Second Dimension",
       title = "LLE") +
  theme_minimal()
```



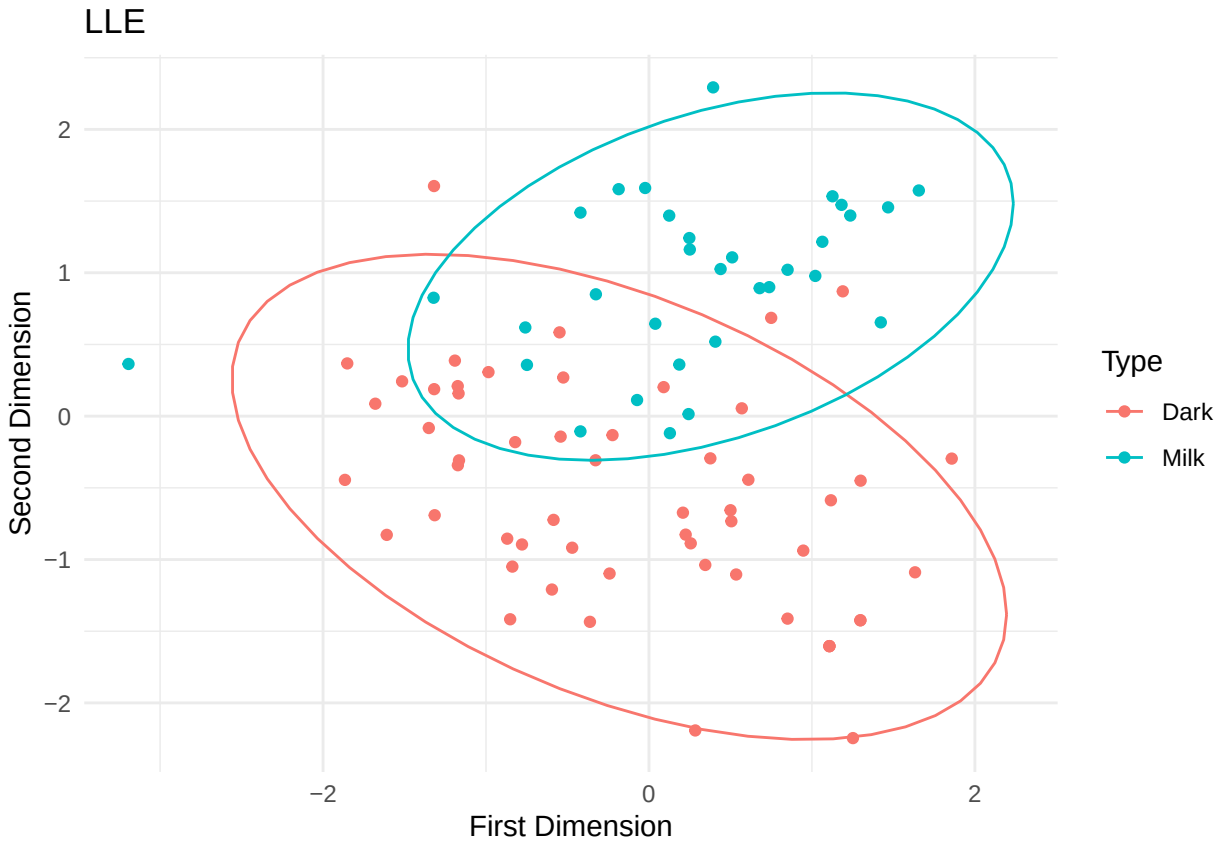
In contrast to k = 2, we see a clearer separation between types of chocolate when k = 10, but the clusters have certain overlaps. Properly increasing k is important to cope with noise and obtain smoother boundary, and hence better generalization performance.

Question 4: Setting k = 50 and m = 2. Plot the results with color varying by chocolate Type.

