Machine Learning for Business

Module 10: Random forest

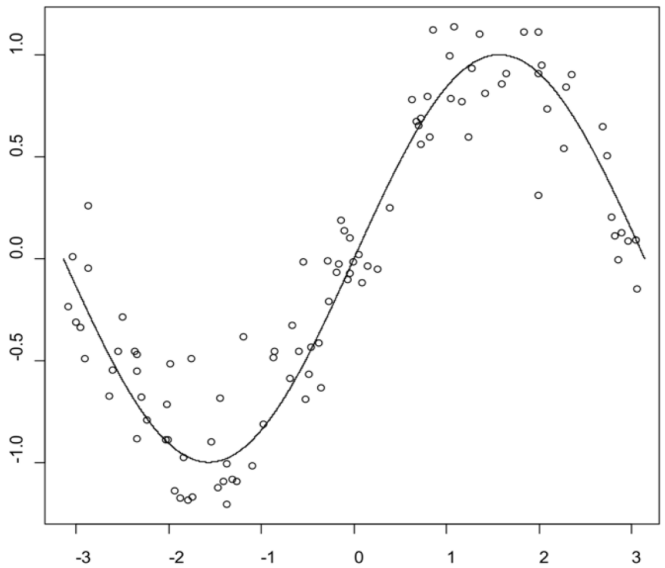
Day 5, 13.00 – 16.00

Asst. Prof. Dr. Santitham Prom-on

Department of Computer Engineering, Faculty of Engineering
King Mongkut's University of Technology Thonburi

