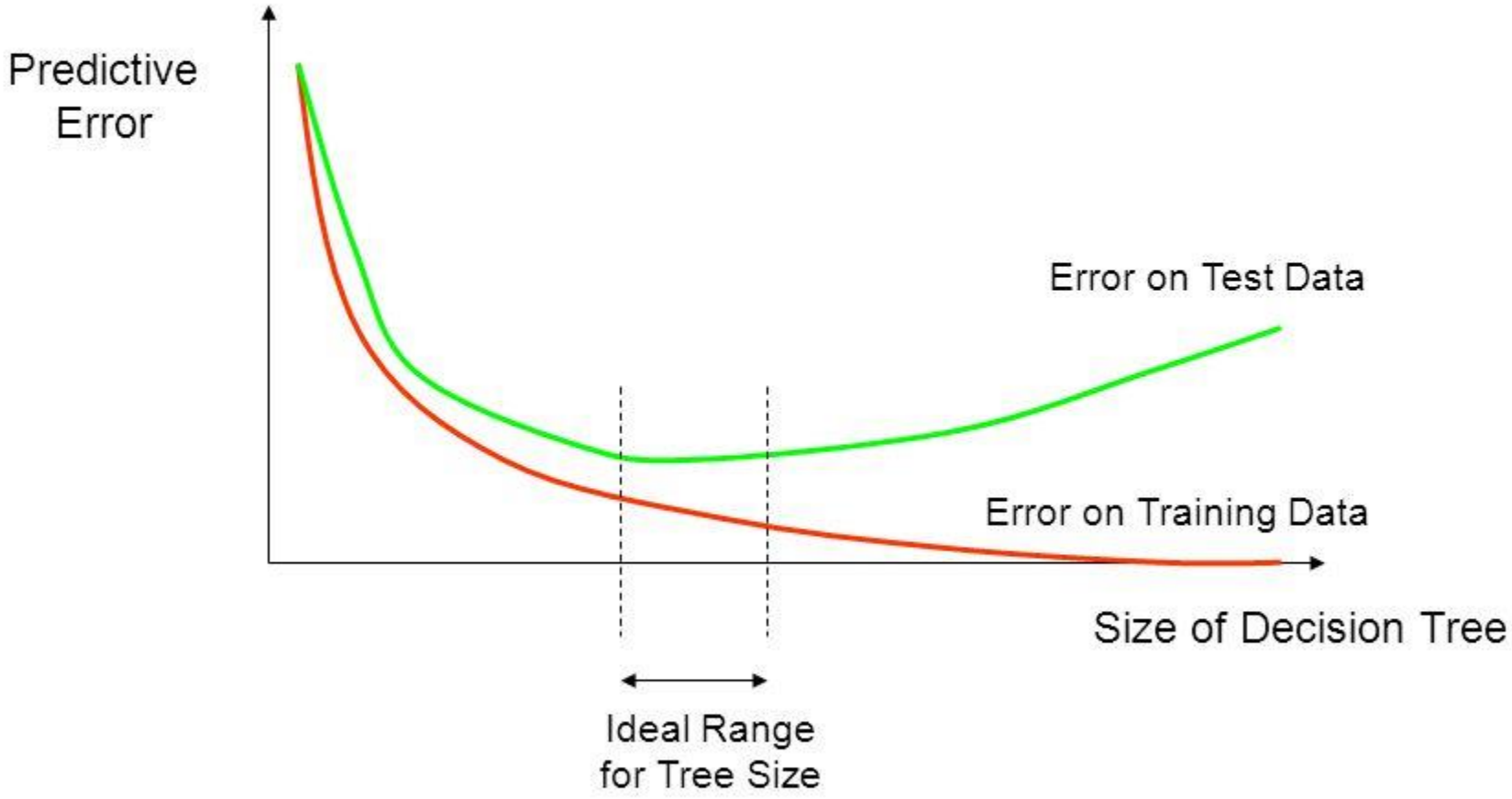
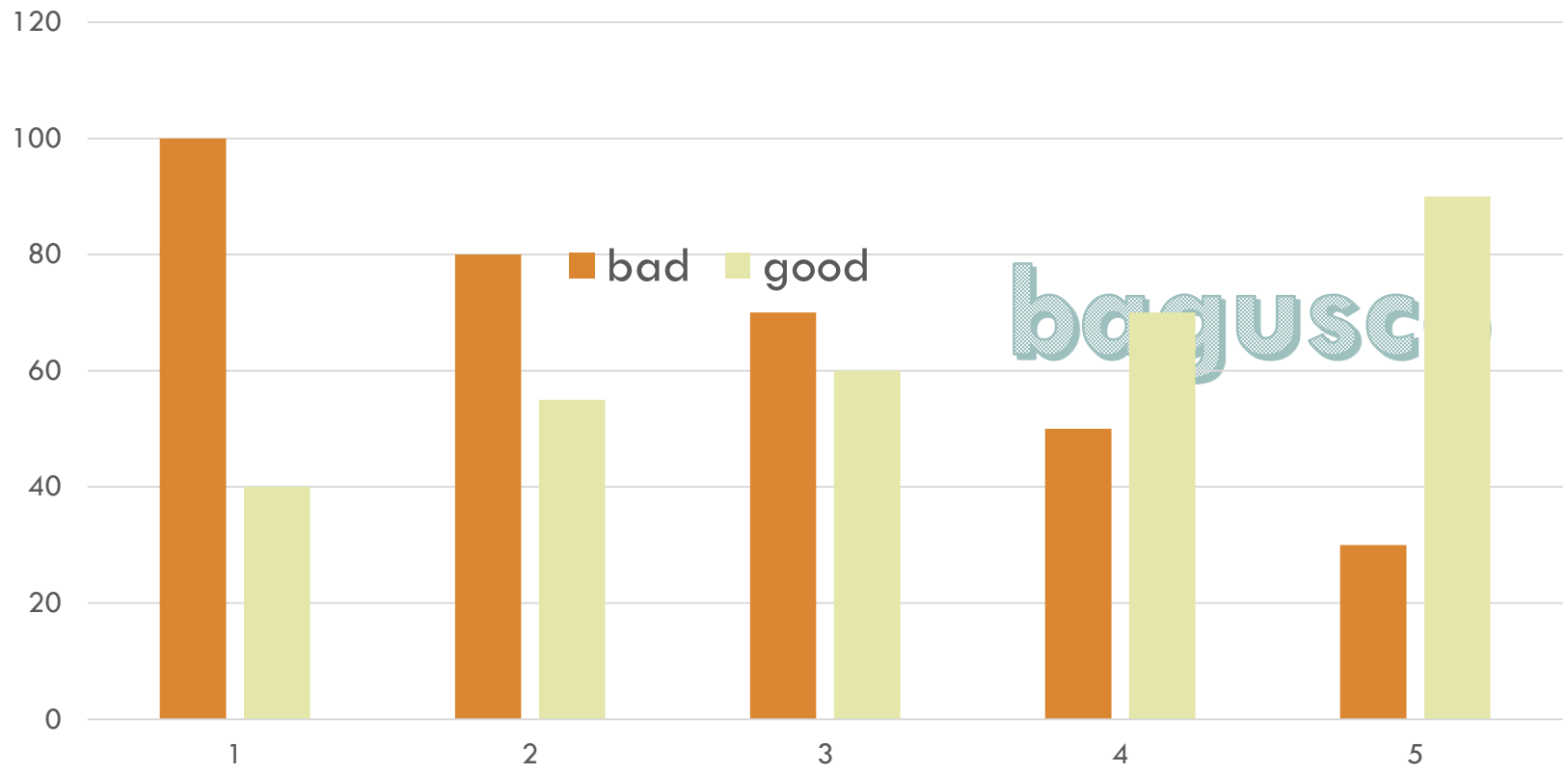
