"Big Data Assignment 2"

Devon Grossett

Question 1: Manifold Learning

For this question I have selected the Isolet dataset from the UCI Machine Learning Repository. It contains data from 150 subject who were recorded speaking each letter of the English alphabet twice, for a total of 7,797 observations (3 observations are missing). The purpose of this dataset is to train a model to classify recordings of speech by what letter of the alphabet is being said. The data has already been split into a training set of 6,238 observations (120 subjects) and a training set of 1,559 observations (30 subjects). I will limit my analysis to the training set.

There are 617 features, the majority of the which (448) are discrete Fourier transform coefficients of different parts of the waveform corresponding to the different sonorant intervals (SON) of each letter. The other features are a variety of other wave measurements corresponding to different SON, such as zero-crossing rate, amplitude, and duration. Detailed information can be found in the paper by Fanty and Cole (1990).

This dataset provides a highly dimensional problem with which to test different dimensionality reduction techniques, with applications to the real world. Voice recognition software is becoming more widespread with voice assistants such as Alexa and Siri, which rely on models to translate voice recordings into text that it can parse.

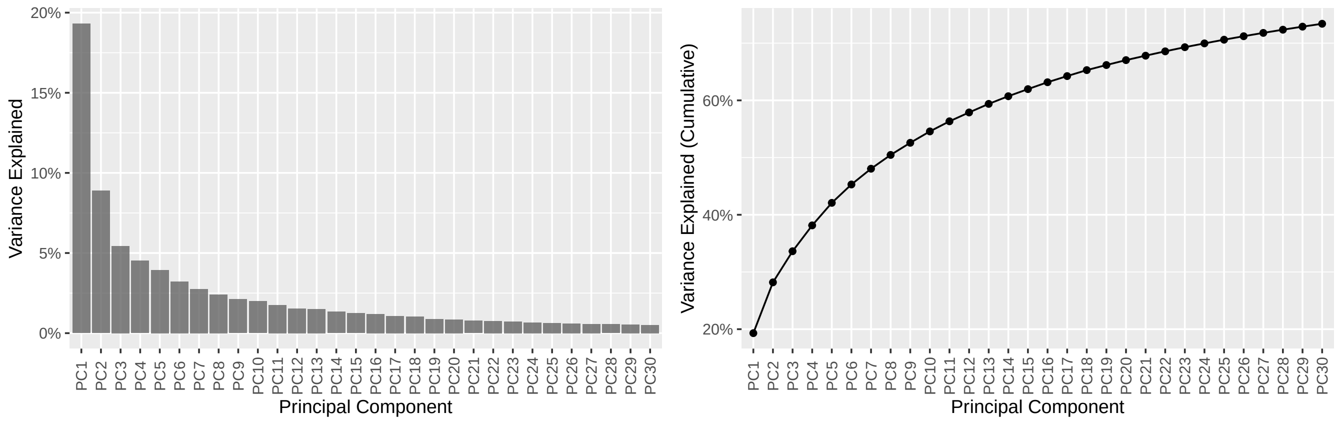
Principal Component Analysis (PCA)

```r
X <- scale(isolet_features)
pc <- prcomp(X)

# get transformed data and add back on target class
P <- data.frame(pc$x) %>%
  mutate(class = as.factor(isolet_target$class))

# percent of variance explained by PCs
var_explained <- data.frame(pc = factor(paste0("PC", 1:(dim(X)[2])),
                                        levels = paste0("PC", 1:(dim(X)[2]))),
  pct_var = pc$sdev**2 / sum(pc$sdev**2)) %>%
  mutate(cum_sum = cumsum(pct_var))
```

Here we have performed principal component analysis on our training set. This is a linear technique that finds linear combinations of the original features that are uncorrelated and maximise the variance of the new principal components. As shown in Figure 1a, the first principal components contains 19% of the variance of overall variance in our data, then 9% for the next principal component. From Figure 1b, we can also see that cumulatively, half of the variance is explained by the first 8 principal components, although