BIG DATA EXPERIENCE CENTER

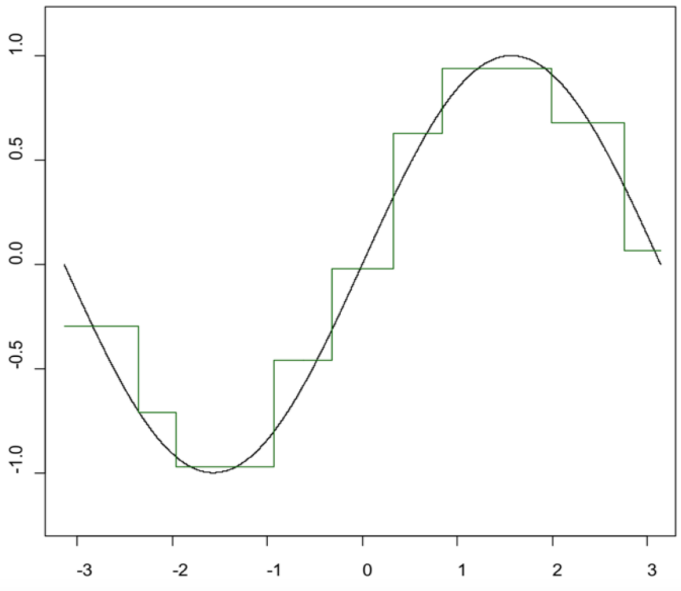
Data and underlying function



BIG DATA EXPERIENCE

g·able

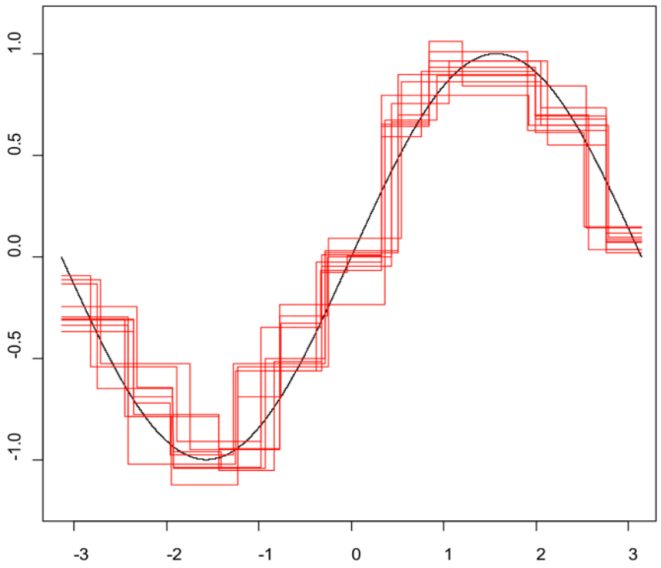
Single regression tree



BIG DATA EXPERIENCE

g·able

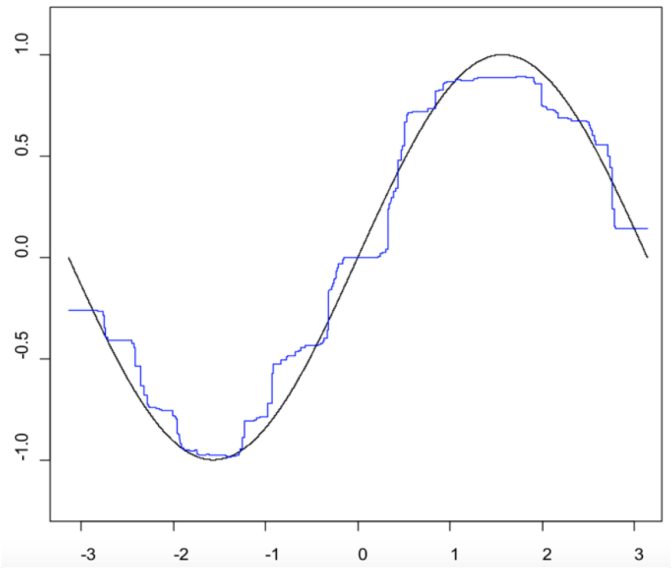
10 Regression Trees



BIG DATA EXPERIENCE

g·able

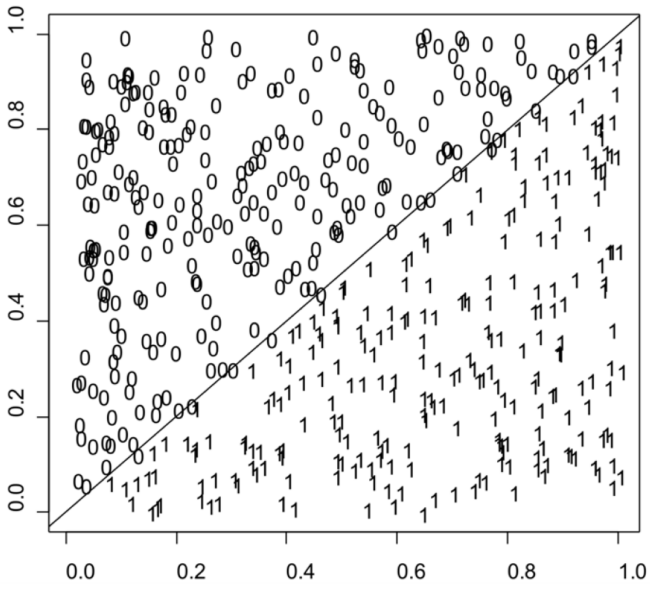
Average of 100 Regression Trees



BIG DATA EXPERIENCE

g·able

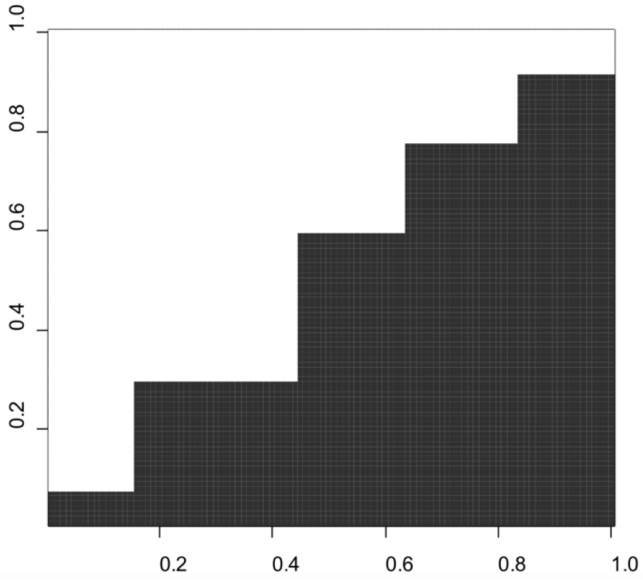
