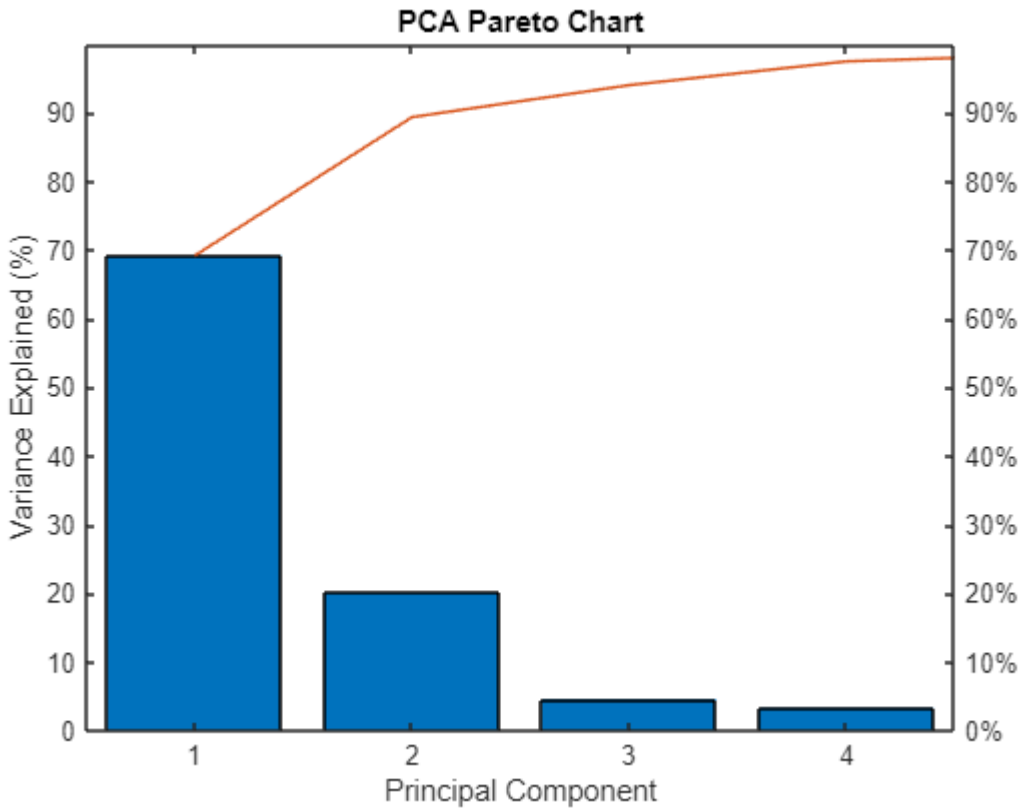
```
% Performing PCA on Training data
[coeff, scoreTrain, ~, ~, explained, mu] = pca(X_train);
```

```
Warning: Columns of X are linearly dependent to within machine precision.
Using only the first 28 components to compute TSQUARED.
```

```
% Select number of components to retain 95% variance
numPCs = find(cumsum(explained) >= 95, 1)
```

