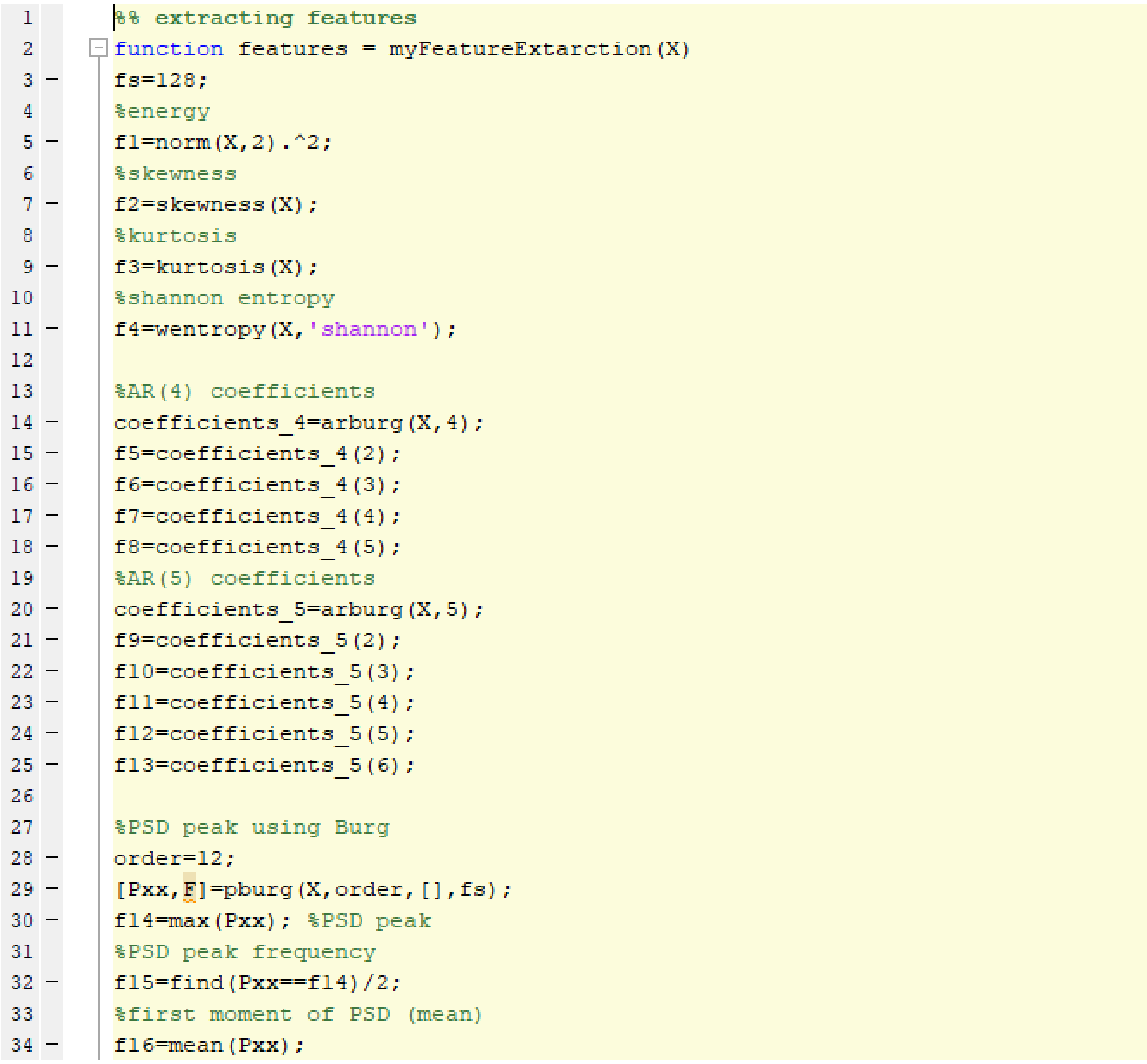
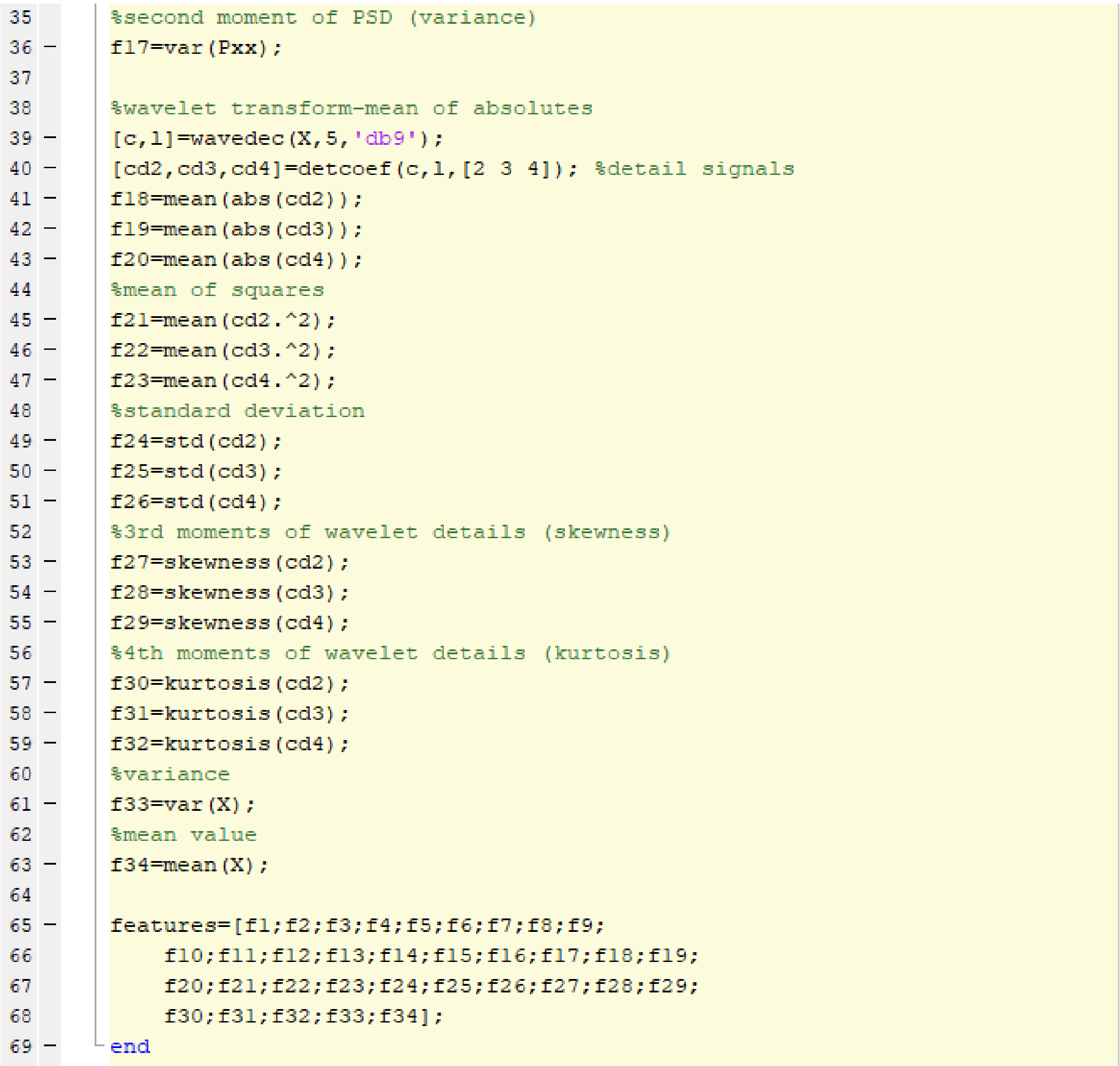
myFeatureExtraction

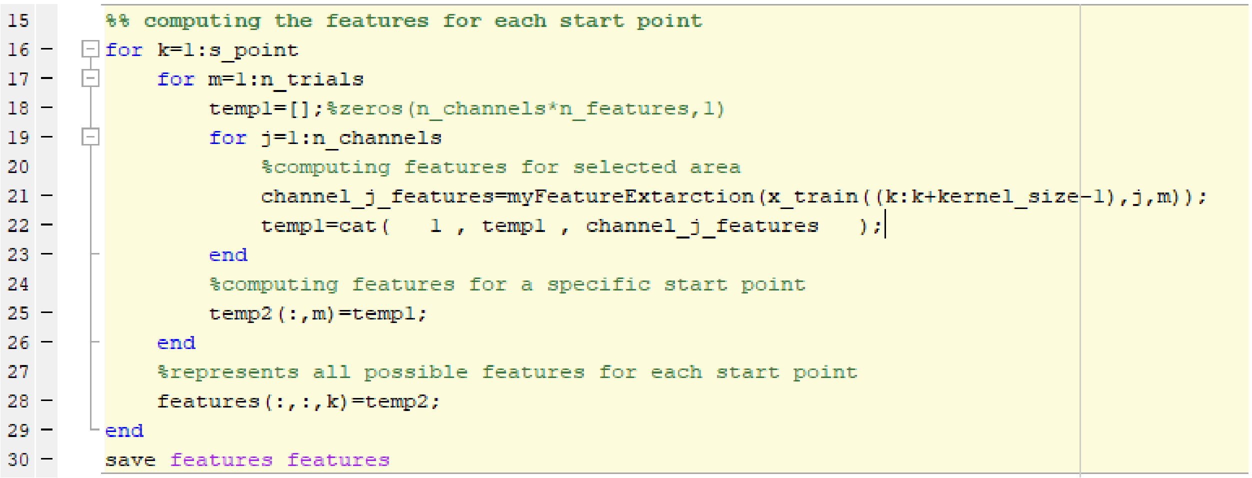


```
%% extracting features
function features = myFeatureExtarction(X)
fs=128;
%energy
f1=norm(X,2).^2;
%skewness
f2=skewness(X);
%kurtosis
f3=kurtosis(X);
%shannon entropy
f4=wentropy(X,'shannon');

%AR(4) coefficients
coefficients_4=arburg(X,4);
f5=coefficients_4(2);
f6=coefficients_4(3);
f7=coefficients_4(4);
f8=coefficients_4(5);
%AR(5) coefficients
coefficients_5=arburg(X,5);
f9=coefficients_5(2);
f10=coefficients_5(3);
f11=coefficients_5(4);
f12=coefficients_5(5);
f13=coefficients_5(6);

%PSD peak using Burg
order=12;
[Pxx,F]=pburg(X,order,[],fs);
f14=max(Pxx); %PSD peak
%PSD peak frequency
f15=find(Pxx==f14)/2;
%first moment of PSD (mean)
f16=mean(Pxx);
```



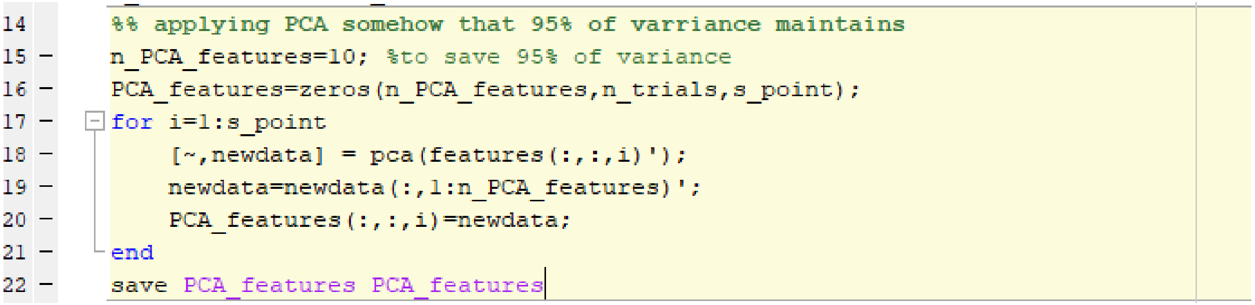