Hard problem for a single tree



BIG DATA EXPERIENCE

g·able

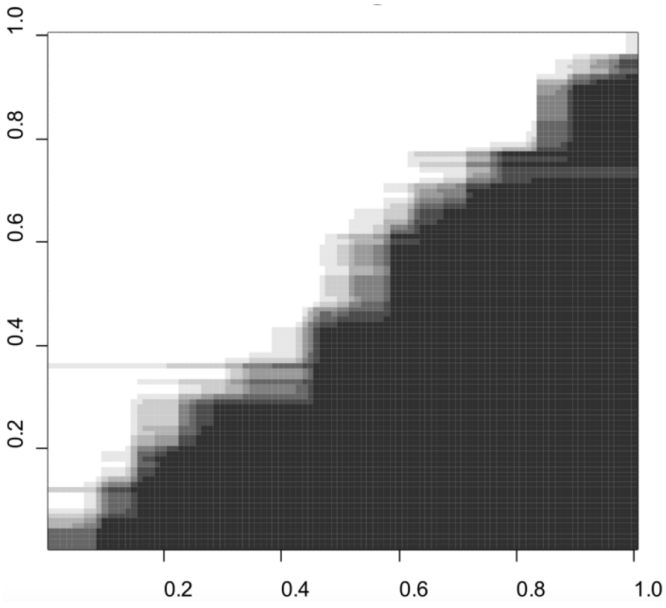
Single tree



BIG DATA EXPERIENCE

g·able

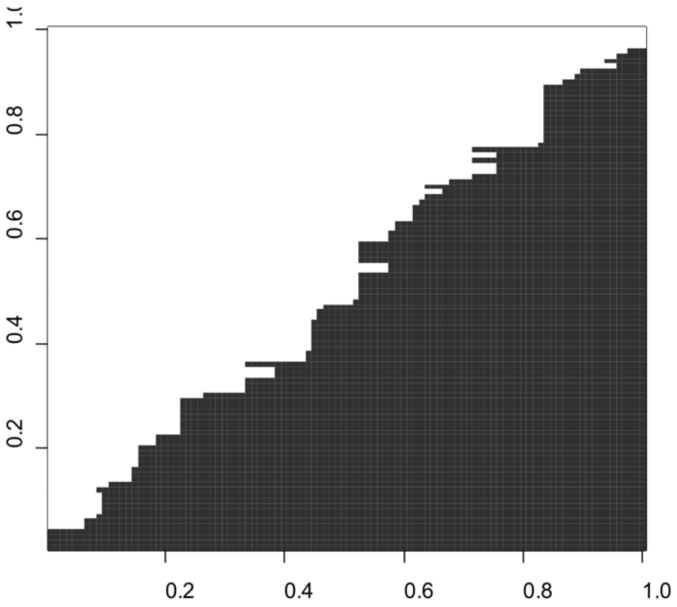
25 Averaged Trees



BIG DATA EXPERIENCE

g·able

25 Voted Trees



BIG DATA EXPERIENCE

g·able

Bagging (Bootstrap Aggregating)

- Fit classification or regression models to bootstrap samples from the data and combine by voting (classification) or averaging (regression).

