ENSEMBLE LEARNING : STACKING / BLENDING

Deepanshu Bhalla 6 Comments
Machine Learning, R, R Programming

Stacking (aka Blending)

Stacking is a form of ensemble learning which aims to improve accuracy by combining predictions from several learning algorithms.

Ensemble Learning : Stacking

Step I : Multiple different algorithms are trained using the available data. For example, Boosting Trees and Single Decision Tree were trained for a data set. These are the two classifiers.

Step II : Calculate Predicted Probabilities of these multiple different algorithms

Step III : Combine dependent variable and two columns of the above predicted probabilities of different multiple algorithms.

Step IV : Run Logistic Regression on data set prepared in step III. In this ensemble process, logistic regression is considered as a meta classifier.

Step V : Capture two coefficients (ignoring intercept) derived from logistic regression.

Step VI : Calculate linear weights based on the coefficients.

> ***Weight I :*** *CoefficientI / Sum (CoefficientI + CoefficientII)*
> ***Weight II :*** *CoefficientII / Sum (CoefficientI + CoefficientII)*

Step VII : Calculate **Ensemble Learning Prediction Probability Score** by multiplying weights with predicted scores.

> ***Ensemble Learning =*** *W1 * P1 + W2 * P2*

W1 : Weight of First Algorithm, **W2 :** Weight of Second Algorithm, **P1 :** Predicted Probability of First Algorithm, **P2 :** Predicted Probability of Second Algorithm

How to know individual models suitable for an ensemble

writing, statisti...

- **Excel : Intersection of two linear straight lines**
  To find intersection of two straight lines: First we need the equations of the two lines. Then, since at the point of intersection, the...
- **Sample Size Calculator with Excel**
  Determining sample size is a very important issue because samples that are too large may waste time, resources and money, while samples tha...
- **Importing Excel Data into SAS**
  PROC IMPORT is the SAS procedure used to read data from excel into SAS. Syntax: PROC IMPORT DATAFILE="filename" OU...
- **Excel : Intersection between curve and straight line**
  To find intersection of curve and a straight line we first need to know the mathematical condition behind it. When two lines cross...
- **Two ways to increment formula row when copied across columns in Excel**
  Scenario Suppose you are asked to calculate cumulative sale. And the figure should be displayed in columns. Hence the formula should incr...
- **Creating Infographics with Powerpoint : Free Templates**
  Infographics An infographic (information graphic) is a representation of information in a graphic format designed to make the data e...
- **List of free softwares for econometrics**
  1. Gretl It's a cross-platform software package for econometric analysis, written in the C programming language. 2. FreeMat I...

> *The individual models make a good candidate for an ensemble if their predicitons are fairly un-correlated, but their overall accuracy is similar.*

Can we use Boosting/Bagging Trees instead of Logistic Regression for an ensemble?

Yes, we can, They use more sophisticated ensembles than simple linear weights, but these models are much more susceptible to over-fitting.

We should use Trees instead of Logistic Regression for an ensemble when we have :

1. Lots of data
2. Lots of models with similar accuracy scores
3. Your models are uncorrelated

Alternative Technique : Ensemble with Linear Greedy Optimization

R Code : Ensemble Learning - Stacking

```
# Loading Required Packages
library(caret)
library(caTools)
library(RCurl)
library(caretEnsemble)
library(pROC)

# Reading data file
urlfile <-
'https://raw.githubusercontent.com/hadley/fueleconomy
```