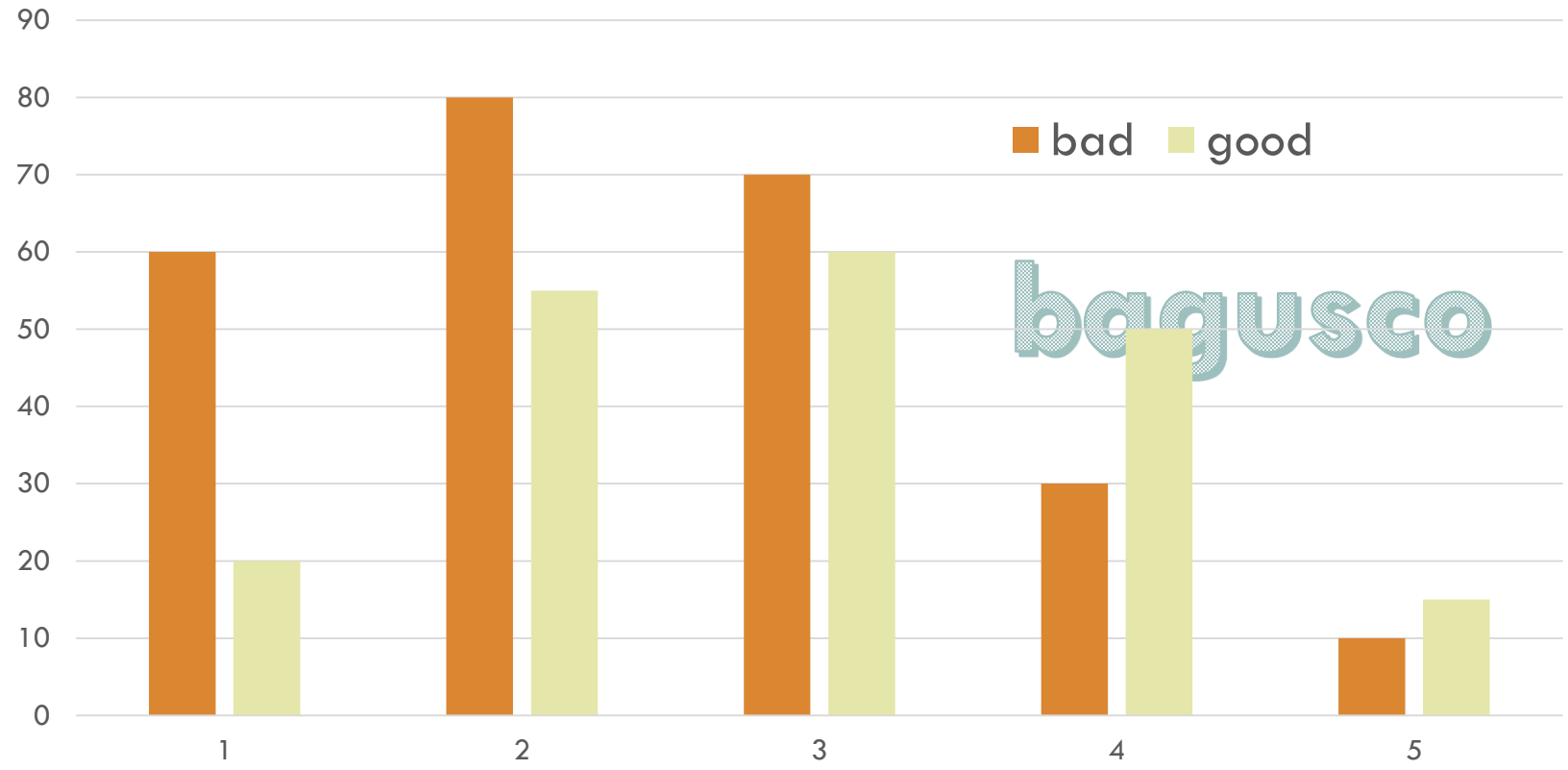
How to Choose the Right-Sized Tree?




