Force Plate Data Dimension Reduction Report

Lauren Temple, Andrew Sisitzky, Rachel Donahue

5/9/2022

Introduction

This report will provide an explanation of the techniques our team used to perform dimension reduction and data reconstruction. The data we were provided for this project was force plate data compiled by a team of graduate students at the Boston University Sargent College of Health and Rehabilitation Services. Our team chose to use the time normalized stance data in our data reconstruction. Each dataset includes trial records for 15696 individuals. Each trial spans 100 time normalized data points.

Since the dataset consisted of a large number of features for each trial, our team used PCA (Principal Component Analysis), Autoencoders and fPCA (functional Principal Component Analysis) techniques to perform dimension reduction. This enabled us to see which features were most significantly contributing to the data, and to reduce highly correlated measurements. We were also able to perform reconstruction on the PCA and the Autoencoder, which allowed us to see how well the dimension reduction was performing.

Why is Reconstruction Useful?

This report will show reconstruction error in order to quantify the loss of the reconstructed data. This data is reconstructed using a reduced number of dimensions so it is important to understand how well the reconstruction represents the original data set. After completing PCA the data is reconstructed using a specified number of principal components. The number of principal components used was based on the percentage of variance in the data covered by each component. The Autoencoder reconstructs data using weights assigned to the reduced dimensions during the bottleneck layer of the network. From there the data is decoded and we can compute the reconstruction error.

PCA Reconstruction Error

Our team began this project by performing a PCA of the Vertical Ground Reaction Force (V GRF) Stance time normalized data. PCA is a technique that involves computing principal components, which are linear combinations of the original variables, and then finding the optimal number that allows for the most information to be included in each component. From our PCA we then reconstructed the data to calculate the reconstruction error. This will tell us how much error there is when the data is reconstructed using a reduced number of dimensions. The plots below illustrate the PCA performance in two ways. The first shows the percentage of variation explained by each of the principal components, the first two combined explained over 90% of the variance. The second shows the mean squared reconstruction error, which decreases substantially with k additional principal components.