```
{
  tic()
  lle_fit3 <- lle(cho_scaled[,2:11],
                  m = 2,
                  nnk = TRUE,
                  k = 50)
  toc()
}
```

```
## finding neighbours
## calculating weights
## computing coordinates
## 0.168 sec elapsed
```

```
cho_scaled %>%
  tibble() %>%
  ggplot(aes(x = lle_fit3$Y[,1],
             y = lle_fit3$Y[,2],
             col = Type)) +
  geom_point() +
  stat_ellipse() +
  labs(x = "First Dimension",
       y = "Second Dimension",
       title = "LLE") +
  theme_minimal()
```



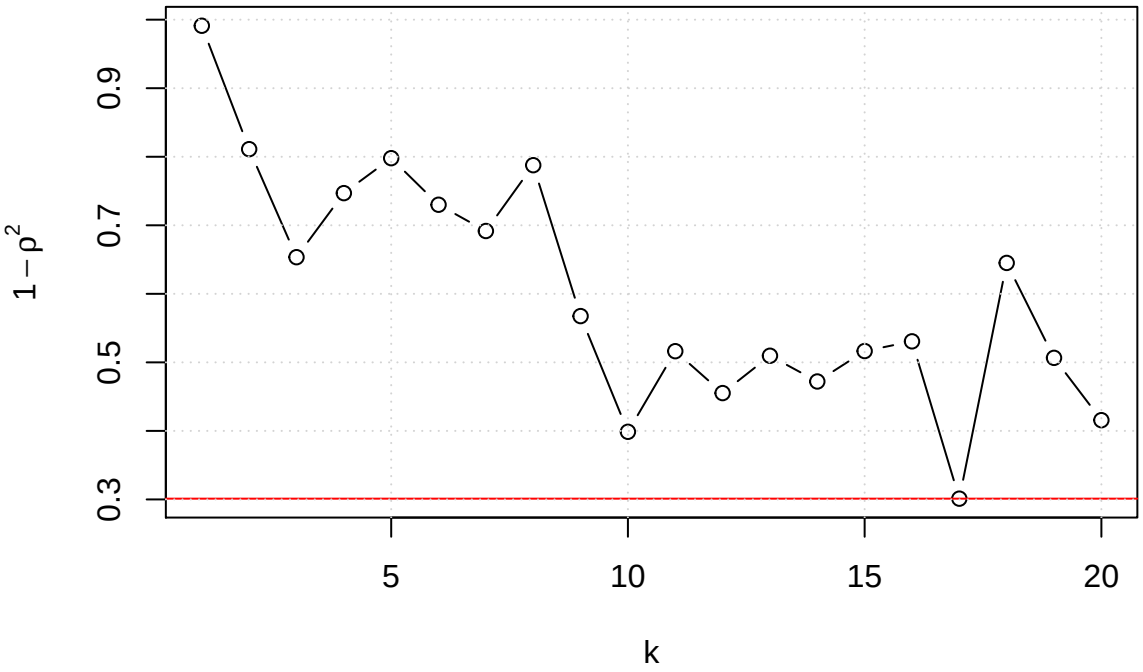
When k = 50, we see a blurry separation between types of chocolate, but the separation is not so clear and there is a heavy overlap between clusters. I think k = 50 in this case is so large, considering that there are only 88 observations. The number of nearest neighbors should be small enough so the samples of the other classes are not included.

Question 5: Conduct a grid search to home in on the optimal value of k.