Bootstrap sample $\Rightarrow$ $f_1(x)$
Bootstrap sample $\Rightarrow$ $f_2(x)$
Bootstrap sample $\Rightarrow$ $f_3(x)$
...
Bootstrap sample $\Rightarrow$ $f_M(x)$

MODEL AVERAGING

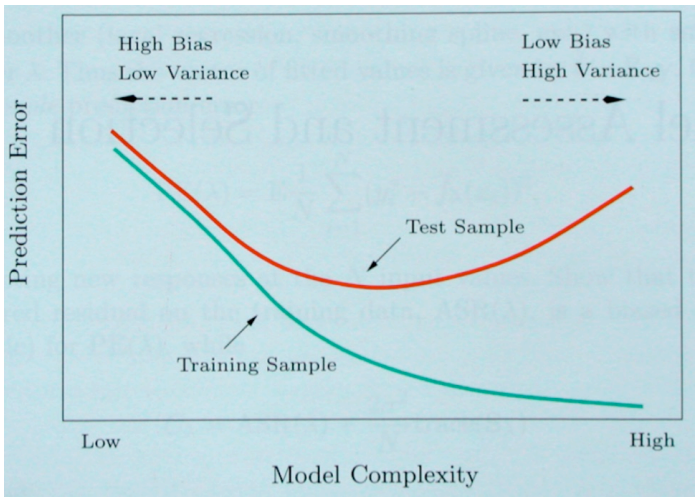
Combine $f_1(x),\ldots, f_M(x) \Rightarrow f(x)$

$f_i(x)$'s are "base learners"

Bagging (Bootstrap Aggregating)

- A bootstrap sample is chosen at random *with* replacement from the data.
- Some observations end up in the bootstrap sample more than once, while others are not included ("out of bag").
- Bagging reduces the *variance* of the base learner but has limited effect on the *bias*.
- It is most effective if we use *strong* base learners that have very little bias but high variance (unstable). E.g. trees.

