

```
clear all
rng("default")

%load the test data
TestData = csvread('.\Test1.csv',1);
%load mat file
mat = load('.\Final_NB.mat');

%call model from mat file
NBmodel_final = mat.NBmodel_final;

% feature selection using Chi2
TestData = TestData(:, [1,5,2,8,10]);

% Hyperparametes list
dis_list = {'Normal', 'kernel'};
prior = {'empirical', 'uniform'};

%matrixes that have the results of hyperparametet tuning with grid search
hp_perf = mat.hp_perf;
cv_results = mat.cv_results;

% choosing the best set of Hyper parameters

[MaxAccuracy,I] = max(hp_perf(:));
[I_row, I_col] = ind2sub(size(hp_perf),I); %I_row is the row index and I_col is the column index
best_distribution = dis_list(I_row);
best_prior = prior(I_col);

%evaluating the model on test set
tic
predictions = predict(NBmodel_final,TestData(:,1:4));
%predictions = str2num(cell2mat(predictions));
predict_time = toc
% Print the elapsed time
fprintf('Elapsed time: %f seconds\n', predict_time);

%Accruracy
incorrect = predictions == TestData(:,5);
Test_accuracy = sum(incorrect)/numel(predictions);

% Generate the confusion matrix
cm = confusionmat(TestData(:,5),predictions);
```

```
% Extract the values from the confusion matrix
TP = cm(1,1);
TN = cm(2,2);
FP = cm(2,1);
FN = cm(1,2);

% Calculate precision and recall
precision = TP / (TP + FP);
recall = TP / (TP + FN);
f1 = 2 * precision * recall / (precision + recall);
%plot confusion matrix
confusionchart(TestData(:,end),predictions,"Normalization","absolute")
confusionchart(TestData(:,end),predictions,"Normalization","row-normalized")
confusionchart(TestData(:,end),predictions,"Normalization","column-normalized")

% Get the predicted probabilities for the test set
[predictions, scores] = predict(NBmodel_final,TestData(:,1:end-1));
% Convert the predicted labels to a binary vector
% Compute the ROC curve
[fpr, tpr, thr] = perfcurve(TestData(:,end), scores(:,2), 1);
% Compute the AUC value
auc = trapz(fpr,tpr);
% Plot the ROC curve
figure;
plot(fpr,tpr);
xlabel('False Positive Rate');
ylabel('True Positive Rate');
title(sprintf('ROC curve (AUC = %0.2f)', auc));
```