PSS718 - Data Mining

Lecture 9 - Random Forests

Asst.Prof.Dr. Burkay Genç

Hacettepe University

November 20, 2016

Introduction

- A single decision tree is often too simple
- Many models working together is better than a single model
- Consider a panel of experts to make a decision
- Usually some variables are indistinguishable
- Example: 75% Yes, 25% No ([75, 25])
  - Splitting by A : [80, 20] - [40, 60]
  - Splitting by B : [80, 20] - [40, 60]
  - Idea: build both then merge into an ensemble

Stability

- The decision tree model is instable
  - Remove a few items and the model can change marginally
- Random forest models are much more stable
- They can tolerate noise and change

Underrepresentation

- Random forests handle underrepresented classification tasks quite well
- This is where, in the binary classification task, one class has very few (e.g., 5% or fewer) observations compared with the other class

Randomness

- Variables are selected randomly
- This provide robustness
- Also each tree is built much faster
  - Much less needs to be done at each split

Variable vs Observation

- Random forests are very useful when there are many variables and and not that many observations

Decision Tree

- The main building block is the decision tree (usually)
- However, any other model may also be used
- Random Forest model is a meta-algorithm

How does it work?

- Build many random DTs
- Treat them equally
- If majority says *yes* then the decision is a *yes*, otherwise it is a *no*
- For regression, compute the average

Bagging

- RF algorithm uses the concept of bagging
- BAG: bootstrap aggregation
- Randomly create a sample of all observations
  - Sample size is about two thirds of the original dataset
  - Replacements are allowed
  - An observation may appear more than once in a sample
- Use each sample to train a different DT

Sampling the variables

- At each split, we randomize the variable selection:
  - Choose a small set of variables arbitrarily
  - Select the best variable among these
- At each split (node), do this with a different subset of variables
- How does this affect speed?

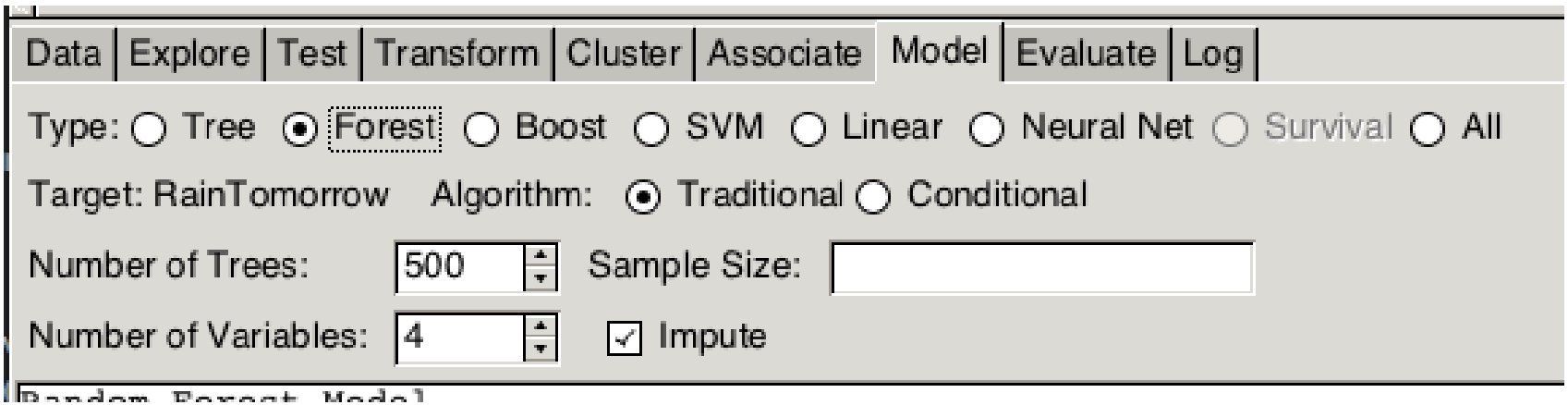
Randomness

- Randomness allows to purposefully create different trees
- Different trees represent different experts
- Allow each tree to overfit, they only contribute a small amount to the overall decision

Ensemble Scoring

- Do not favor any DTs over others
- They are all equally valuable perspectives on the same data
- If the majority of different perspectives agrees on a decision, then it is likely to be the correct decision

User interface



Examining the output

```
Summary of the Random Forest Model
==================================

Number of observations used to build the model: 256
Missing value imputation is active.
```

Missing value imputation

If imputation is not checked then sample size may be lower, as the default behavior is to drop the observations with missing values.