```
# find the optimal k
cores <- detectCores() - 1
{
tic()
find_k <- calc_k(cho_scaled[,2:11],
                 m = 2,
                 parallel = TRUE,
```

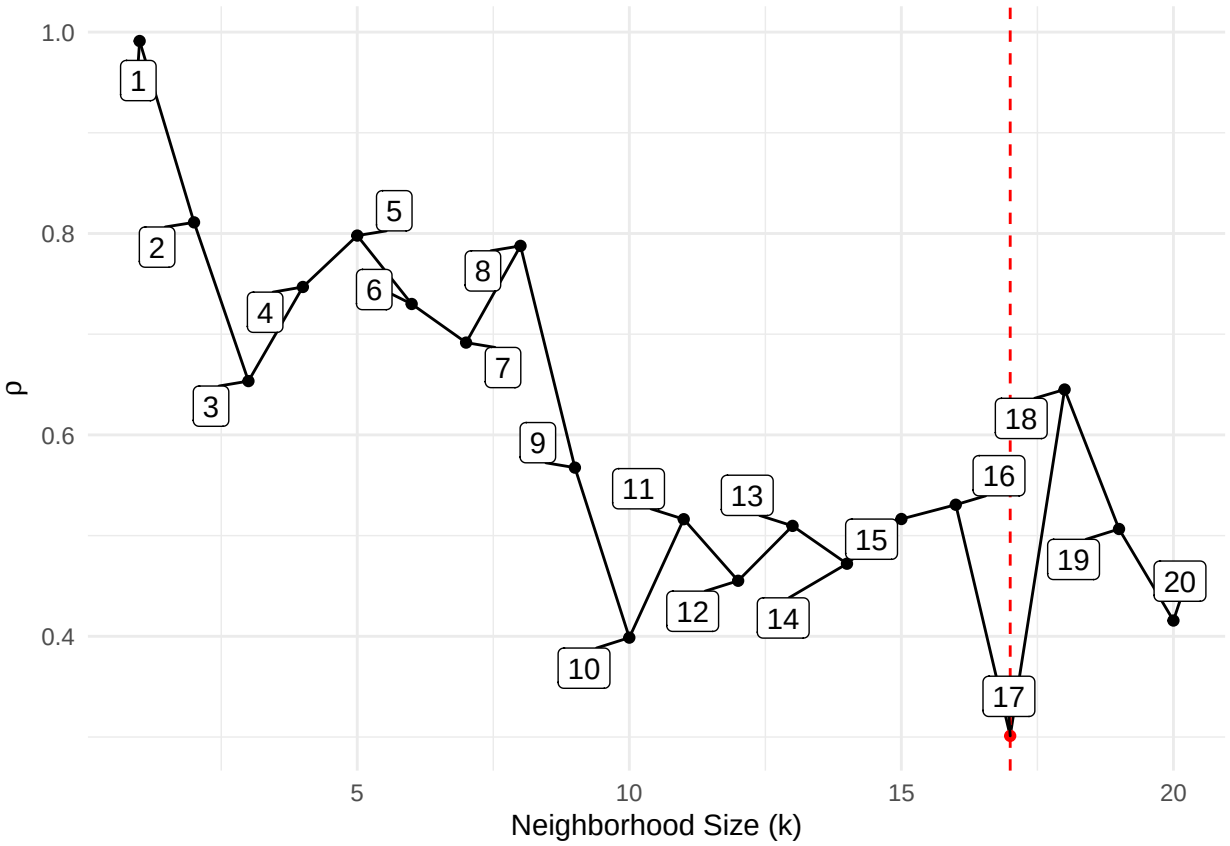
```
                cpus = cores)
toc()
}
```