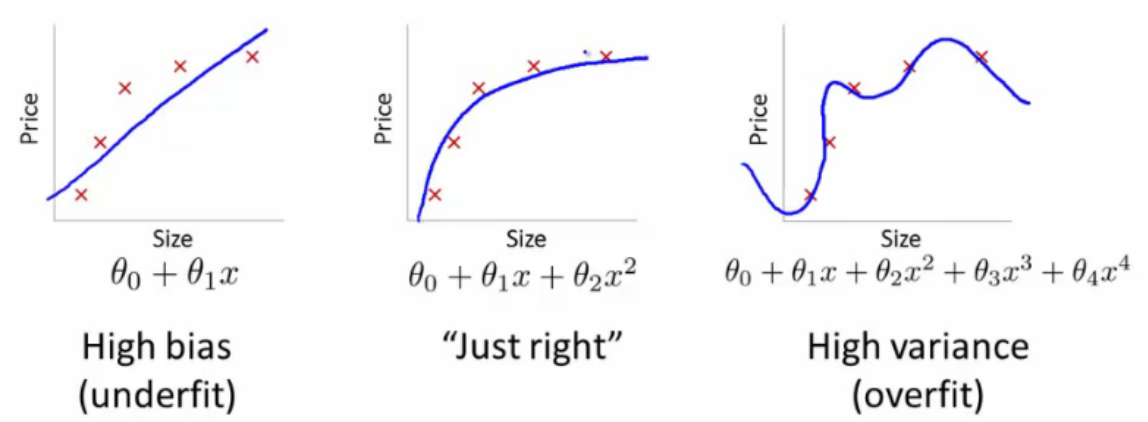
Bias/variance tradeoff



Hastie, Tibshirani, Friedman "Elements of Statistical Learning" 2001

BIG DATA EXPERIENCE

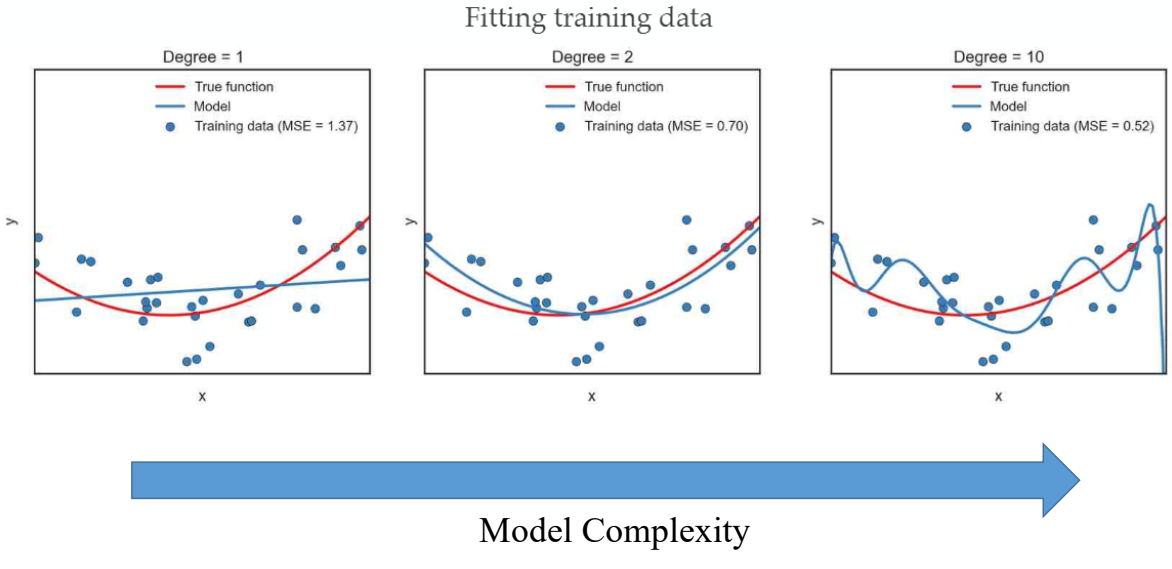
Example: bias vs variance



g·able

BIG DATA EXPERIENCE

Example: bias vs variance



g·able

BIG DATA EXPERIENCE

Reducing variance without increasing bias

- Averaging reduces variance:

$$\mathrm{var}\left(\bar{X}\right) = \frac{\mathrm{var}\left(X\right)}{n}$$

Average models to reduce model variance

One problem

- Only one training set
- Where do multiple models come from?

g·able

BIG DATA EXPERIENCE

Bagging: bootstrap aggregation

- Take repeated bootstrap samples from training set D.
- Bootstrap sampling: Given set D containing N training examples, create D' by drawing N examples at random with replacement from D.

- Bagging:
  - Create k bootstrap samples D_1 … D_k
  - Train distinct classifier on each D_i
  - Classify new instance by majority vote / average

g·able

BIG DATA EXPERIENCE

Random forest algorithm

Grow a **forest** of many trees. (R default is 500)

Grow each tree on an independent **bootstrap sample*** from the training data.

At each node:

1. Select ***m*** variables **at random** out of all ***M*** possible variables (independently for each node).
2. Find the best split on the selected ***m*** variables.

Grow the trees to maximum depth (classification).

Vote/average the trees to get predictions for new data.

*Sample N cases at random with replacement.

g·able

BIG DATA EXPERIENCE

RF Predictor

- A case in the training data is *not* in the bootstrap sample for about one third of the trees (we say the case is "out of bag" or "oob"). Vote (or average) the predictions of *these trees* to give ***the RF predictor.***
- The ***oob error rate*** is the error rate of the ***RF predictor.***
- The ***oob confusion matrix*** is obtained from the ***RF predictor.***
- For new cases, vote (or average) *all* the trees to get the ***RF predictor.***

g·able

BIG DATA EXPERIENCE

Advantages

- Accuracy – Random forest is competitive with the best known machine learning methods
- Stability – If we change the data little, the individual trees may change but the forest is relatively stable it is combination of many trees

g·able

BIG DATA EXPERIENCE

RF Predictor

For example, suppose we fit 1000 trees, and a case is out-of-bag in 339 of them, of which:

