> n15c15 \_ ensemble

Call:

summary.resamples(object = results)

Models: svmRadial, rf, J48, AdaBoost.M1

Number of resamples: 25

Accuracy

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

svmRadial 0.8443396 0.8699764 0.8794326 0.8796840 0.8891509 0.9007092 0

rf 0.8773585 0.8888889 0.8983452 0.8992464 0.9080189 0.9267139 0

J48 0.8207547 0.8443396 0.8537736 0.8572880 0.8699764 0.9054374 0

AdaBoost.M1 0.8254717 0.8672986 0.8770686 0.8773271 0.8886256 0.9054374 0

Kappa

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

svmRadial 0.6884852 0.7399135 0.7588720 0.7592860 0.7782525 0.8013951 0

rf 0.7545641 0.7776597 0.7966596 0.7983735 0.8159313 0.8533008 0

J48 0.6413498 0.6885961 0.7076512 0.7145116 0.7399309 0.8108652 0

AdaBoost.M1 0.6506947 0.7343346 0.7540423 0.7545252 0.7770208 0.8106790 0

> dotplot(results)

>

>

> # correlation between results

> modelCor(results)

svmRadial rf J48 AdaBoost.M1

svmRadial 1.0000000 0.6002914 0.4861566 0.6695890

rf 0.6002914 1.0000000 0.5128527 0.6538304

J48 0.4861566 0.5128527 1.0000000 0.5328726

AdaBoost.M1 0.6695890 0.6538304 0.5328726 1.0000000

> splom(results)

>

> #individual prediction of each method on data

> p <- as.data.frame(predict(models, newdata=head(Exdata[,predictors])))

> print(p)

svmRadial rf J48 AdaBoost.M1

1 0.99552607 0.862 0.9723992 0.8977125

2 0.04261104 0.162 0.1176471 0.4119593

3 0.31435041 0.406 0.9319372 0.5202791

4 0.77121364 0.680 0.7500000 0.5299069

5 0.98765460 0.902 0.9723992 0.6599886

6 0.99424021 0.764 1.0000000 0.7425305

>

>

> # stack using random forest

> set.seed(3233)

> stack.rf <- caretStack(models, method="rf", metric="Accuracy", trControl=control)

> print(stack.rf)

A rf ensemble of 2 base models: svmRadial, rf, J48, AdaBoost.M1

Ensemble results:

Random Forest

10580 samples

4 predictor

2 classes: 'neg', 'pos'

No pre-processing

Resampling: Cross-Validated (5 fold, repeated 5 times)

Summary of sample sizes: 8464, 8464, 8464, 8464, 8464, 8464, ...

Resampling results across tuning parameters:

mtry Accuracy Kappa

2 0.8959924 0.7918982

3 0.8956522 0.7912201

4 0.8943856 0.7886847

Accuracy was used to select the optimal model using the largest value.

The final value used for the model was mtry = 2.

>

> # validation

> library("caTools")

> model\_preds <- lapply(models, predict, newdata=Exdata[,predictors], type="prob")

> #model\_preds <- lapply(model\_preds, function(x) x[,"M"])

> model\_preds <- data.frame(model\_preds)

>

> model\_preds$ensemble <- predict(stack.rf,newdata=Exdata[,predictors],type="prob")

> #CF <- coef(stack.rf$ens\_model$finalModel)[,outcomeName]

> CF <- coef(stack.rf$models$svmRadial)[,-1]

> CF

NULL

> colAUC(model\_preds, Exdata[,outcomeName])

svmRadial.neg svmRadial.pos rf.neg rf.pos J48.neg J48.pos

neg vs. pos 0.9398085 0.9398085 0.9503288 0.9503288 0.856711 0.856711

AdaBoost.M1.neg AdaBoost.M1.pos ensemble

neg vs. pos 0.9350629 0.9350629 0.9294025

> CF

NULL

>

>

> colAUC(model\_preds$ensemble,Exdata[,outcomeName])

[,1]

neg vs. pos 0.9294025

>

>

> #my prediction trial

> p\_ens <- as.data.frame(predict(stack.rf, newdata=head(Exdata[,predictors])))

> print(p\_ens)

predict(stack.rf, newdata = head(Exdata[, predictors]))

1 neg

2 pos

3 pos

4 neg

5 neg

6 neg