```
## R Version:  R version 3.6.1 (2019-07-05)
##
## Library lle loaded.
```




```
## 2.062 sec elapsed
```

```
# extract optimal k based on rho
optimal_k_rho <- find_k %>%
  arrange(rho) %>%
  filter(rho == min(.))
# visualization
find_k %>%
  arrange(rho) %>%
  ggplot(aes(k, rho)) +
  geom_line() +
  geom_point(color = ifelse(find_k$k == min(find_k$k),
                            "red",
                            "black")) +
  geom_vline(xintercept = optimal_k_rho$k,
             linetype = "dashed",
             color = "red") +
  geom_label_repel(aes(label = k),
                   box.padding = unit(0.5, 'lines')) +
  labs(x = "Neighborhood Size (k)",
       y = expression(rho)) +
  theme_minimal()
```



Using a LLE's version of grid search, we find that the optimal value of k is 17.

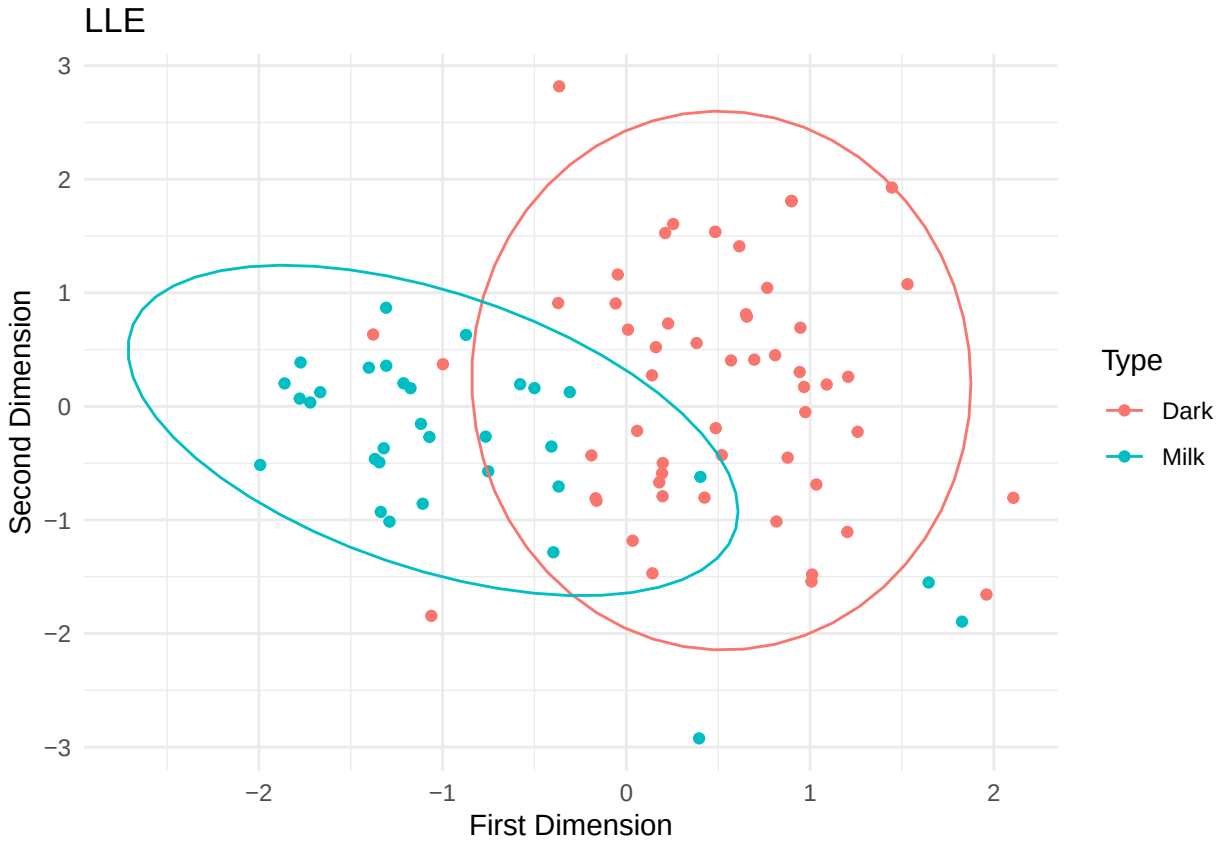
Question 6: Setting k = optimal_k found from answering the previous question and setting m = 2 again. Plot the results with color varying by chocolate Type.

```
# fit
{
  tic()
  lle_fit4 <- lle(cho_scaled[,2:11],
                  m = 2,
                  nnk = TRUE,
                  k = 17)
  toc()
}
```

```
## finding neighbours
## calculating weights
## computing coordinates
## 0.067 sec elapsed
```

```
# viz
cho_scaled %>%
  tibble() %>%
  ggplot(aes(x = lle_fit4$Y[,1],
             y = lle_fit4$Y[,2],
             col = Type)) +
  geom_point() +
```