(a) Individual %

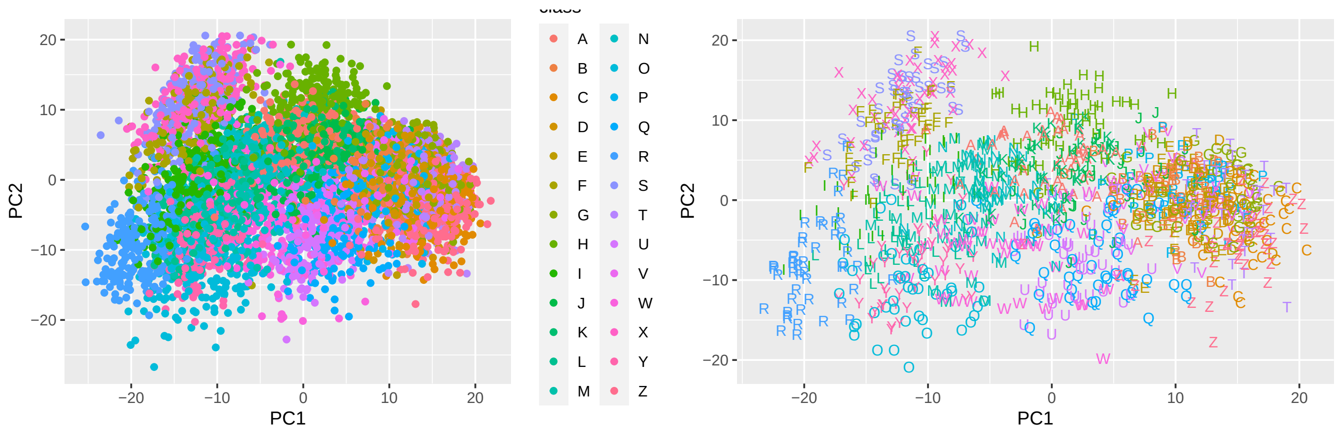
(b) Cumulative %

Figure 1: Variance explained by successive principal components (up to first 30)

this tapers off and it isn't until the 112th principal component has been added that 90% of the overall variance in our data is explained by its principal components.

Figure 2 plots the target classes against the first two principal components of the data. Due to the large number of classes, and the number of data points plotted, I also found it useful to take a stratified sample of 20% of the data to declutter the plot as well as using the letter to identify which class each plot point is. I will also do this when analysing the other methods in this report to make the visualisations clearer.

We do see some clusters of letters forming in Figure 2b. In the bottom left, "R" has been quite well isolated, which I suspect is due to it having quite a unique sound when spoken. In contrast to this, on the right hand side of Figure 2b, there is a mix of letters like "B", "C", "D", "E", "P", "T", and "Z" clustered together. These letters all share a common "ee" sound on the end when spoken, which explains why they have been grouped together, although it would be good to find a dimensionality reduction technique that is further able to separate them based on the different sounds made before the "ee" sound.



(a) All data - no labels

(b) 20% of data, with labels to visualise better

Figure 2: Data plotted against first two principal components

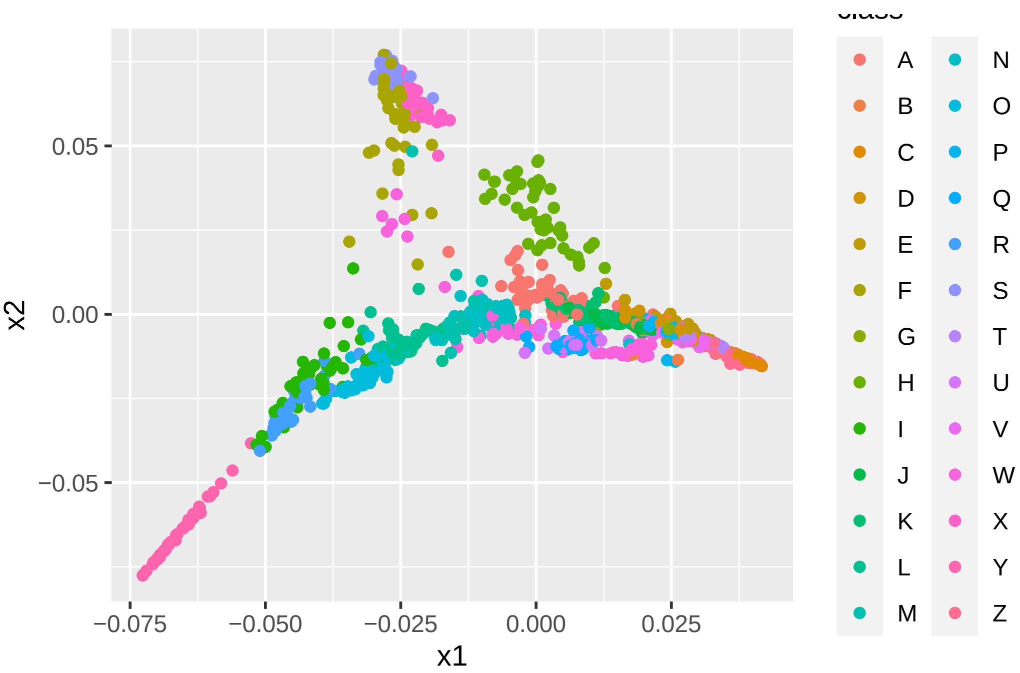
Locally Linear Embedding (LLE)

```r
# need to sample the data in order to get LLE to run in a reasonable time
lle_samp <- caret::createDataPartition(isolet_target$class, p = 0.2,
                                       list = FALSE)

isolet.lle <- do.lle(X[lle_samp, ], type = c("knn", 10))

isolet.lle.data <- data.frame(x1 = isolet.lle$Y[, 1], x2 = isolet.lle$Y[, 2],
                              class = isolet_target[lle_samp, ]$class)

ggplot(isolet.lle.data, aes(x = x1, y = x2, color = class)) + geom_point()
```



Fanty, Mark, and Ronald Cole. 1990. "Spoken Letter Recognition." In *NIPS*, 3:220–26. https://doi.org/10.3115/116580.116725.