Performance evaluation

```
                Type of random forest: classification
                      Number of trees: 500
No. of variables tried at each split: 4

        OOB estimate of  error rate: 13.28%
Confusion matrix:
     No Yes class.error
No  207   8   0.0372093
Yes  26  15   0.6341463
```

OOB

OOB is out-of-bag. It means the error is computed using the observations left out of the bag for each tree.

Underrepresented Classes

- Notice the high amount of false-negatives in the previous table
- That means you won't be prepared for the rain tomorrow -> Problem!
- RF can fix this by balancing the underrepresented and overrepresented classes
  - RainTomorrow=='Yes' is underrep. with 66/366 observations
  - RainTomorrow=='No' is overrep. with 300/366 observations

Sample Size

```
        OOB estimate of  error rate: 28.52%
Confusion matrix:
     No Yes class.error
No  148  67   0.3116279
Yes   6  35   0.1463415
```

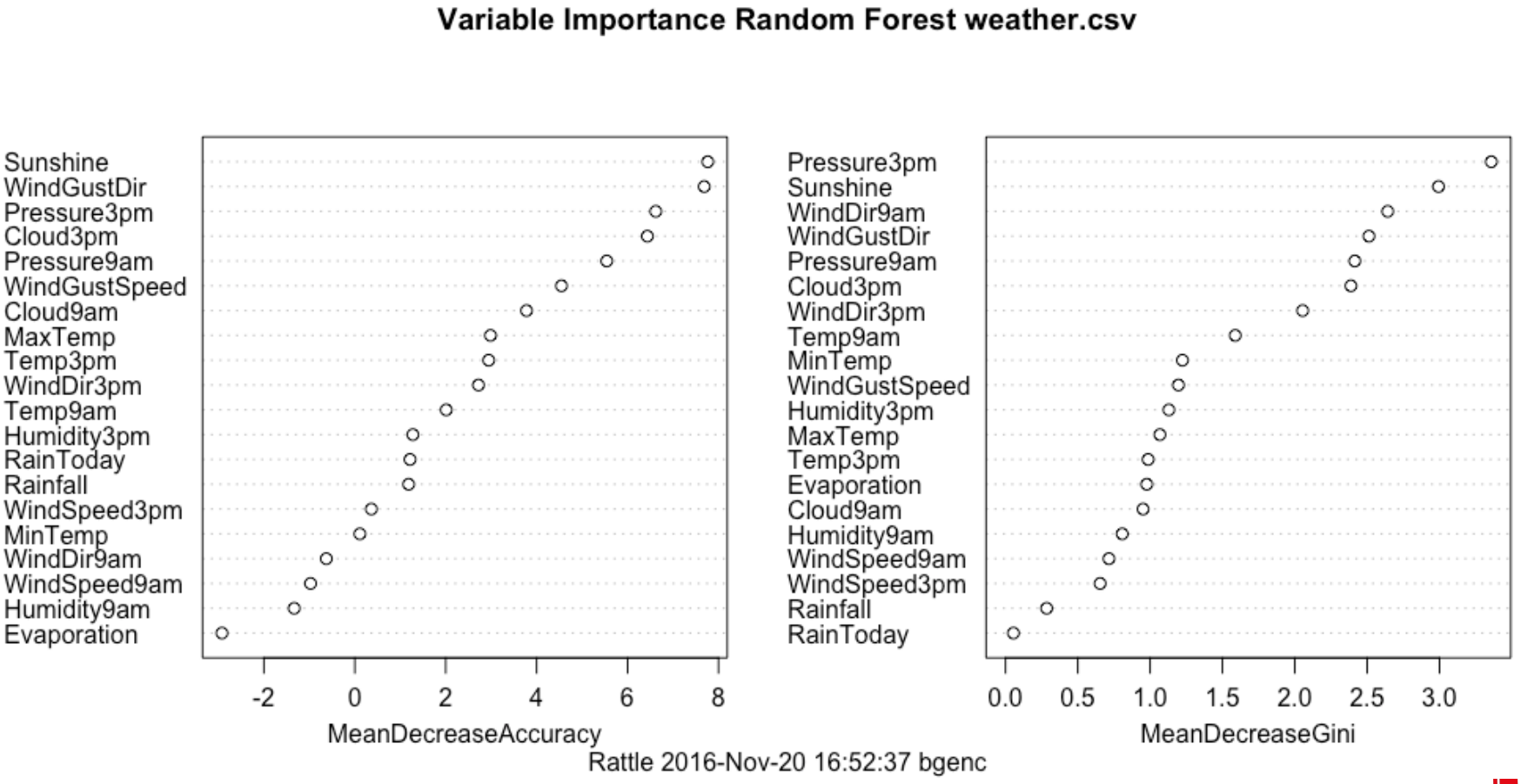
- We sacrificed false-positives in favor of false-negatives
- We will carry an umbrella when there is no need, but we will be caught unprepared less frequently