```
  stat_ellipse() +
  labs(x = "First Dimension",
       y = "Second Dimension",
       title = "LLE") +
  theme_minimal()
```



We see a clearer separation between types of chocolate from this "optimal" version of LLE (k = 17). The data points are evenly scattered over relatively narrow ranges in both dimensions. This plot tells us something about the data structure: there is a separation/ difference between dark chocolate and milk chocolate in the 10 selected features. This means that the dark chocolates of all brands together are different in these ten features from the milk chocolates of all brands together. `Type` is a good categorical variable that classifies chocolates into different groups, where each group has certain common features.

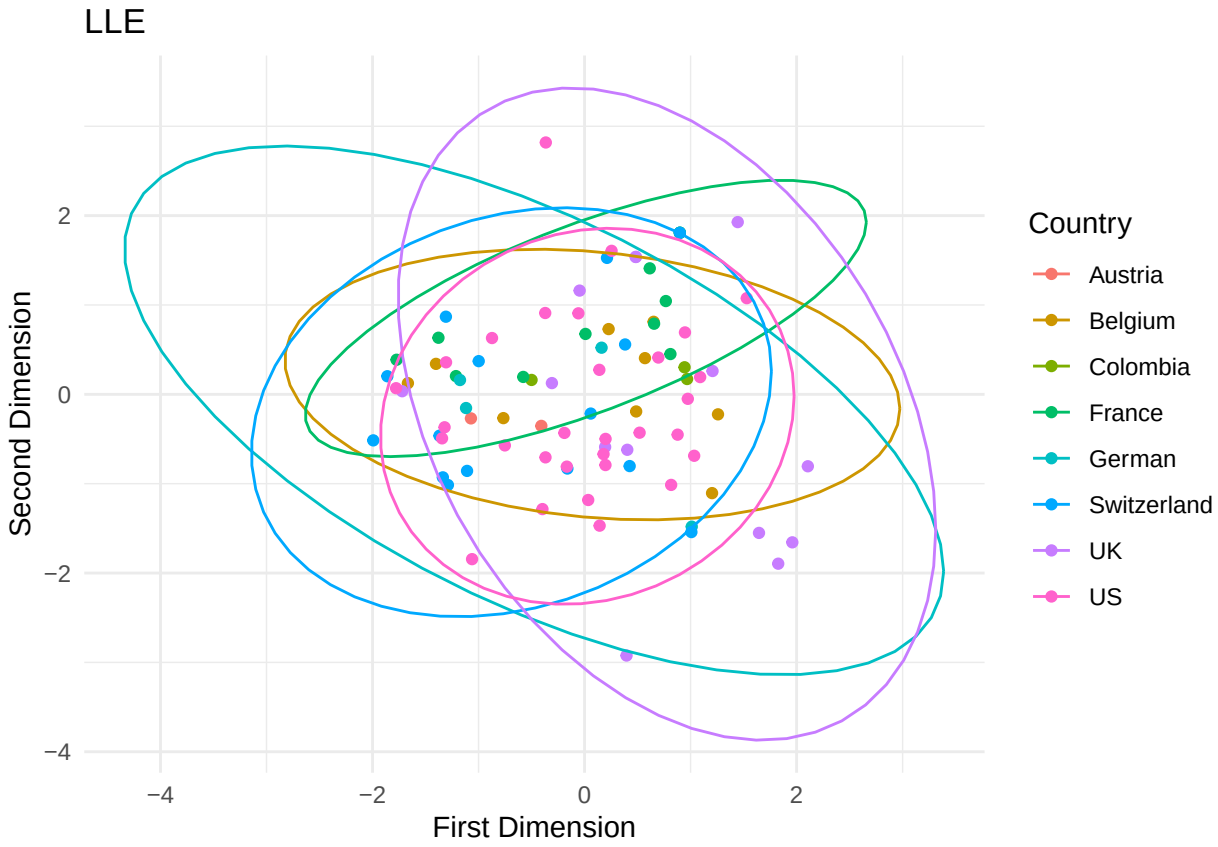
Question 7: Finally, plot the results from your optimal LLE fit again in two dimensions, but this time color the points by Country.

```
cho_scaled %>%
  tibble() %>%
  ggplot(aes(x = lle_fit4$Y[,1],
             y = lle_fit4$Y[,2],
             col = unlist(chocolates[,3]))) +
  geom_point() +
  stat_ellipse() +
  labs(x = "First Dimension",
       y = "Second Dimension",
       title = "LLE",
```

```
      col = "Country") +
  theme_minimal()
```

```
## Too few points to calculate an ellipse
## Too few points to calculate an ellipse
```

```
## Warning: Removed 2 row(s) containing missing values (geom_path).
```



I don't see similar patterns of separation in the projection space as I did when coloring by Type of chocolate in the previous question. There is no clear separation when we color the point by `Country`. This plot tells us that there is no separation/ difference between chocolates from different countries in the 10 selected features. `Country` is not a good categorical indicator that can classify chocolates into different groups, where each group has certain common features.