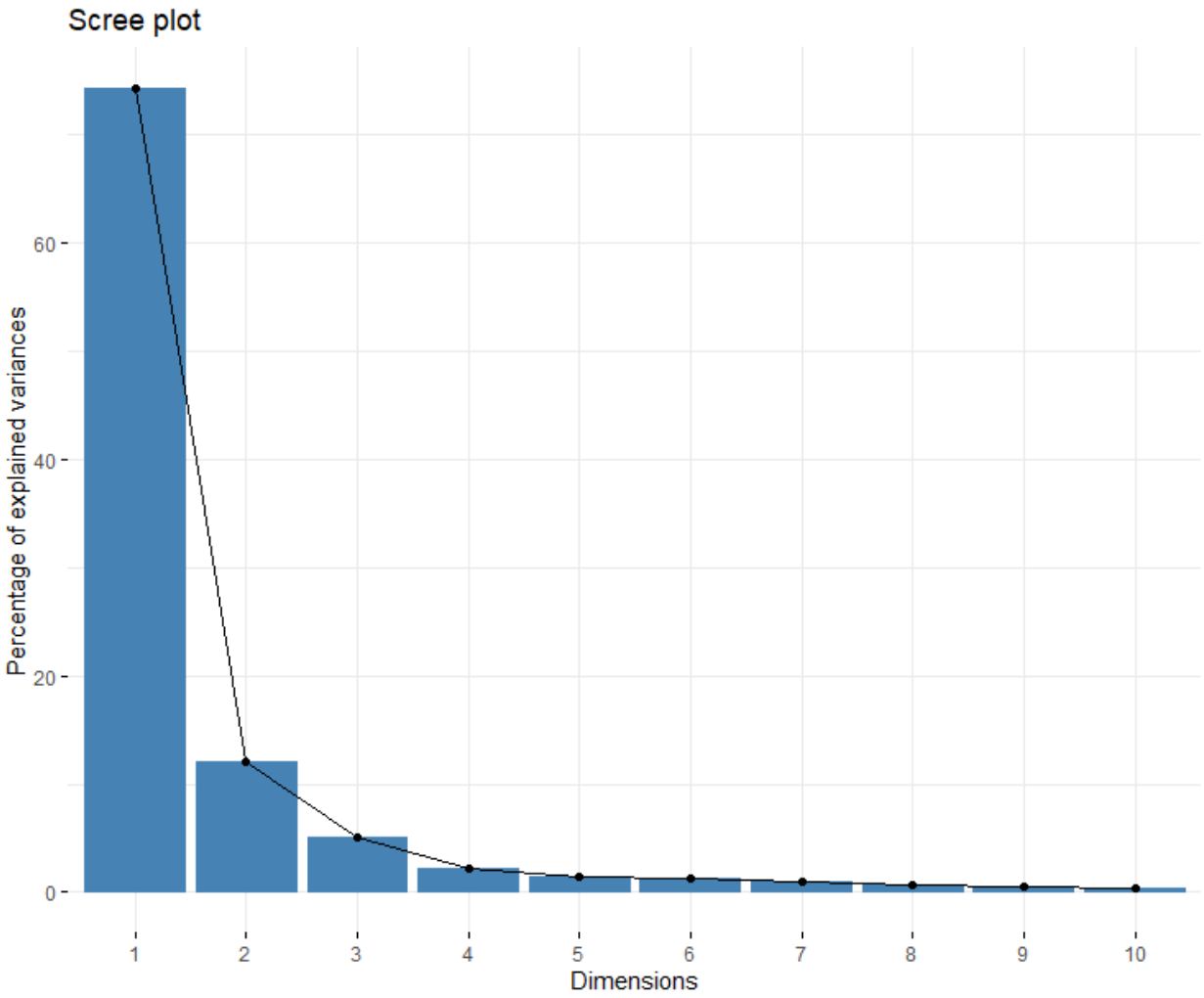

Figure 1: PCA Scree Plot

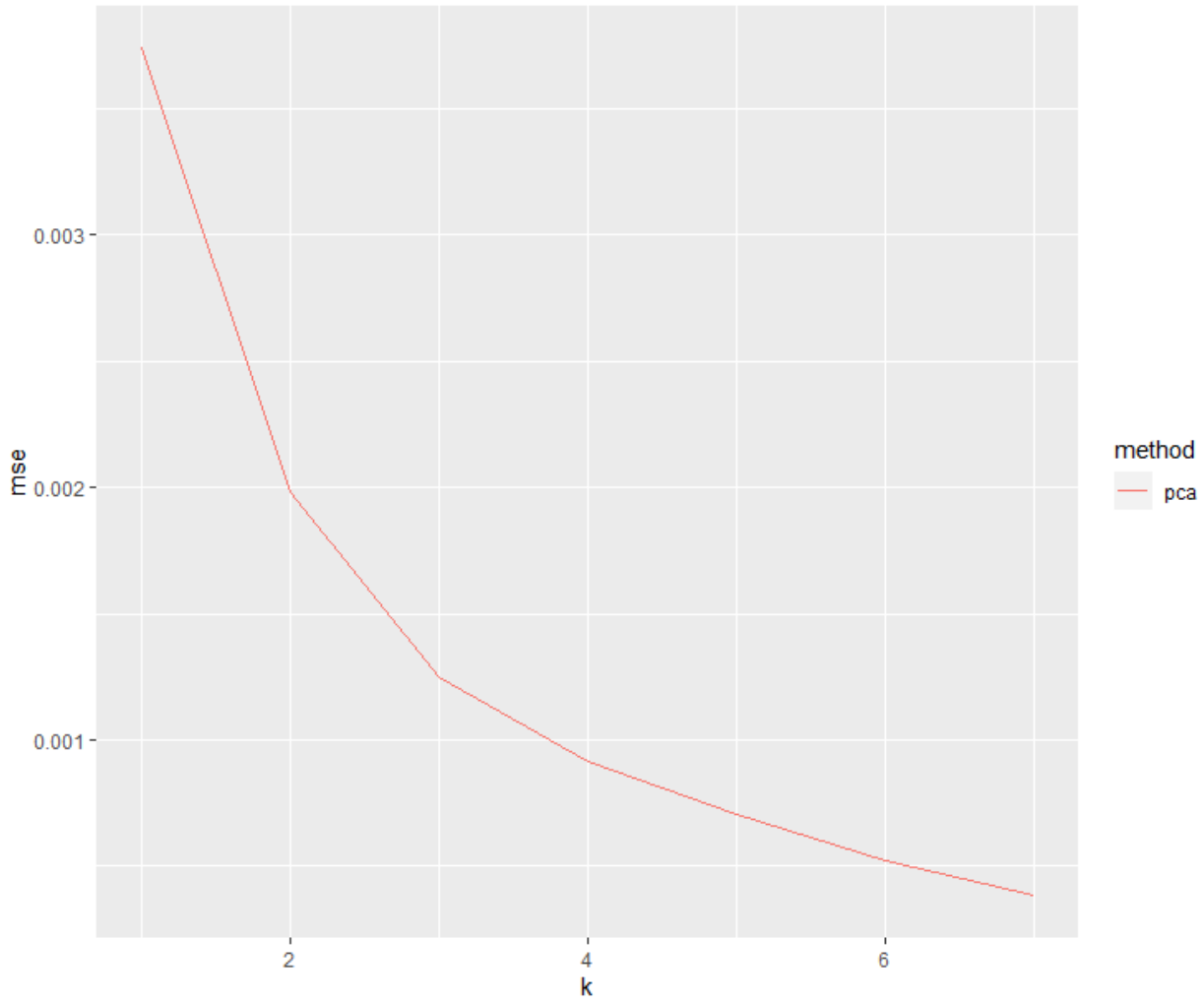

Figure 2: PCA Loss

Autoencoder

Our team created a simple autoencoder as a second method to perform dimension reduction on the data. Autoencoders are unsupervised artificial neural networks that learn how to efficiently compress and encode data and then reconstruct the data from the reduced encoded representation of that data. An autoencoder will consist of four main parts, encoder, bottleneck, decoder, reconstruction loss. The bottleneck layer contains the compressed representation of the input data and has the lowest possible dimensions of the input data.

The autoencoder for this project used the V GRF Stance time normalized data. The parameters of the autoencoder are shown below. The bottleneck layer assigned weights to the encoded data. These weights were then used to decode the data using a reduced number of dimensions. We calculated the reconstruction error of the autoencoder using the mean squared error and got a loss of 0.0004. We decided to rerun this model with varying values for the bottleneck layer to see if we could get a lower reconstruction error but found that regardless of the bottleneck layer size the reconstruction loss stayed the same.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_2 (Dense)              (None, 49)                4949
_________________________________________________________________
bottleneck (Dense)           (None, 7)                 350
_________________________________________________________________
dense_1 (Dense)              (None, 49)                392
_________________________________________________________________
dense (Dense)                (None, 100)               5000
=================================================================
Total params: 10,691
Trainable params: 10,691
Non-trainable params: 0
_________________________________________________________________
```

Figure 3: Autoencoder Parameters

```
> mse.ae2 <- evaluate(model, x_train, x_train)
344/344 [==============================] - 1s 2ms/step - loss: 4.3319e-04
> mse.ae2
        loss
0.0004331928
```

Figure 4: Autoencoder Loss