```
%second moment of PSD (variance)
f17=var(Pxx);

%wavelet transform-mean of absolutes
[c,l]=wavedec(X,5,'db9');
[cd2,cd3,cd4]=detcoef(c,l,[2 3 4]); %detail signals
f18=mean(abs(cd2));
f19=mean(abs(cd3));
f20=mean(abs(cd4));
%mean of squares
f21=mean(cd2.^2);
f22=mean(cd3.^2);
f23=mean(cd4.^2);
%standard deviation
f24=std(cd2);
f25=std(cd3);
f26=std(cd4);
%3rd moments of wavelet details (skewness)
f27=skewness(cd2);
f28=skewness(cd3);
f29=skewness(cd4);
%4th moments of wavelet details (kurtosis)
f30=kurtosis(cd2);
f31=kurtosis(cd3);
f32=kurtosis(cd4);
%variance
f33=var(X);
%mean value
f34=mean(X);

features=[f1;f2;f3;f4;f5;f6;f7;f8;f9;
    f10;f11;f12;f13;f14;f15;f16;f17;f18;f19;
    f20;f21;f22;f23;f24;f25;f26;f27;f28;f29;
    f30;f31;f32;f33;f34];
end
```

Saving all features



```
%% computing the features for each start point
for k=1:s_point
    for m=1:n_trials
        temp1=[];%zeros(n_channels*n_features,1)
        for j=1:n_channels
            %computing features for selected area
            channel_j_features=myFeatureExtarction(x_train((k:k+kernel_size-1),j,m));
            temp1=cat(   1 , temp1 , channel_j_features   );
        end
        %computing features for a specific start point
        temp2(:,m)=temp1;
    end
    %represents all possible features for each start point
    features(:,:,k)=temp2;
end
save features features
```

Applying PCA



```
%% applying PCA somehow that 95% of varriance maintains
n_PCA_features=10; %to save 95% of variance
PCA_features=zeros(n_PCA_features,n_trials,s_point);
for i=1:s_point
    [~,newdata] = pca(features(:,:,i)');
    newdata=newdata(:,1:n_PCA_features)';
    PCA_features(:,:,i)=newdata;
end
save PCA_features PCA_features
```