BME 6717 Dataset 2 Spike Sorting with SVD

Gilgal Ansah

February 2022

Analysis

Visualization

All 300 recordings are visualized at once to get a sense of the overall structure of the data set.

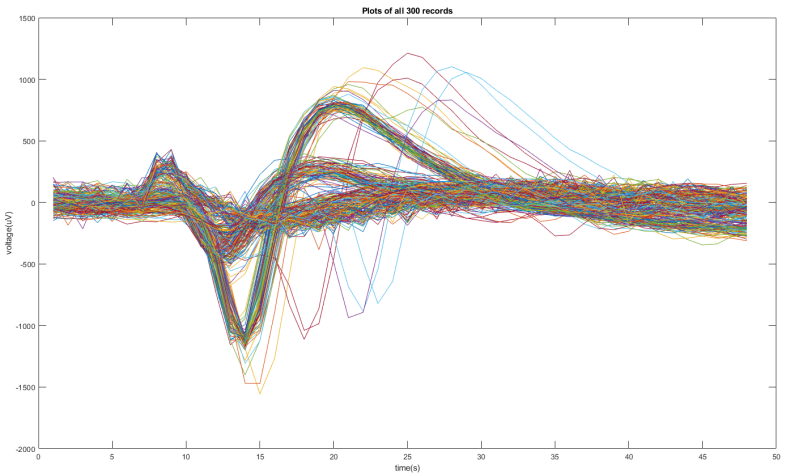

Figure 1: A plot of all 300 records

Figure 1 shows patterns of action potentials - signals having some major negative and positive peak. However, the first 15 seconds show some noisy (jagged) patterns in the signals.
4 random observations are chosen (using the MATLAB random number generator) and visualized in figure 2.

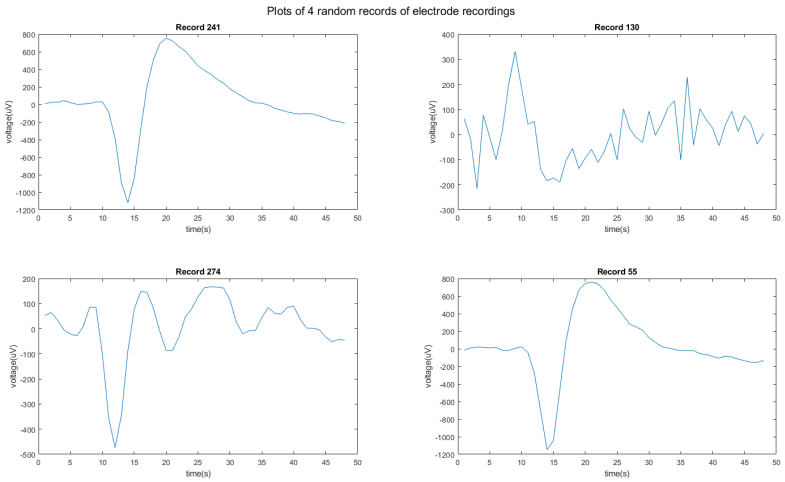

Figure 2: A plot of 4 randomly selected records

Record 241 shows a clear negative and positive peak. As does 55. In all these plots, however, there is evident noise (jaggedness) in the signal. The noise is very prominent in Record 130.

Noise Removal with PCA

To remove the noisy features in the signal, we conduct a principal component analysis to find the most important components of the signals.
The different components and their contribution to variance in the data are shown in the bar plots in figure 3.

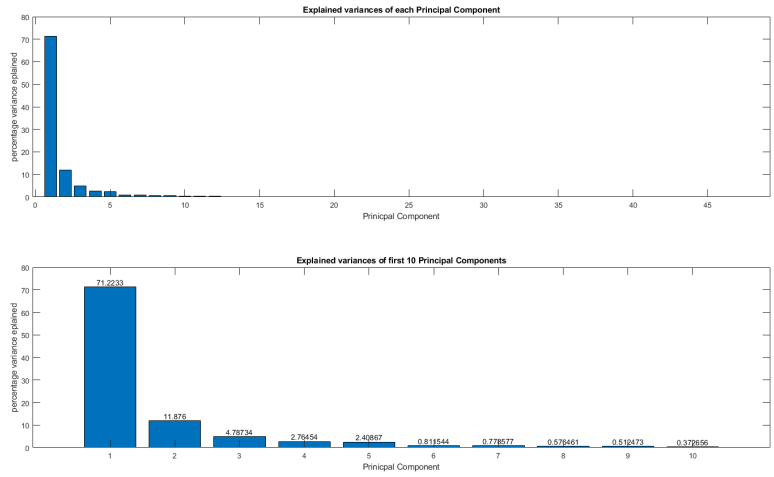

Figure 3: Subplot 1 shows the explained variances of all principal component. Subplot 2 shows the variances of the first 10

From the above plot, the first 4 principal components contribute to 90% of the total data. The signals were reconstructed with the first 1-4 components and visualized.