120
100
80
60
40
20
0
bad
good
1
2
3
4
5


90
80
70
60
50
40
30
20
10
0
1
2
3
4
5
bad
good
bagusco

Departemen Statistika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Pertanian Bogor

PEMODELAN KLASIFIKASI

PERTEMUAN #10 DAN #11

PENGENALAN ENSEMBLE LEARNING

Bagus Sartono
bagusco@ipb.ac.id bagusco@gmail.com

Basic Ideas

Suppose we want to predict default status of customer based on the following predictors:

age	Age in years
ed	Level of education
employ	Years with current employer
address	Years at current address
income	Household income in thousands
debtinc	Debt to income ratio (x100)
creddebt	Credit card debt in thousands
othdebt	Other debt in thousands

We can use: binary logistic regression (BLR), discriminant analysis (DA), etc

Basic Ideas

Customer	BLR	DA	Actual
1	0	1	0
2	1	0	1
3	0	0	0
4	0	0	0
5	0	0	0
6	1	1	1
7	1	0	0
8	0	1	0
…			

note: 1 → default; 0 → not default

- After modeling, we obtain classification accuracy:
  - BLR: 80%
  - DA: 78%

  which method we choose?
- Can we combine both method so that the accuracy will increase?
  → Ensemble approach

Rationale

There is no algorithm that is always the most accurate

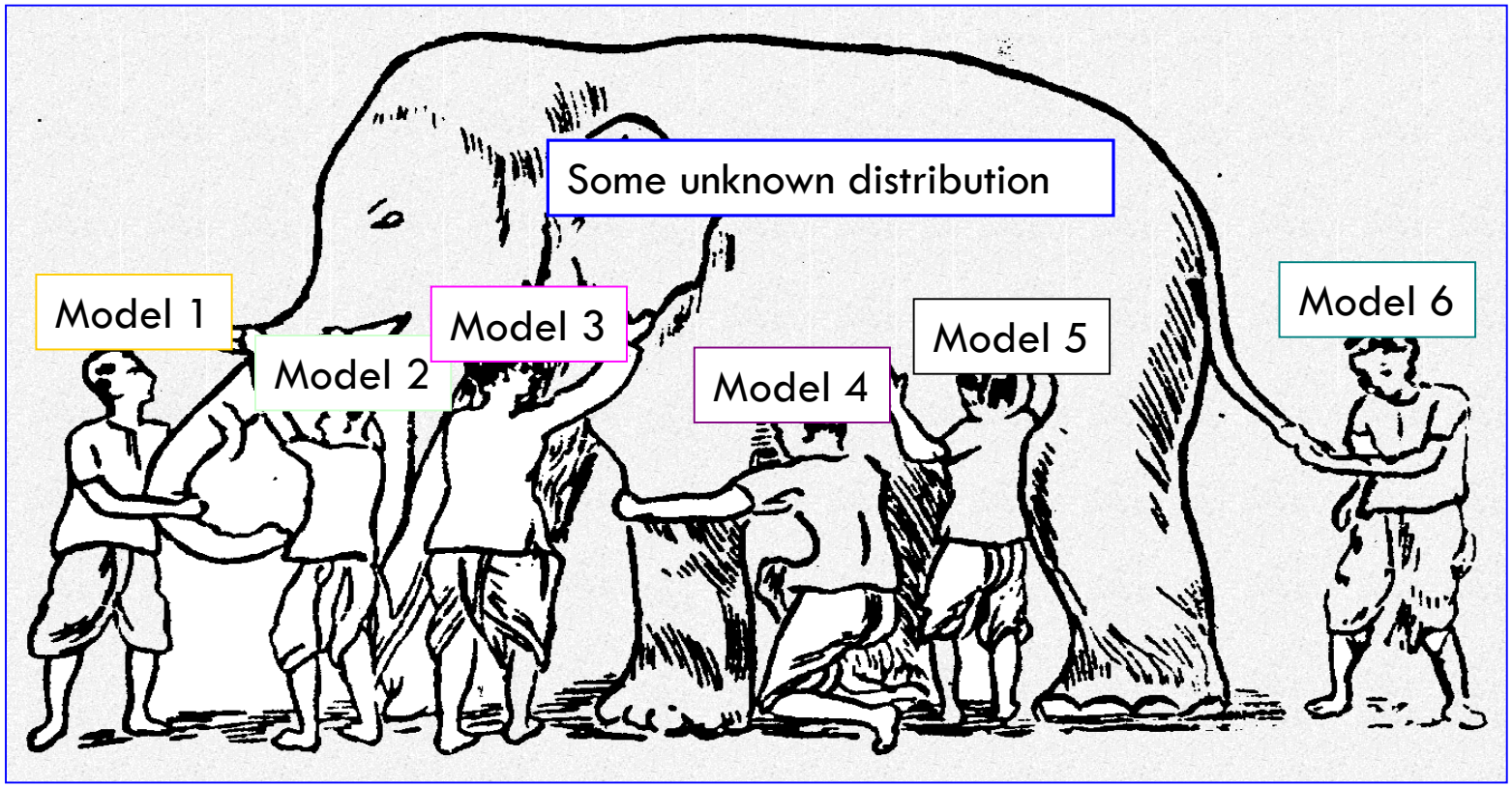
Generate a group of base-learners which when combined has higher accuracy

Each algorithm makes assumptions which might be or not be valid for the problem at hand.

Different learners use different

- Algorithms
- Hyperparameters
- Representations (Modalities)
- Training sets
- Subproblems

Why Ensemble Works?



Ensemble gives the global picture!

Why does it work?

Suppose there are 25 base classifiers

- Each classifier has error rate, $\varepsilon = 0.35$
- Assume classifiers are independent
- Probability that the ensemble classifier makes a wrong prediction:

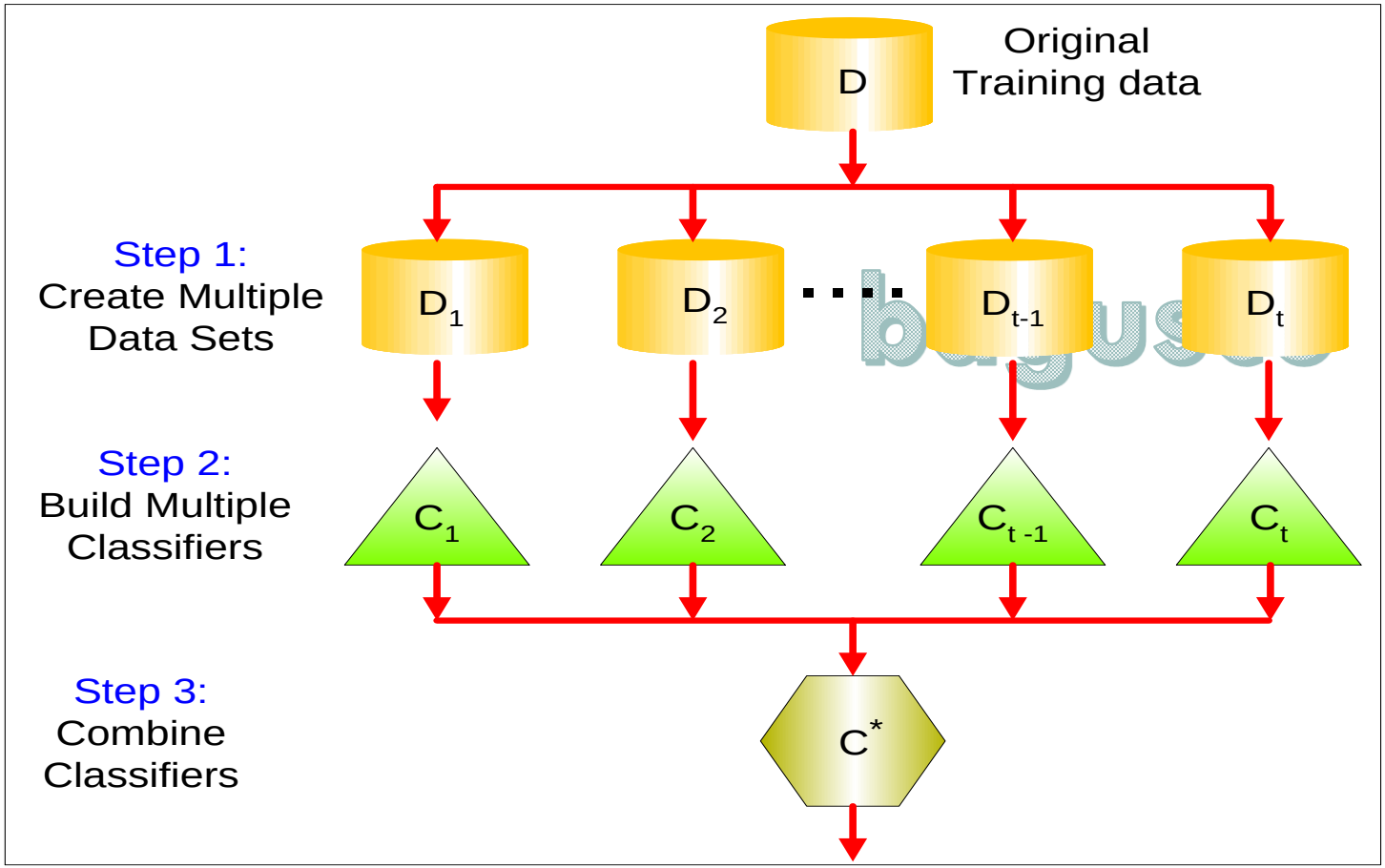
$$\sum_{i=13}^{25} \binom{25}{i} \varepsilon^i (1-\varepsilon)^{25-i} = 0.06$$

What is the Main Challenge for Developing Ensemble Models?

The main challenge is not to obtain highly accurate base models, but rather to obtain base models which make different kinds of errors.

High accuracies can be accomplished if different base models misclassify different training examples, even if the base classifier accuracy is low.

General Idea



Bagging

Bagging stands for Bootstrap Aggregation

Use bootstrapping to generate *L* training sets and train one base-learner with each

Use voting (Average or median with regression)