Variable Importance

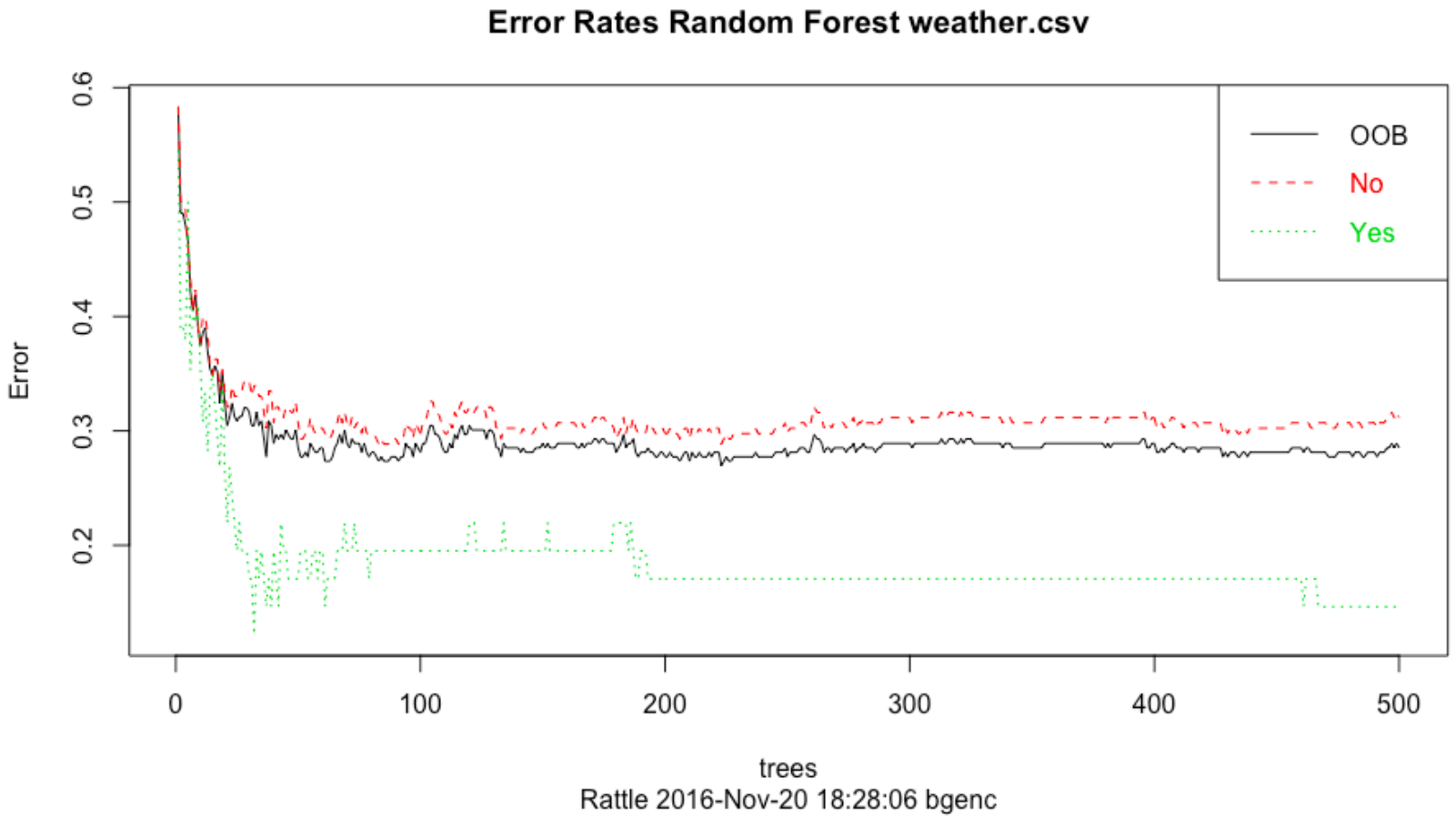
```
Variable Importance
===================

                 No   Yes MeanDecreaseAccuracy MeanDecreaseGini
Sunshine       6.28  7.92                 7.77             2.99
WindGustDir    7.72  1.48                 7.69             2.51
Pressure3pm    4.13 10.10                 6.62             3.36
Cloud3pm       4.67  8.79                 6.44             2.39
Pressure9am    4.20  5.52                 5.54             2.41
WindGustSpeed  4.03  3.38                 4.55             1.20
Cloud9am       3.42  1.94                 3.78             0.95
MaxTemp        2.45  2.77                 2.98             1.07
Temp3pm        2.46  3.14                 2.95             0.99
WindDir3pm     2.89 -0.19                 2.72             2.05
Temp9am        1.21  5.62                 2.01             1.59
Humidity3pm    0.82  1.98                 1.28             1.13
RainToday      1.41 -0.58                 1.21             0.05
Rainfall       1.51 -1.30                 1.18             0.28
WindSpeed3pm   0.24  0.83                 0.36             0.65
MinTemp       -0.16  2.07                 0.11             1.22
WindDir9am    -0.63 -0.10                -0.63             2.64
WindSpeed9am  -1.14  0.95                -0.97             0.72
Humidity9am   -1.29 -0.10                -1.33             0.81
Evaporation   -2.79 -0.91                -2.92             0.98
```