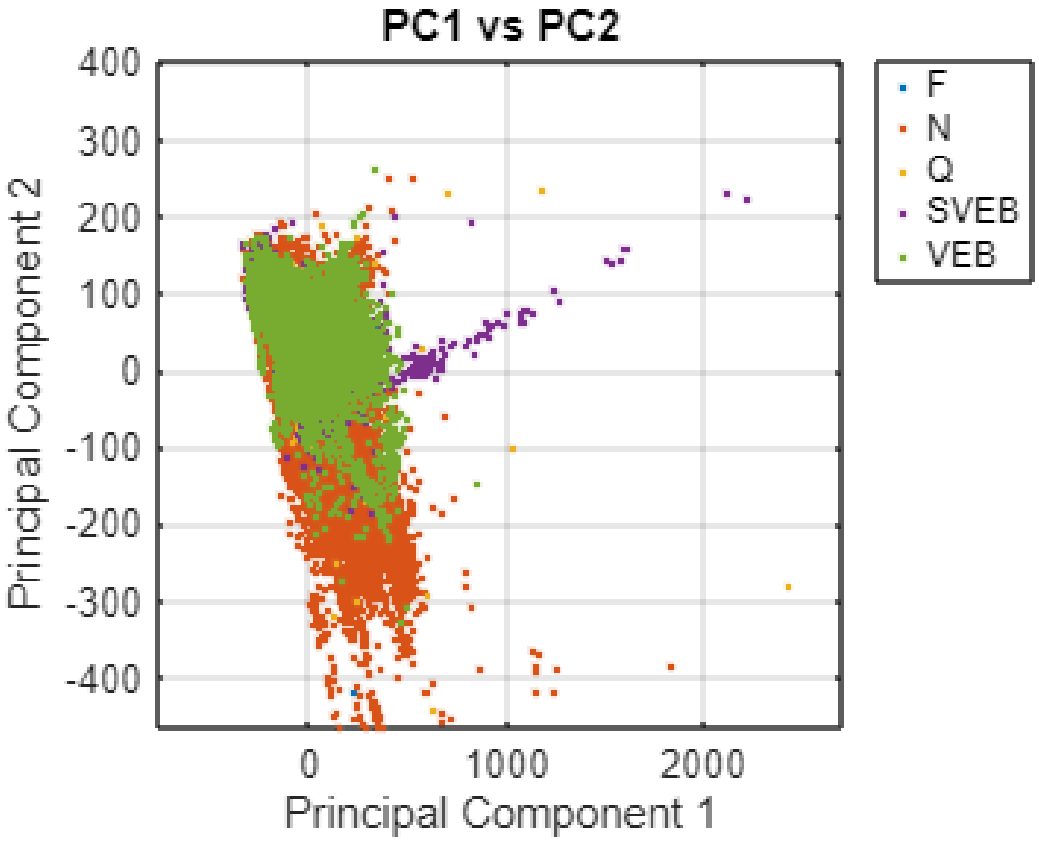
```
numPCs =
4
```

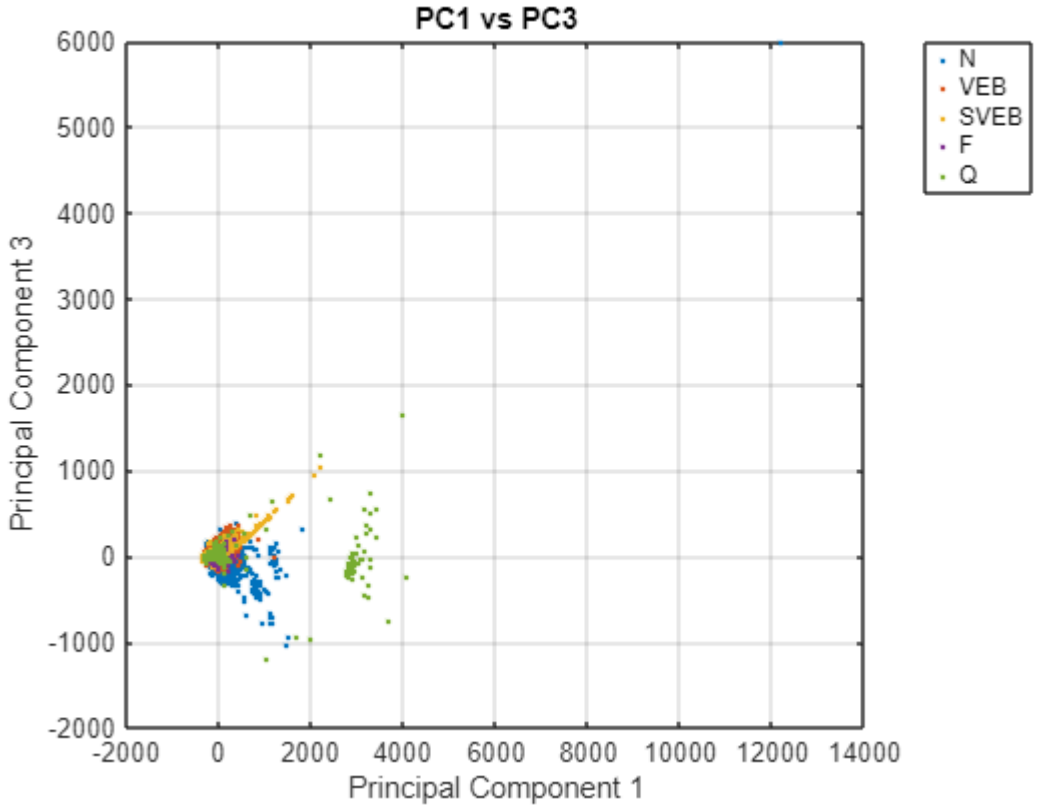
```
%Plotting Pareto Chart
figure;
pareto(explained);
xlabel('Principal Component');
ylabel('Variance Explained (%)');
title('PCA Pareto Chart');
```