Unstable algorithms profit from bagging

BAGGING

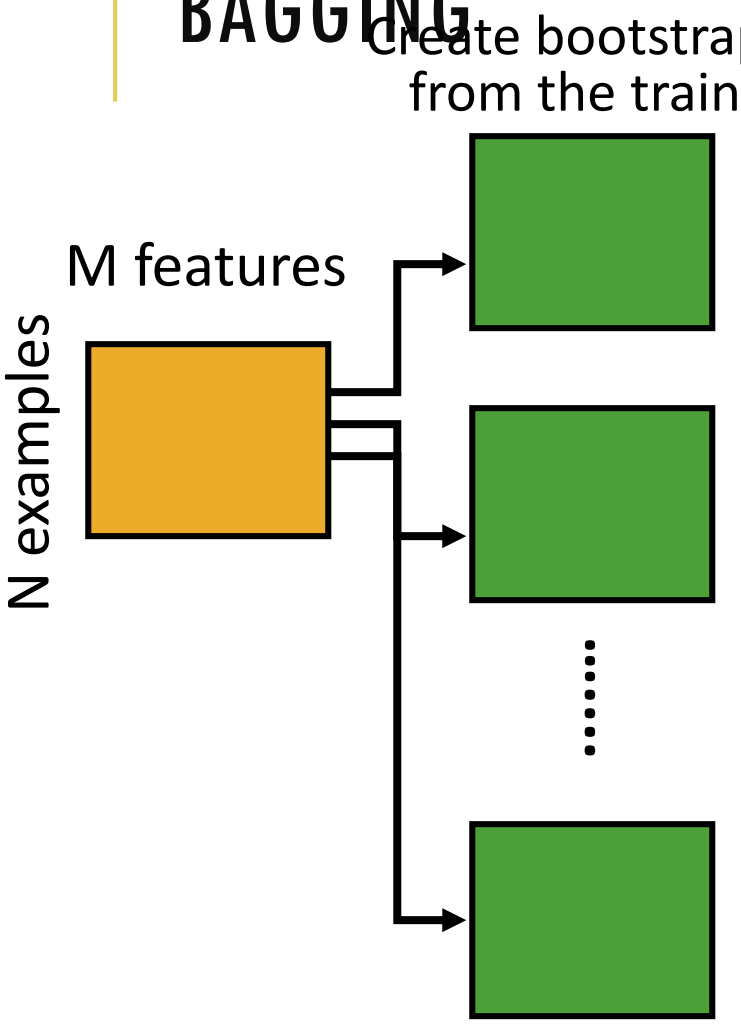
Training Data

M features

N examples

bagusco

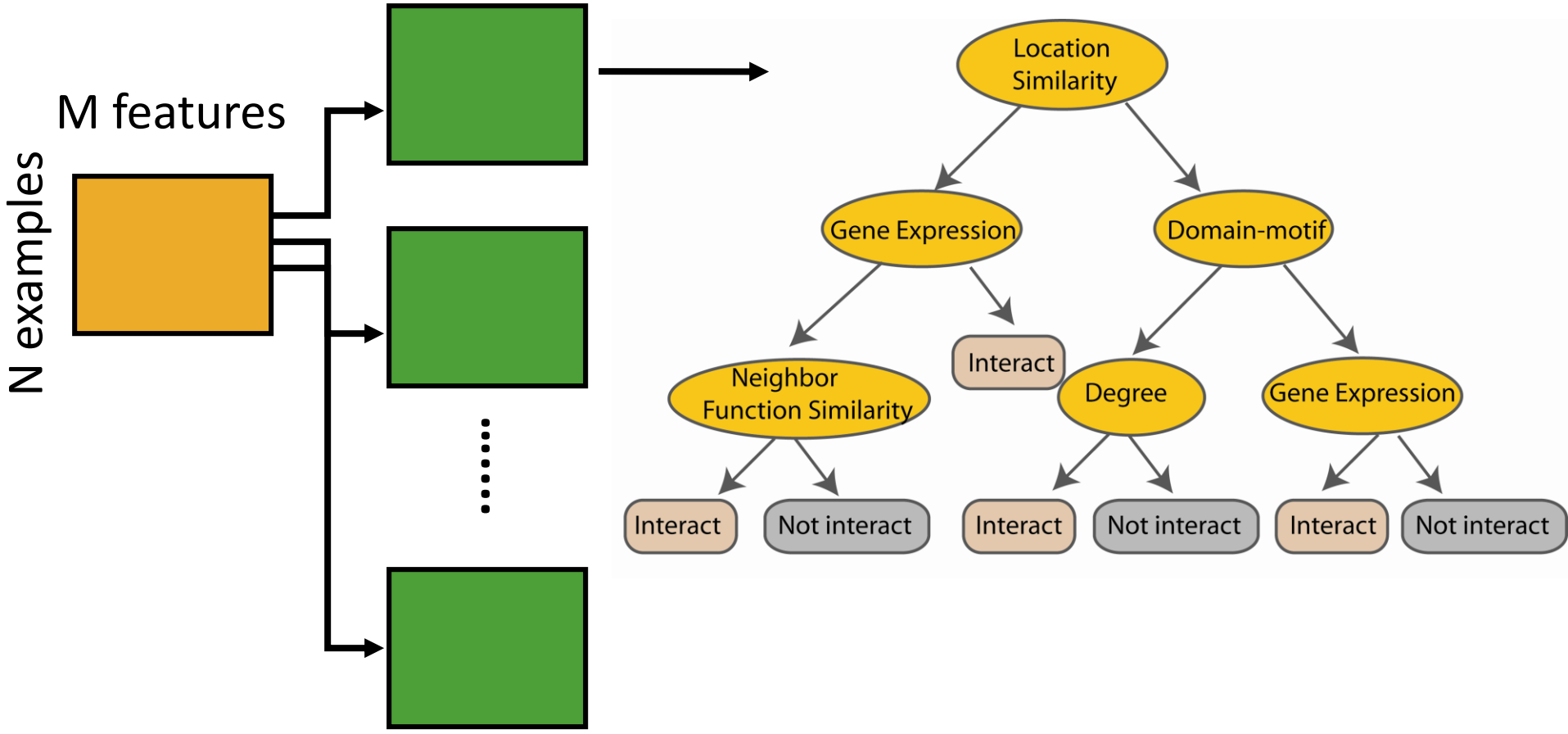
BAGGING



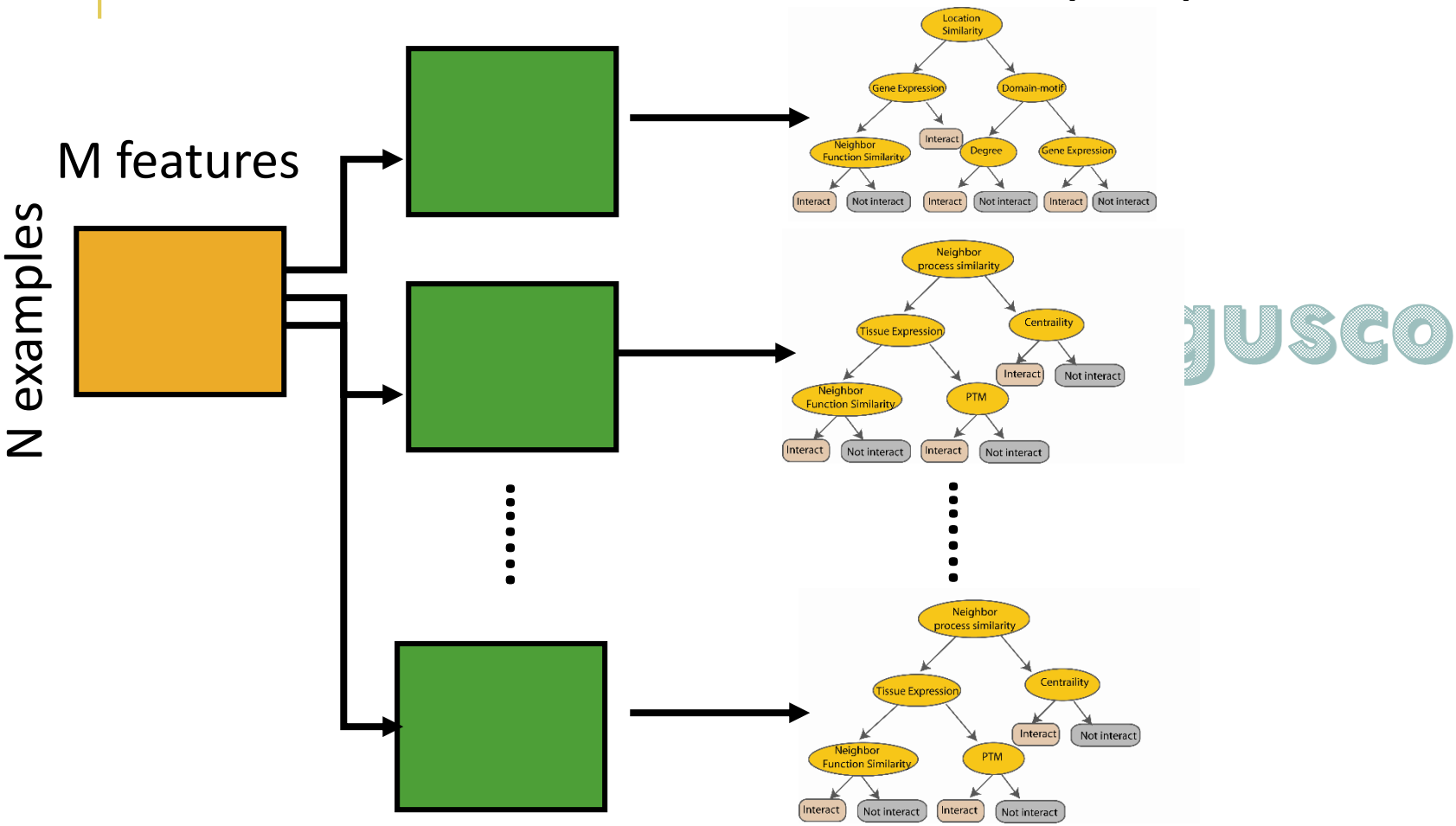
bagusco

BAGGING

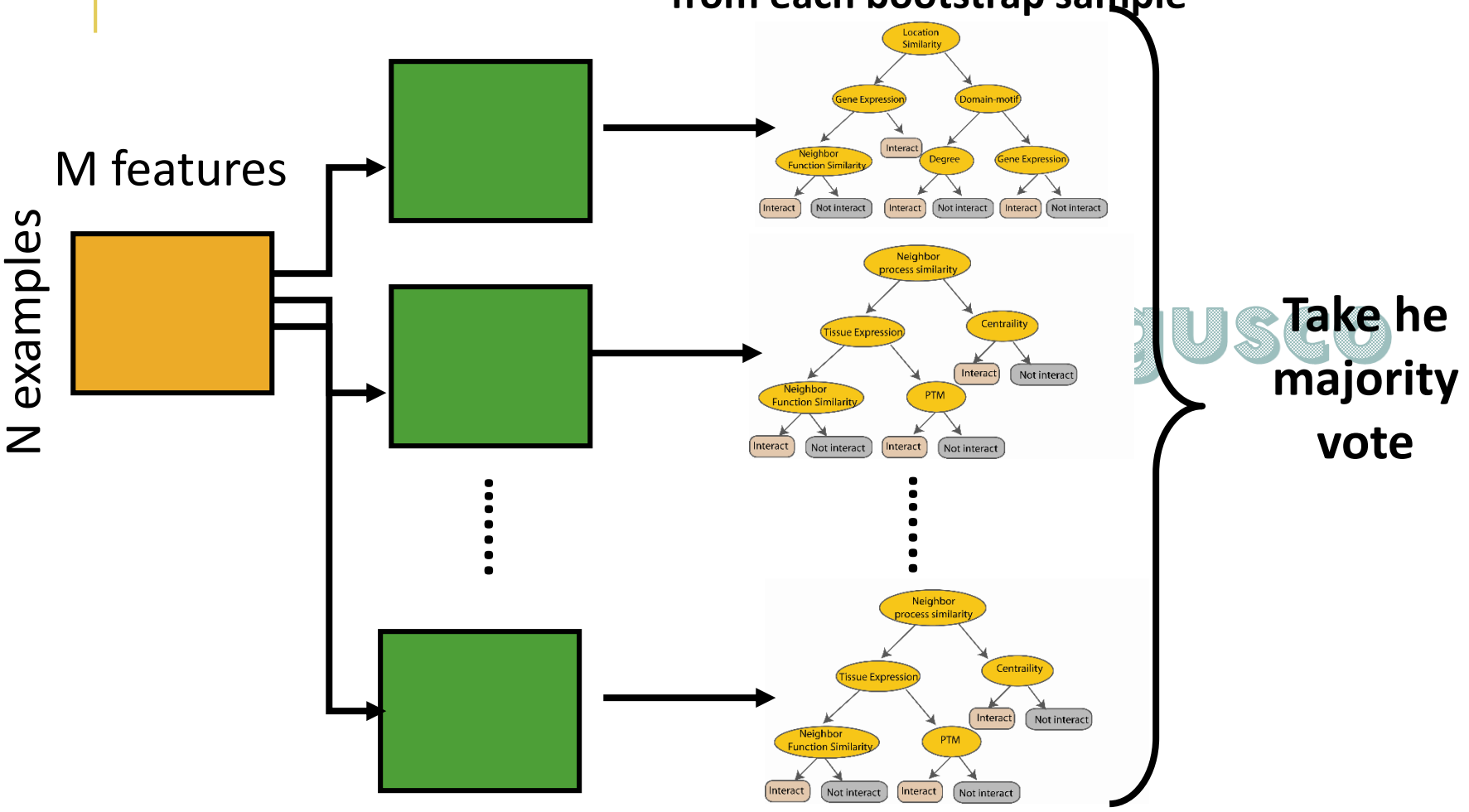
Construct a decision tree



BAGGING


Create decision tree
from each bootstrap sample
M features
N examples
Location Similarity
Gene Expression
Domain-motif
Neighbor Function Similarity
Interact
Degree
Gene Expression
Interact
Not interact
Interact
Not interact
Interact
Not interact
Neighbor process similarity
Tissue Expression
Centrality
Interact
Not interact
Neighbor Function Similarity
PTM
Interact
Not interact
Interact
Not interact
Neighbor process similarity
Tissue Expression
Centrality
Interact
Not interact
Neighbor Function Similarity
PTM
Interact
Not interact
Interact
Not interact

BAGGING


Create decision tree from each bootstrap sample
M features
N examples
Location Similarity
Gene Expression
Domain-motif
Neighbor Function Similarity
Interact
Degree
Gene Expression
Interact
Not interact
Interact
Not interact
Interact
Not interact
Neighbor process similarity
Tissue Expression
Centrality
Interact
Not interact
Neighbor Function Similarity
PTM
Interact
Not interact
Interact
Not interact
Neighbor process similarity
Tissue Expression
Centrality
Interact
Not interact
Neighbor Function Similarity
PTM
Interact
Not interact
Interact
Not interact
Take he majority vote

RANDOM FOREST

Definition

Random forest merupakan salah satu bentuk teknik klasifikasi ensemble yang tersusun atas banyak pohon klasifikasi dan kelas prediksi merupakan modus (suara terbanyak) dari kelas prediksi yang dihasilkan oleh masing-masing pohon

Istilah ini berasal dari **random decision forests** yang diusulkan pertama kali oleh Tin Kam Ho dari Bell Labs pada tahun 1995.

Metode ini merupakan gabungan dari ide Breiman tentang bagging dan seleksi variabel secara acak.

Review tentang Classification Tree atau Decision Trees

Pohon klasifikasi merupakan teknik klasifikasi tunggal yang digabungkan dalam metode random forest.