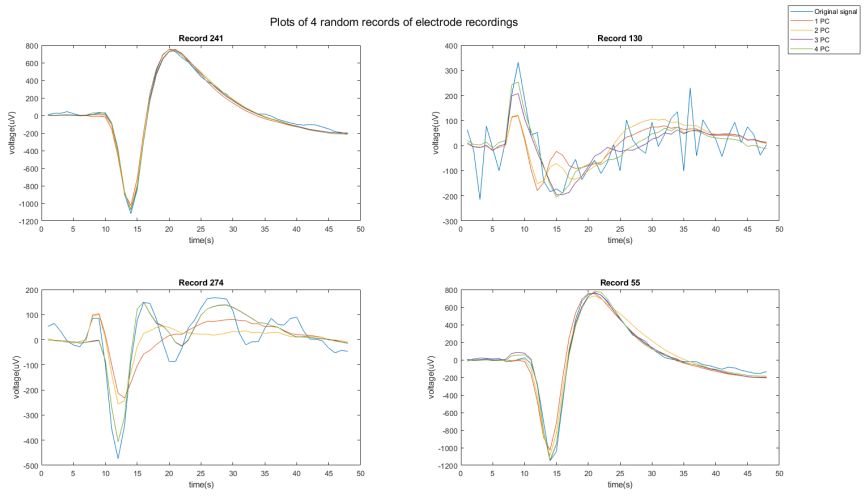

Figure 4: A plot of 4 randomly selected records with reconstructed data sets

From figure 4, the data reconstructed with 1 components seems to lose some information.In Record 274 for instance, the signal amplitude is greatly reduced. Using 3 and 4 components seem to capture some noise in the data. From $t = 15s \quad to \quad t = 30s$, in Record 130, there is evident jaggedness in the signals.
Using 2 Principal Components seemed to be a good balance between noise removal and retaining relevant information about the signals.
However, a plot of all signals using 1,2,3 and 4 Principal Components showed clear distinction between signal groups when 1 Principal Component was used.

Estimation

Looking at figure 5, the 300 records cluster into 3 different groups. Just before the $15s$ mark, two of the groups have a prominent negative peak whiles the other has a prominent positive peak.
Between $t = 20s \quad and t = 25s$, the groups with the initial negative peaks have a positive peak while the group with the positive peak shows a negative peak. Similar trends are shown in the reconstructed data using 2,3 and principal