```
Model: "sequential_15"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_47 (Dense)             (None, 49)                4949
_________________________________________________________________
bottleneck (Dense)           (None, 7)                 350
_________________________________________________________________
dense_46 (Dense)             (None, 49)                392
_________________________________________________________________
dense_45 (Dense)             (None, 100)               5000
=================================================================
Total params: 10,691
Trainable params: 10,691
Non-trainable params: 0
_________________________________________________________________
```

Figure 5: Autoencoder Parameters Varying Bottleneck Size 1:7

Figure 6: Autoencoder Loss Varying Bottleneck Size 1:7

fPCA

Functional Principal Component Analysis was done due to the sequential nature of the data set. In this dataset there are about four trials per participant and each trial spans 100 time normalized data points. It is important to consider each trial as a whole. FPCA was done on each of the time normalized data sets. Our fPCA was created using twenty knots, four basis functions, and b-splines. This setup was consistent with each of the time normalized stance datasets. After completing the fPCA for each dataset we looked at the variation proportion covered by each principal component. In order to keep our results interpretable we aimed to cover about 90% of variance with the least number of principal components. We found that each data set could be explained using at most five principal components. With five principal components we could explain about 90% of the variation in the data for the Medial Lateral Ground Reaction Force Stance (ML GRF) data as well as the Anterior Posterior Ground Reaction Force Stance (AP GRF) data. The V GRF data was able to be explained to the same degree with only two principal components. Due to the way our fPCA was constructed we were unable to reconstruct the data. If we were given more time we could construct our fPCA differently so that the data may be reconstructed. If this is done then we would be able to compare the reconstruction error of our fPCA to that of the auto encoder. When reconstructing the fPCA, our team would split the data such that participant IDs do not cross over from the training data to the testing data. This is important because of any correlation that may exist between participant trials.