IMPORTANT LINKS

- 4 Simple VBA Lessons
- Actuarial Science
- Advanced Excel
- Business Analytics
- Charts
- Decision Tree
- Excel
- Excel Macros
- Functions
- Infographics
- Linear Regression
- Machine Learning
- Mathematics Using Excel
- Outlier
- Powerpoint
- R
- R Programming
- random forest
- Resumes
- SAS
- SAS Base Certification Questions and Answers
- SAS For Beginners
- SAS Interview Questions
- SPSS
- SQL
- Statistics
- Statistics Using Excel
- Text Analytics
- Text Mining
- Time Series
- Time Series Forecasting
- VBA
- Web Analytics

```
/master/data-raw/vehicles.csv'
x <- getURL(urlfile, ssl.verifypeer = FALSE)
vehicles <- read.csv(textConnection(x))

# Cleaning up the data and only use the first 24
columns
vehicles <- vehicles[names(vehicles)[1:24]]
vehicles <- data.frame(lapply(vehicles, as.character),
stringsAsFactors=FALSE)
vehicles <- data.frame(lapply(vehicles, as.numeric))
vehicles[is.na(vehicles)] <- 0
vehicles$cylinders <- ifelse(vehicles$cylinders == 6,
1,0)

# Making dependent variable factor and label values
vehicles$cylinders <- as.factor(vehicles$cylinders)
vehicles$cylinders <- factor(vehicles$cylinders,
                  levels = c(0,1),
                  labels = c("level1", "level2"))

# Split data into two sets - Training and Testing
set.seed(107)
inTrain <- createDataPartition(y = vehicles$cylinders, p
= .7, list = FALSE)
training <- vehicles[ inTrain,]
testing <- vehicles[-inTrain,]

# Setting Control
ctrl <- trainControl(
  method='cv',
  number= 3,
  savePredictions=TRUE,
  classProbs=TRUE,
  index=createResample(training$cylinders, 10),
  summaryFunction=twoClassSummary
)
```

```
# Train Models
model_list <- caretList(
  cylinders~., data=training,
  trControl = ctrl,
  metric='ROC',
  tuneList=list(
  rf1=caretModelSpec(method='rpart', tuneLength =
10),
  gbm1=caretModelSpec(method='gbm', distribution =
"bernoulli",
                      bag.fraction = 0.5,
tuneGrid=data.frame(n.trees = 50,
                                              interaction.depth =
2,
                                              shrinkage = 0.1,
                                              n.minobsinnode =
10))
  )
)

# Check AUC of Individual Models
model_list$rf1
model_list$gbm1

#Check the 2 model's correlation
#Good candidate for an ensemble: their predicitons are
fairly un-correlated,
#but their overall accuaracy is similar
modelCor(resamples(model_list))


###########################################
####################
# Technique I : Linear Greedy Optimization on AUC
```

```
############################################
#####################

greedy_ensemble <- caretEnsemble(model_list)

#Check AUC Scores on individual and ensemble
models
summary(greedy_ensemble)

############################################
################
# Validation on Testing Sample
############################################
################

ens_preds <- predict(greedy_ensemble,
newdata=testing)

#Preparing dataset for Pred. Probabilities of both
individual and ensemble models
model_preds <- lapply(model_list, predict,
newdata=testing, type='prob')
model_preds <- lapply(model_preds, function(x)
x[,'level2'])
model_preds <- data.frame(model_preds)
model_preds$ensemble <- ens_preds

#Calculate AUC for both individual and ensemble
models
colAUC(model_preds, testing$cylinders)


############################################
#####################
# Technique II : Stacking / Blending
############################################
```

```
#####################

glm_ensemble <- caretStack(
  model_list,
  method='glm',
  metric='ROC',
  trControl=trainControl(
    method='cv',
    number=3,
    savePredictions=TRUE,
    classProbs=TRUE,
    summaryFunction=twoClassSummary
  )
)

# Check Results
glm_ensemble


############################################
############
# Validation on Testing Sample
############################################
############

model_preds2 <- model_preds
model_preds2$ensemble <- predict(glm_ensemble,
newdata=testing, type='prob')$level2
CF <- coef(glm_ensemble$ens_model$finalModel)[-1]
colAUC(model_preds2, testing$cylinders)

#Checking Weights
CF/sum(CF)
```

It’s Your Turn!

If you want me to keep writing this site, please post your feedback in the comment box below. While I love having friends who agree, I only learn from those who don't!

RELATED POSTS:

- Predict Functions in R
- Weighting in Conditional Tree and SVM
- R : Apply Function on Rows
- Split a data frame
- R : Convert Data from Wide to Long Format
- Validate Cluster Analysis
- Cluster Analysis with R
- Ensemble Learning : Stacking / Blending
- GBM (Boosted Models) Tuning Parameters
- Dimensionality Reduction with R

6 RESPONSES TO "ENSEMBLE LEARNING : STACKING / BLENDING"

cosmos 14 September 2015 at 01:52

Hi...
you need to remove " n.minobsinnode = 10" from tuneList=list(....)

It was a great help in understanding the blending.

Thanks

Reply

Replies

Deepanshu Bhalla 14 September

at 02:39

Why should i remove - n.minobsinnode = 10? It is one of the tuning parameter of GBM.

cosmos 21 September 2015 at 23:42

This comment has been removed by the author.

Deepanshu Bhalla 21 September 2015 at 23:50

It works in the latest version of caret. Check out this link http://topepo.github.io/caret/training.html

cosmos 21 September 2015 at 23:51

ok.
I was using the older version. By the way, very good post.

Deepanshu Bhalla 21 September 2015 at 23:54

Cool. Glad you found it useful.

Reply

Add comment

Enter your comment...

Comment as: Google Accou ▼

Publish Preview

← PREV NEXT →