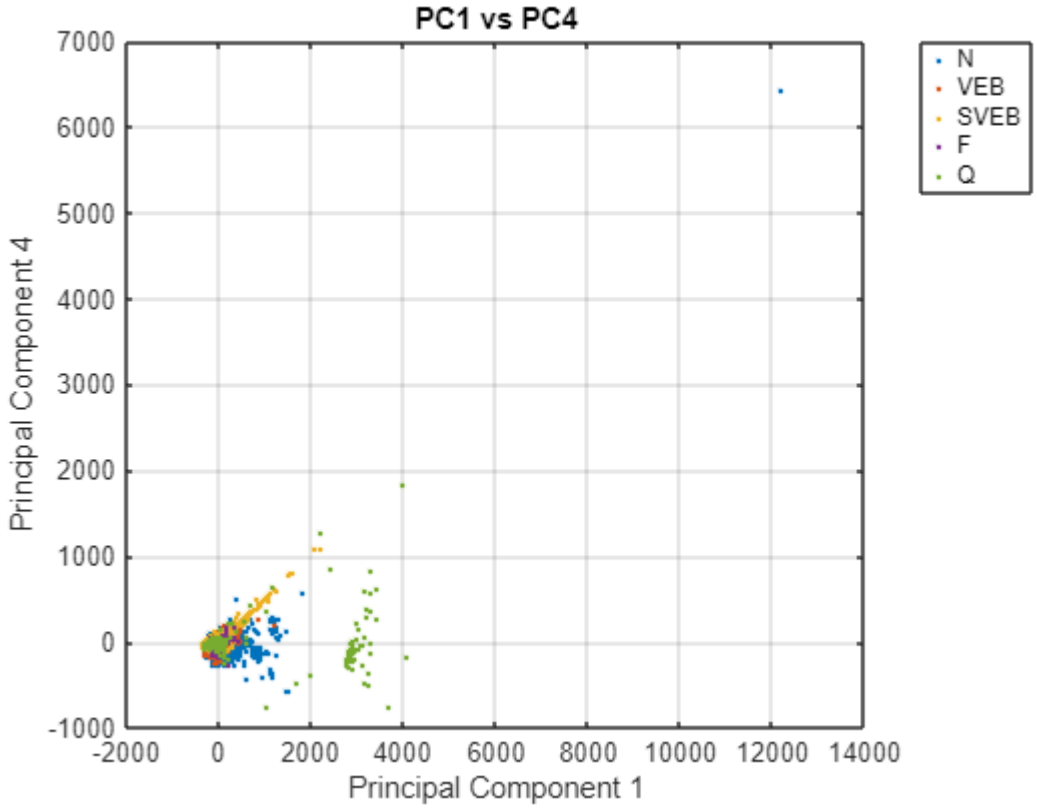


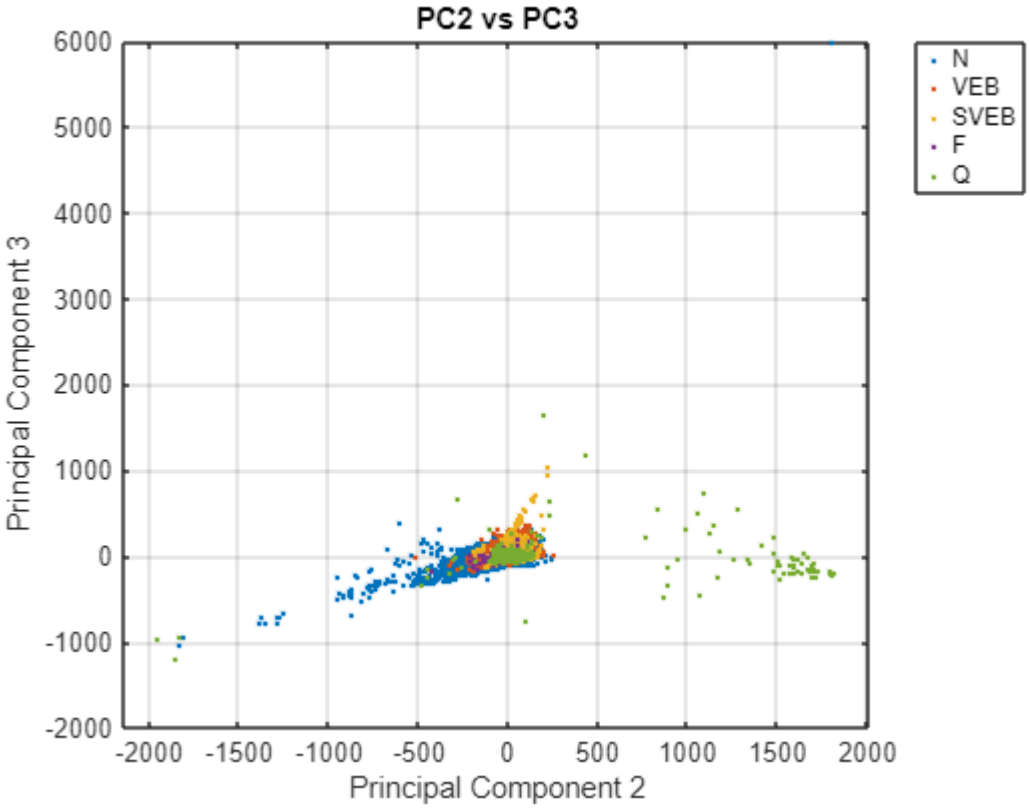

```
% Plot all combinations of the first few principal components (e.g., 4)
numToPlot = 4;  % Adjust this number based on how many PCs you want to explore
```

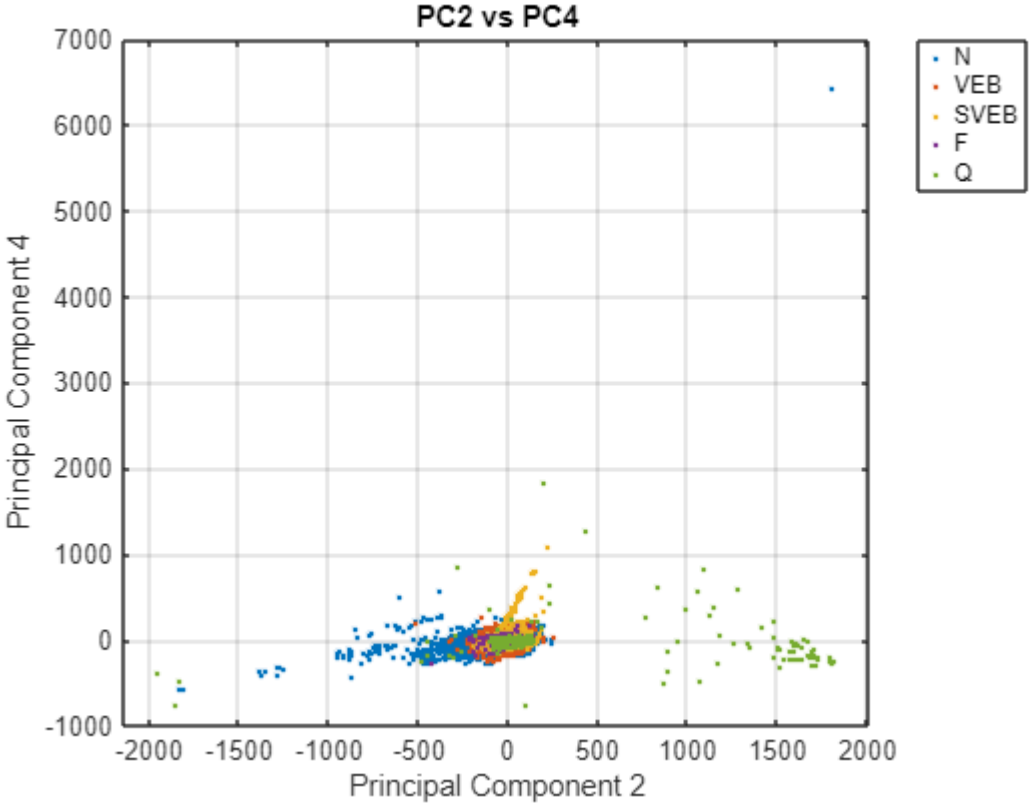
```
for i = 1:numToPlot
    for j = i+1:numToPlot
        figure;
        gscatter(scoreTrain(:,i), scoreTrain(:,j), Y_train);
        xlabel(['Principal Component ' num2str(i)]);
        ylabel(['Principal Component ' num2str(j)]);
        title(['PC' num2str(i) ' vs PC' num2str(j)]);
        legend('Location','bestoutside');
        grid on;
    end
end
```