Salah satu algoritma yang banyak digunakan adalah CART… classification and regression tree.

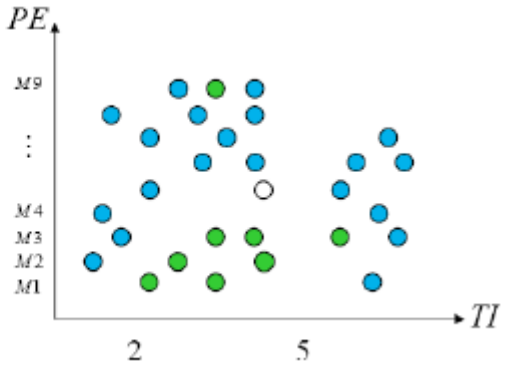
CART … greedy, top-down binary, recursive partitioning, membagi-bagi ruang data menjadi gugus-gugus saling lepas berbentuk rectangular

- Gugus-gugus tersebut dibentuk sedemikian rupa sehingga bersifat homogen kelas variabel responnya
- Prediksi didasarkan pada kelas terbanyak pada simpul akhir

Review tentang Classification Tree atau Decision Trees

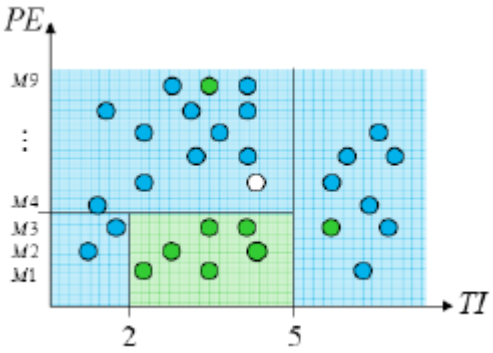
Ilustras

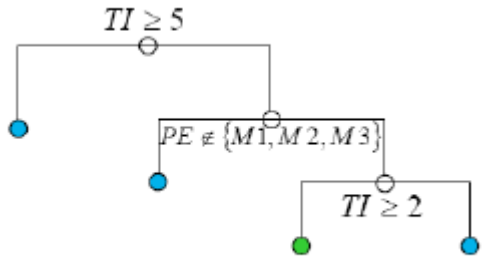
TI	PE	Response
1.0	M2	good
2.0	M1	bad
...	...	...
4.5	M5	?



JSCO

Ilustra





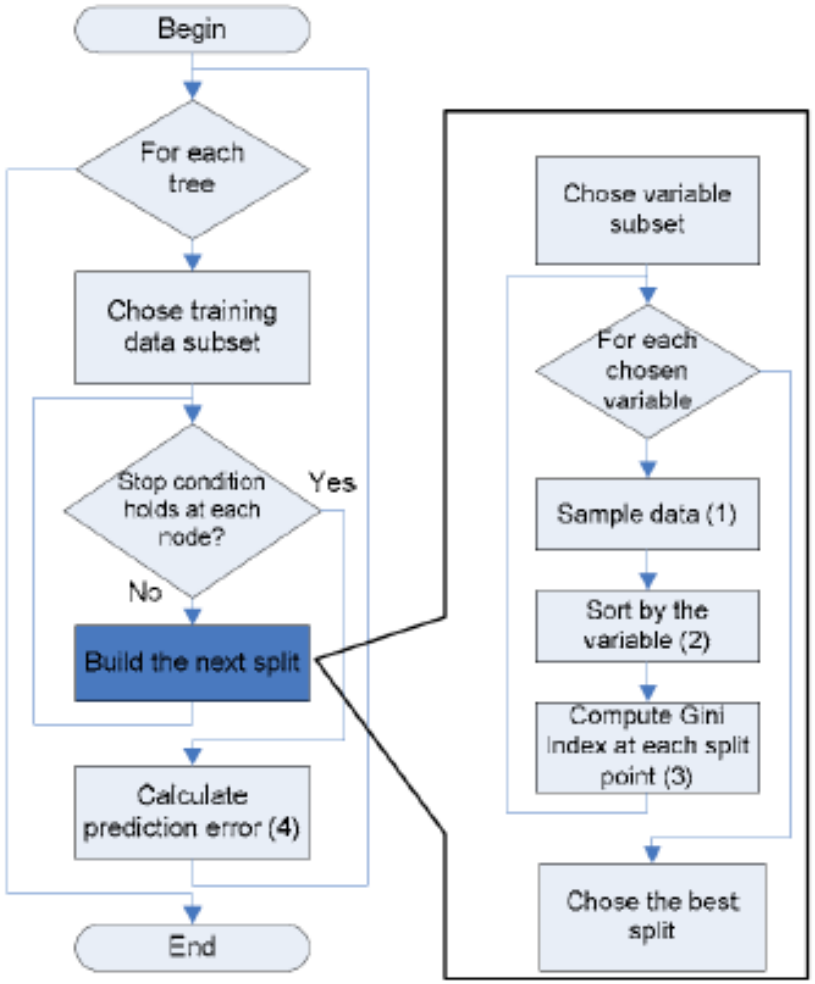
Algoritma umum Random Forest

Setiap pohon dibentuk dengan menggunakan prosedur berikut:

1. Andaikan banyaknya observasi pada data training adalah N dan banyaknya variabel prediktor adalah M.
2. Tentukan besaran m < M, yaitu banyaknya variabel yang dievaluasi dalam setiap pembentukan sekatan (split) pada pembuatan pohon
3. Lakukan pengambilan sampel secara bootstrap (sampling with replacement) sebanyak n dari N buah observasi. Observasi sisanya yang tidak terpilih akan digunakan untuk menduga tingkat kesalahan.
4. Untuk setiap simpul, pilih secara acak m variabel untuk digunakan menentukan sekatan terbaik.
5. Setiap pohon dibentuk tanpa ada proses pemangkasan.

Prediksi terhadap amatan baru dilakukan pada setiap pohon. Selanjutnya proses majority vote dilakukan untuk menghasilkan prediksi dari random forest.

Flow chart dari algoritma



USCO

Random Forest – practical consideration

Berapa banyak pohon?

Berapa banyak variabel yang dipilih secara acak?

bagusco

RANDOM FOREST CLASSIFIER

Random forest classifier, an extension to bagging which uses *de-correlated* trees.

