SVM

Classifying the forest fire area as categorical Variable.

Kernel methods	Efficiency
rbfdot	0.718254
Polydot	0.722222
Tanhdot	0.5634921
vanilladot	0.722222
Laplacedot	0.722222
Besseldot	0.722222
Anovadot	0.722222
splinedot	0.6150794

Here the Confusion matrix for vanilladot is:

Actual Predict	Large	Small
Large	0	0
Small	70	182

Confusion matrix for rbfdot is:

Actual Predict	Large	Small
Large	2	3
Small	68	179

Here the efficiency we got with both of the methods have no significant difference or difference is negligible. I am not going to consider the possible reason for miss classification is for the imbalance data here. Maybe it requires more specific variable to come up with a better conclusion.

As we are unable to find out the F1_scores for vanilladot method so we may consider our rbfdot as our final model to work with.

Classifying the Salary as categorical Variable in our data set Salary_data.

NB: It may take more time than expected to run the for loop written in the r code, as I have not performed any kind of normalization or dummy variable technique in my model. You can see the output in the text format in a text file attached in mail.

Kernel methods	Efficiency
rbfdot	0.8540505
Polydot	0.8462815
Tanhdot	0.6638778
vanilladot	0.8462815
Laplacedot	0.852656
Besseldot	0.7703187
Anovadot	0.7826693
splinedot	0.7517928

Confusion Matrix for

rbfdot

Predict Actual	<=50k	>50k
<=50k	10660	700
>50k	1498	2202

F1_Score = 0.9065397

Confusion Matrix for vanilladot

Predict Actual	<=50k	>50k
<=50k	10599	761
>50k	1554	2146

F1_Score = 0.9015438

Confusion Matrix for Laplasedot

Predict Actual	<=50k	>50k
<=50k	10634	724
>50k	1495	2205

F1_Score = 0.9055383

Here as we can see our rbfdot method is is the highest f1 score as well as highest efficiency among all, so Here I may consider my model with kernel method rbfdot as my final model for classification.