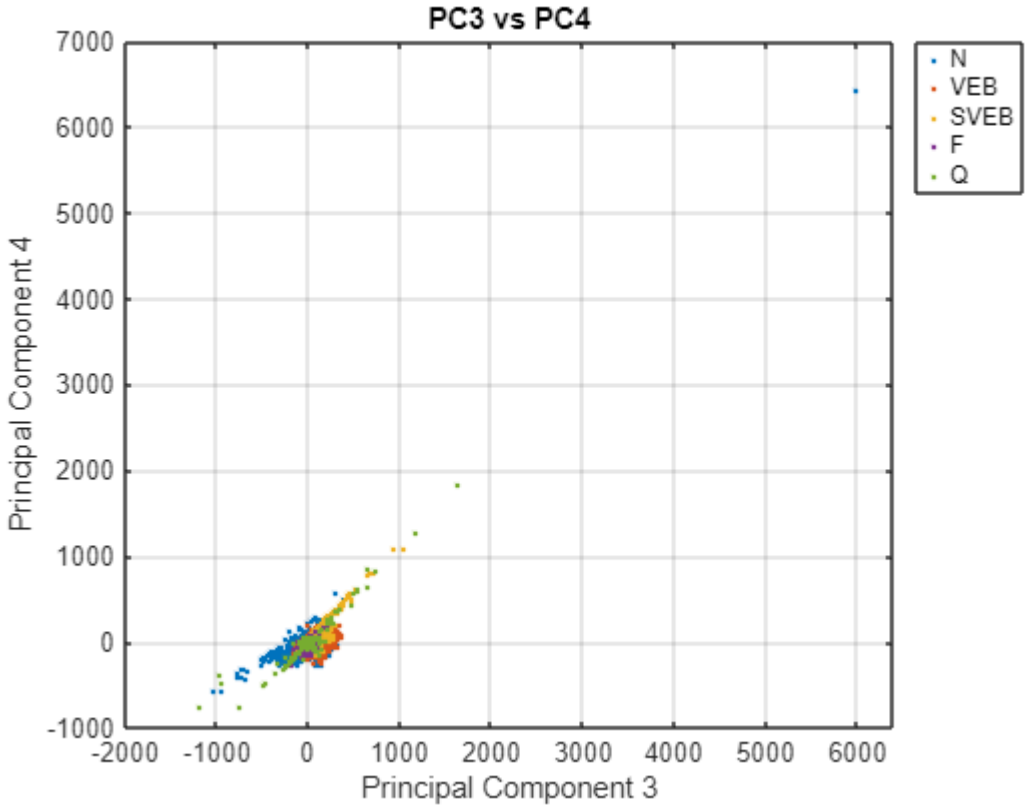



PC1 vs PC3
Principal Component 3
Principal Component 1
6000
5000
4000
3000
2000
1000
0
-1000
-2000
-2000
0
2000
4000
6000
8000
10000
12000
14000
N
VEB
SVEB
F
Q


PC1 vs PC4
Principal Component 4
Principal Component 1
7000
6000
5000
4000
3000
2000
1000
0
-1000
-2000
0
2000
4000
6000
8000
10000
12000
14000
N
VEB
SVEB
F
Q