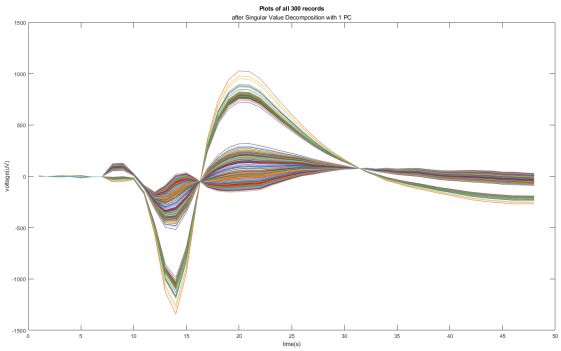

Figure 5: Data reconstructed with 1 Principal Component

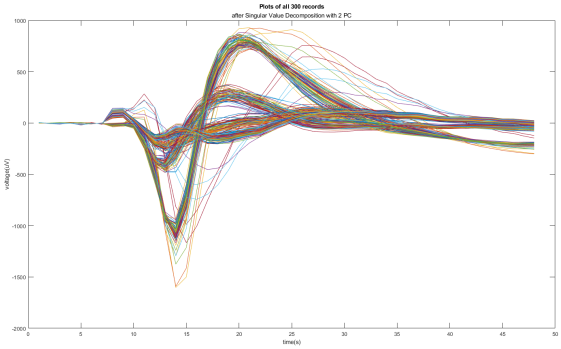

Figure 6: Data reconstructed with 2 Principal Components

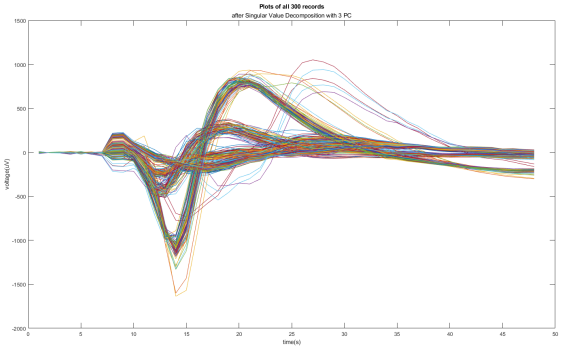

Figure 7: Data reconstructed with 3 Principal Components

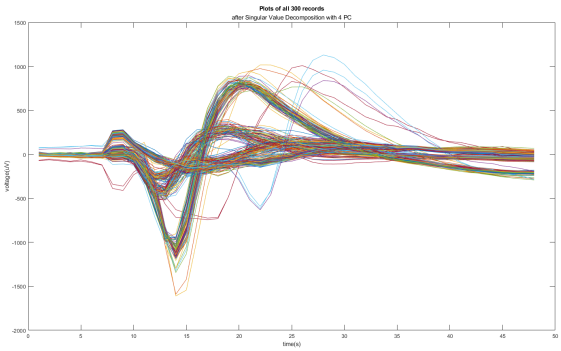

Figure 8: Data reconstructed with 4 Principal Components

components.
By inspection, there are an estimated 3 neurons in the data set.
With this a priori data, kmeans clustering was performed to show these groups distinctly.

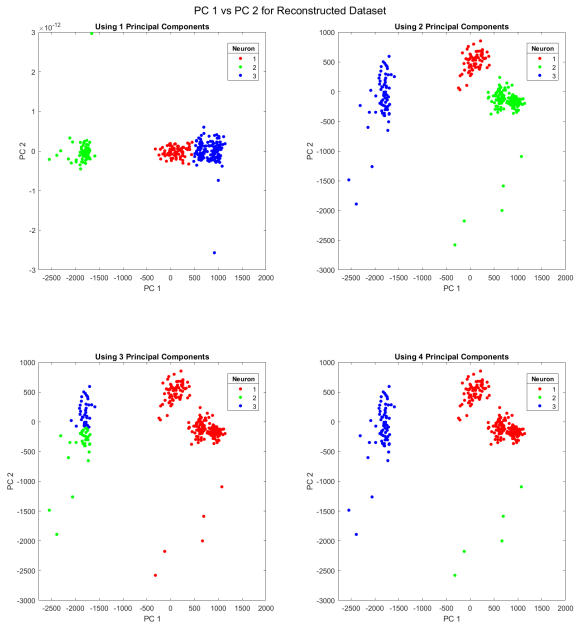

Figure 9: Showing neuron clusters along principal components 1 and 2

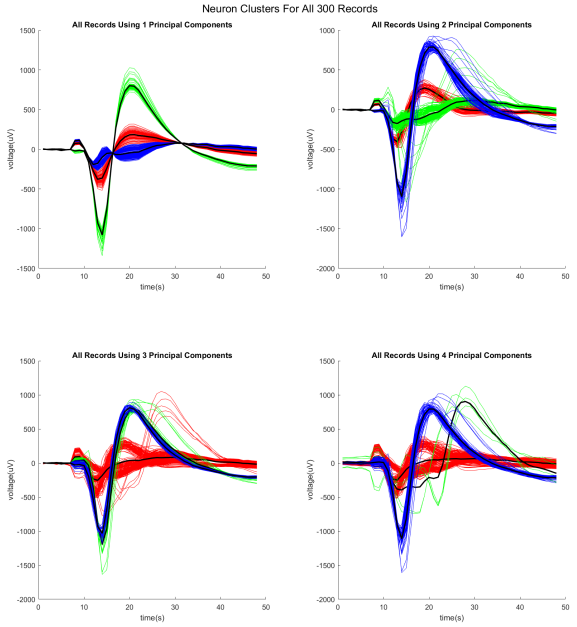

Figure 10: Signal plots showing the different neurons

Conclusion

From figure 9 and 10, using 1 principal components, along with visual inspection, does well to cluster the 300 observations into neuron groups. It is seen that using 4 tends to misclassify some datasets. The PCA plots for 3 and 4 components are similar but with very different clusters. With 1 Principal Component, the clusters are not readily visible in the PCA space, hence visualizing the signals is needed.
It is hard to conclude whether 1 Principal Component is better than 2, given that this is an unsupervised clustering and there is no way to tell which does a better job in grouping the datasets.
Preference is given to 1 Principal Component because there is evident uniformity in the signal plots as compared to using 2 Principal Components.
The mean activity of each neuron is shown as a black line superimposed on the waveform plot.
The neuron clusters in the PCA plot correspond by color to the waveform plots.