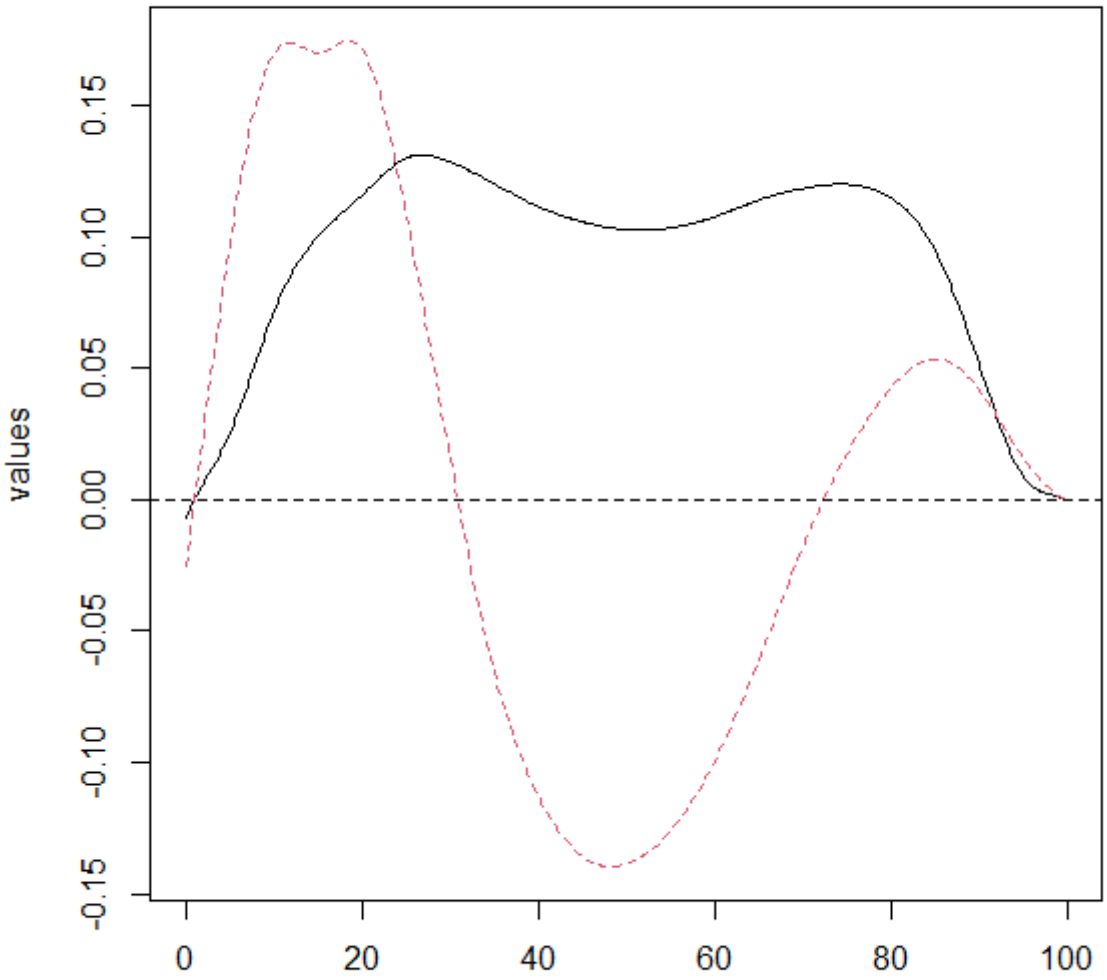

Figure 7: V GRF Stance fPCA Results

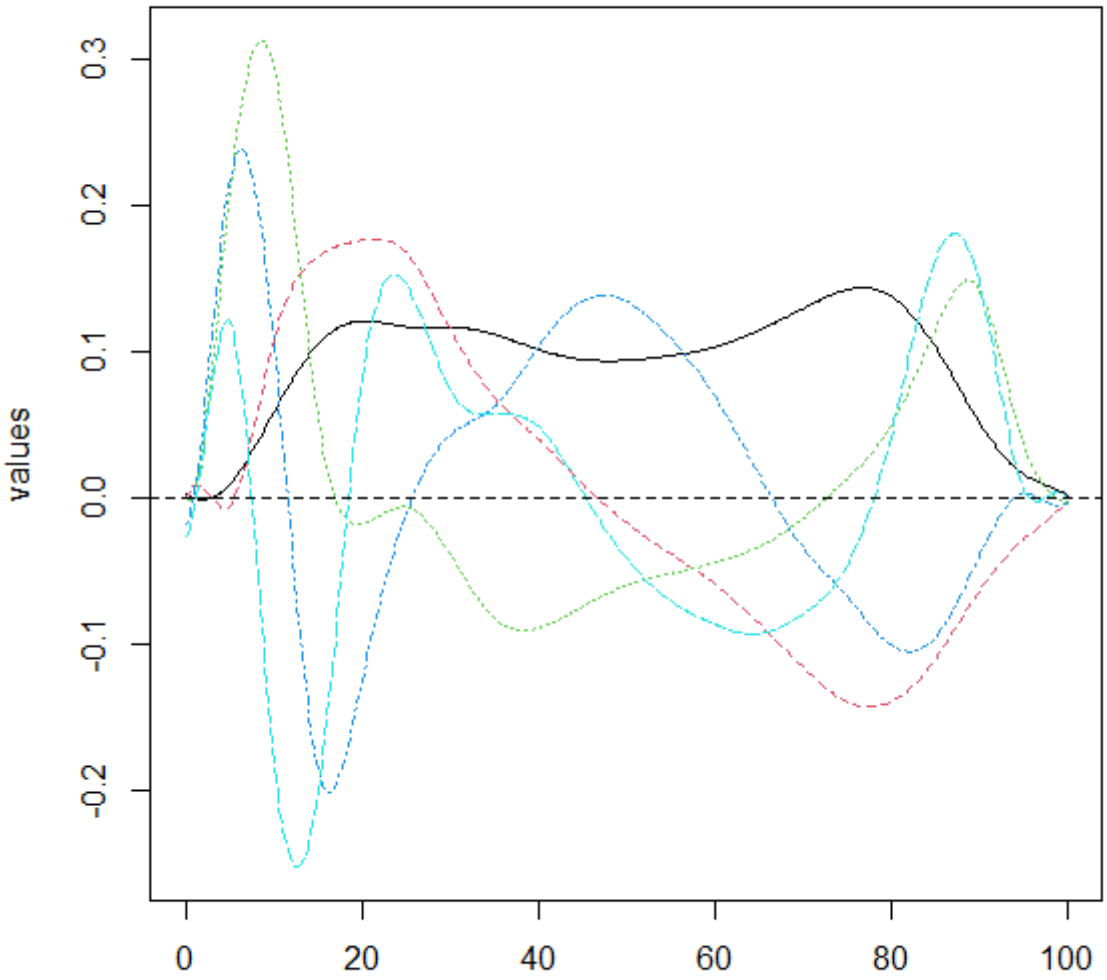

Figure 8: ML GRF Stance fPCA Results

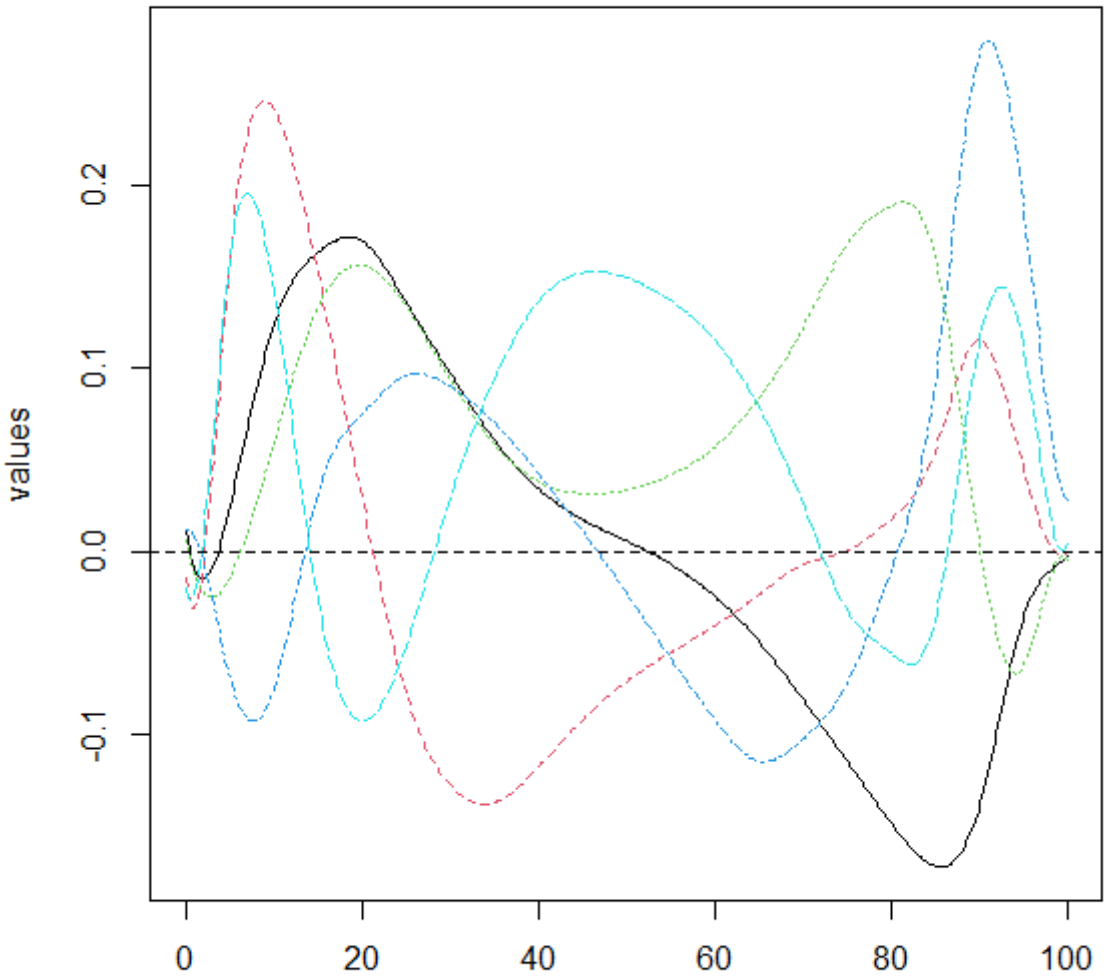

Figure 9: AP GRF Stance fPCA Results

Conclusion

In this report we provided the client with a review of the techniques used in our dimension reduction and data reconstruction of their force plate data using time normalized data files. We concluded that the Autoencoder and PCA were able to reconstruct the data with similar accuracy when seven principal components are used on the V GRF Stance data. We did not extend that analysis to other datasets. We also showed that we can get more meaningful results using fPCA as this method takes into account the sequential nature of the datasets. However at this time we were unable to reconstruct the data using our fPCA. Our report provides the client with multiple techniques that provide valid results in reducing the dimensions of their dataset. By reducing the number of features in the dataset, this provides a better basis for additional data analysis that is more computationally cost efficient.