PC2 vs PC3
Principal Component 3
Principal Component 2
6000
5000
4000
3000
2000
1000
0
-1000
-2000
-2000 -1500 -1000 -500 0 500 1000 1500 2000
N
VEB
SVEB
F
Q


PC2 vs PC4
Principal Component 4
Principal Component 2
7000
6000
5000
4000
3000
2000
1000
0
-1000
-2000 -1500 -1000 -500 0 500 1000 1500 2000
N
VEB
SVEB
F
Q

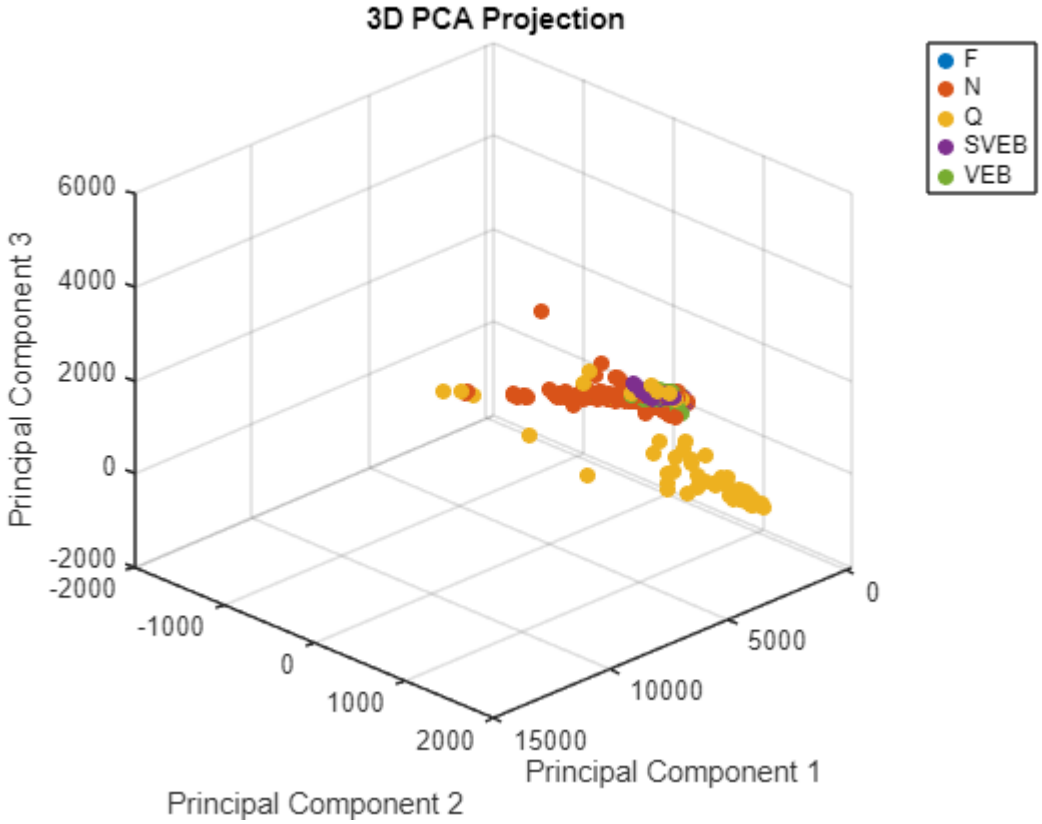



```
% Convert Y_train to string if it's not already
Y_train = string(Y_train);

% Unique class labels as a string array
classes = unique(Y_train);

% Define color map
colors = lines(numel(classes));

% 3D PCA plot
figure;
hold on;
for i = 1:numel(classes)
    idx = Y_train == classes(i);  % use string comparison
    scatter3(scoreTrain(idx,1), scoreTrain(idx,2), scoreTrain(idx,3), ...
             36, colors(i,:), 'filled', 'DisplayName', classes(i));
end
hold off;
xlabel('Principal Component 1');
ylabel('Principal Component 2');
zlabel('Principal Component 3');
title('3D PCA Projection');
legend('Location', 'bestoutside');
grid on;
view(135, 30);
```


3D PCA Projection
F
N
Q
SVEB
VEB
Principal Component 3
6000
4000
2000
0
-2000
Principal Component 2
-2000
-1000
0
1000
2000
Principal Component 1
15000
10000
5000
0