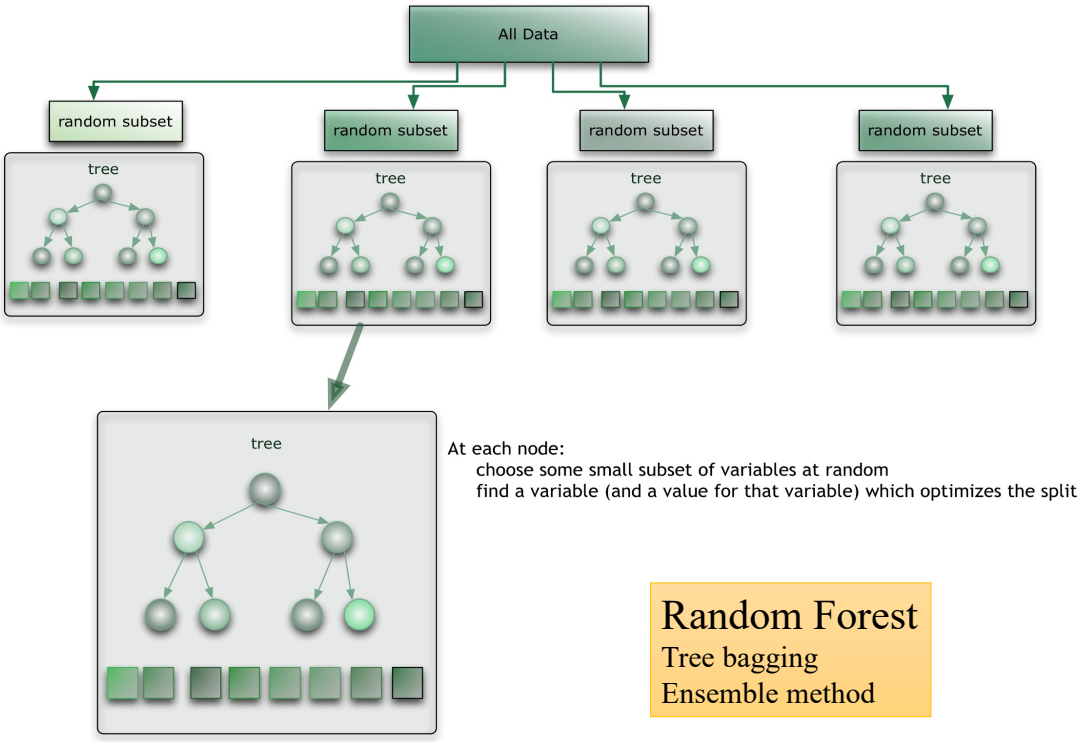
283 say "class 1"

56 say "class 2"

The RF predictor for this case is class 1.

The "oob" error gives an estimate of test set error (generalization error) as trees are added to the ensemble.

g·able



BIG DATA EXPERIENCE

R: Separate training and test data (Again)

```
bankData.train <- sample_frac(bankData, 0.6)
training.ind <- as.integer(rownames(bankData.train))
bankData.test <- bankData[-training.ind,]
```

g·able

BIG DATA EXPERIENCE

R: Build a random forest

```
library(randomForest)
rf <- randomForest(y ~ ., data = bankData.train)
rf
```

```
Call:
 randomForest(formula = y ~ ., data = bankData.train)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 4

        OOB estimate of  error rate: 9.15%
Confusion matrix:
       no  yes class.error
no  23135  845   0.0352377
yes  1638 1509   0.5204957
```

g·able

BIG DATA EXPERIENCE

Prediction

```
res <- predict(rf, bankData.test)
confusionMatrix(res,
                bankData.test$y,
                positive = "yes",
                mode  = "prec_recall")
```

g·able

BIG DATA EXPERIENCE

g·able

```
          Reference
Prediction    no   yes
       no  15294  1031
       yes   648  1111

               Accuracy : 0.9072
                 95% CI : (0.9028, 0.9113)
    No Information Rate : 0.8816
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5181
 Mcnemar's Test P-Value : < 2.2e-16

              Precision : 0.63161
                 Recall : 0.51867
                     F1 : 0.56960
```

BIG DATA EXPERIENCE

g·able

Activity: Decision tree and random forest

- Use 3-credit-approval data
- Separate data into 60:40 for training:testing
- Build a decision tree and a random forest
- Evaluate the model accuracy

BIG DATA
EXPERIENCE

Thank you

Question?

g·able