Importance


Variable Importance Random Forest weather.csv
Sunshine
WindGustDir
Pressure3pm
Cloud3pm
Pressure9am
WindGustSpeed
Cloud9am
MaxTemp
Temp3pm
WindDir3pm
Temp9am
Humidity3pm
RainToday
Rainfall
WindSpeed3pm
MinTemp
WindDir9am
WindSpeed9am
Humidity9am
Evaporation
-2
0
2
4
6
8
MeanDecreaseAccuracy
Pressure3pm
Sunshine
WindDir9am
WindGustDir
Pressure9am
Cloud3pm
WindDir3pm
Temp9am
MinTemp
WindGustSpeed
Humidity3pm
MaxTemp
Temp3pm
Evaporation
Cloud9am
Humidity9am
WindSpeed9am
WindSpeed3pm
Rainfall
RainToday
0.0
0.5
1.0
1.5
2.0
2.5
3.0
MeanDecreaseGini
Rattle 2016-Nov-20 16:52:37 bgenc

Error rates


Error Rates Random Forest weather.csv
Error
0.2
0.3
0.4
0.5
0.6
0
100
200
300
400
500
trees
OOB
No
Yes
Rattle 2016-Nov-20 18:28:06 bgenc

Conversion to rules

- You can convert each tree into the corresponding rule set

- Provide the index of the tree for a specific tree
- If you provide 0, then you get the rules of all trees
  - Be careful though, this can easily get out of control

The R command

```
model <- randomForest(formula(df$RainTomorrow ~ .),
                      data = df,
                      ntree = 100,
                      mtry = 4,
                      importance = TRUE,
                      localImp = TRUE,
                      na.action = na.roughfix,
                      replace = FALSE)
```