Appendix-MATLAB Code

```
%% BME6717 Dataset 2 Spike Sorting with SVD

%% IMPORTING DATA

SpikeData = importdata('NeuronData.mat');

%% PLOTTING
%%
%all records from raw data
figure(1)
plot(SpikeData')
xlabel('time(s)')
ylabel('voltage(uV)')
title('Plots of all 300 records')
%%
%plots of 6 random records of electrode recording
figure(2)
selection=randi(300,1,4);
for i=1:length(selection)
    hold on
    j=selection(i);
    subplot(2,2,i)
    plot(SpikeData(j,:))
    xlabel('time(s)')
    ylabel('voltage(uV)')
    title(['Record ' num2str(j)])
end
sgtitle('Plots of 4 random records of electrode recordings')
%% NOISE REMOVAL - PCA

%computing PCA
[coeff, score, ¬,...
    ¬, explained,mu] = pca(SpikeData);
%mu - column means
%explained varainces by each principal component
%%
%bar plot explained variance from data
figure(3)
subplot(211)
bar(explained)
xlabel('Prinicpal Component')
ylabel('percentage variance eplained')
title('Explained variances of each Principal Component')

subplot(212)
b=bar(explained(1:10));
xlabel('Prinicpal Component')
ylabel('percentage variance eplained')
title('Explained variances of first 10 Principal Components')
xtips1 = b.XEndPoints;
ytips1 = b.YEndPoints;
labels1 = string(b.YData);
text(xtips1,ytips1,labels1,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom')

%%
%Reconstructing from first 1-4 Principal Components
for n=1:4
    SpikeData_new{n} = score(:,1:n) * coeff(:,1:n)' + mu;
end

%%
% Plotting 4 random signals for each reconstructed dataset
figure(2)
for n=1:4
    SpikeData_0=SpikeData_new{n};
    for i=1:length(selection)
        hold on
        j=selection(i);
        subplot(2,2,i)
        plot(SpikeData_0(j,:))
        xlabel('time(s)')
```

```
        ylabel('voltage(uV)')
        title(['Record ' num2str(j)])
    end
end
sgtitle('Plots of 4 random records of electrode recordings')
legend('Original signal','1 PC','2 PC','3 PC','4 PC')
%%
% Plotting reconstructed dataset
figure(4)
plot(SpikeData_new{2}')
xlabel('time(s)')
ylabel('voltage(uV)')
title('Plots of all 300 records')
subtitle('after Singular Value Decomposition with 2 PC')
%%
% Visualizing Clusters
for n=1:4
    SpikeData_0=SpikeData_new{n};
    [coeff, score, latent,...
        tsquared, explained,mu] = pca(SpikeData_0);
    id=kmeans(SpikeData_0,3);

    figure(5)
    sgtitle("Neuron Clusters For All 300 Records")

    subplot(2,2,n)
    hold on
    plot(SpikeData_0(id==1,:)','r')
    plot(mean(SpikeData_0(id==1,:),1),'k',"LineWidth",1.8)
    plot(SpikeData_0(id==2,:)','g')
    plot(mean(SpikeData_0(id==2,:),1),'k',"LineWidth",1.8)
    plot(SpikeData_0(id==3,:)','b')
    plot(mean(SpikeData_0(id==3,:),1),'k',"LineWidth",1.8)

    title(['All Records Using ' ,num2str(n),' Principal Components'])
    xlabel('time(s)')
    ylabel('voltage(uV)')

    figure(6)
    sgtitle("PC 1 vs PC 2 for Reconstructed Dataset")
    subplot(2,2,n)
    gscatter(score(:,1),score(:,2),id)
    xlabel('PC 1'); ylabel('PC 2')
    title(['Using ',num2str(n),' Principal Components'])
    lgnd = legend("1","2","3"); title(lgnd,"Neuron");
end
```