RANDOM FOREST CLASSIFIER

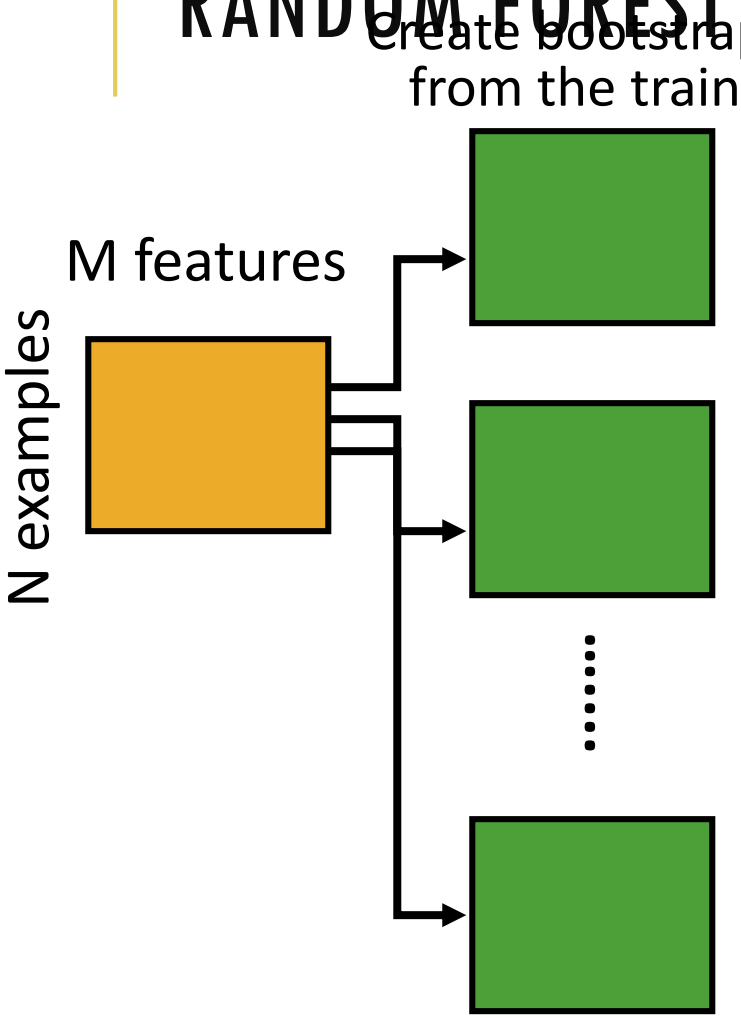
Training Data

M features

N examples

bagusco

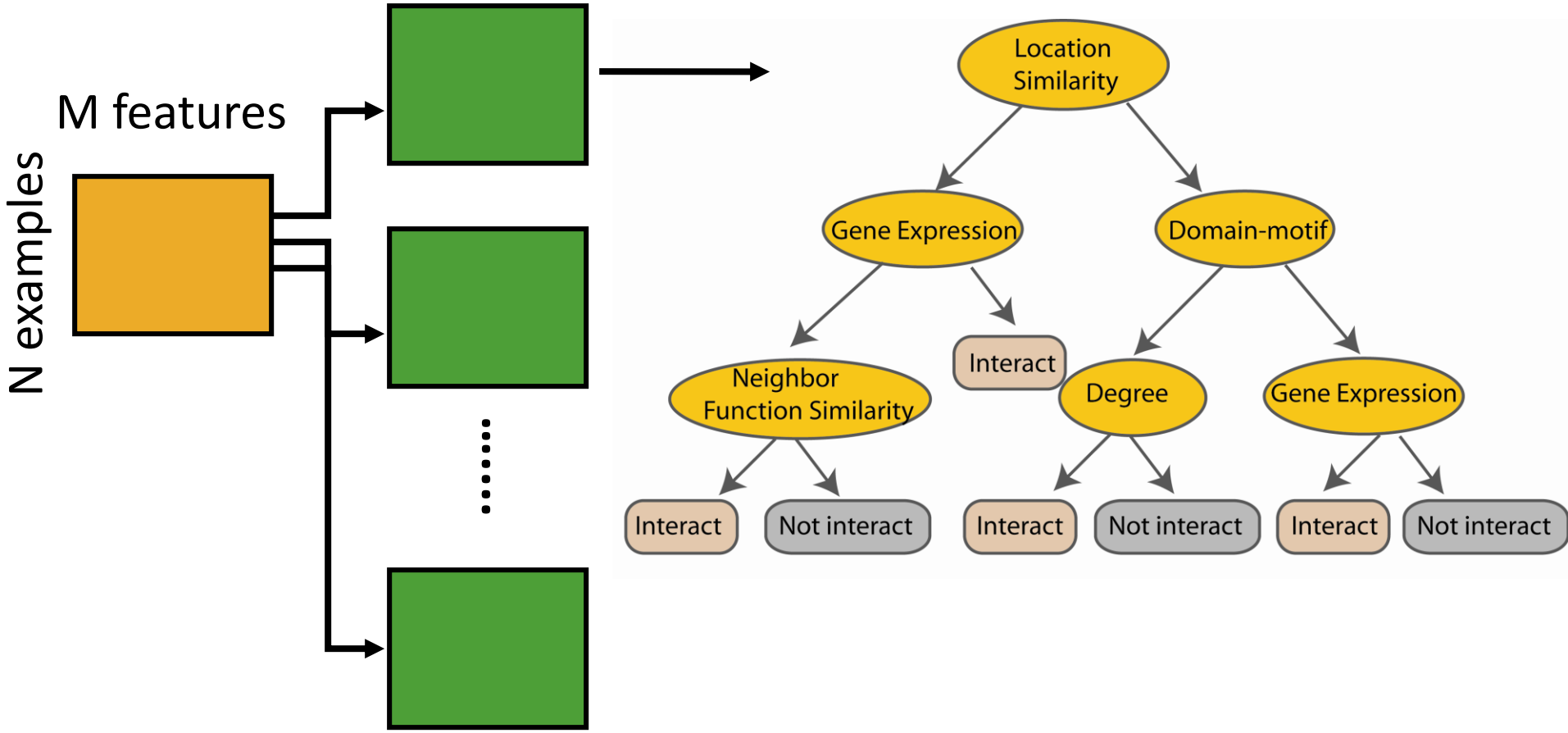
RANDOM FOREST CLASSIFIER



bagusco

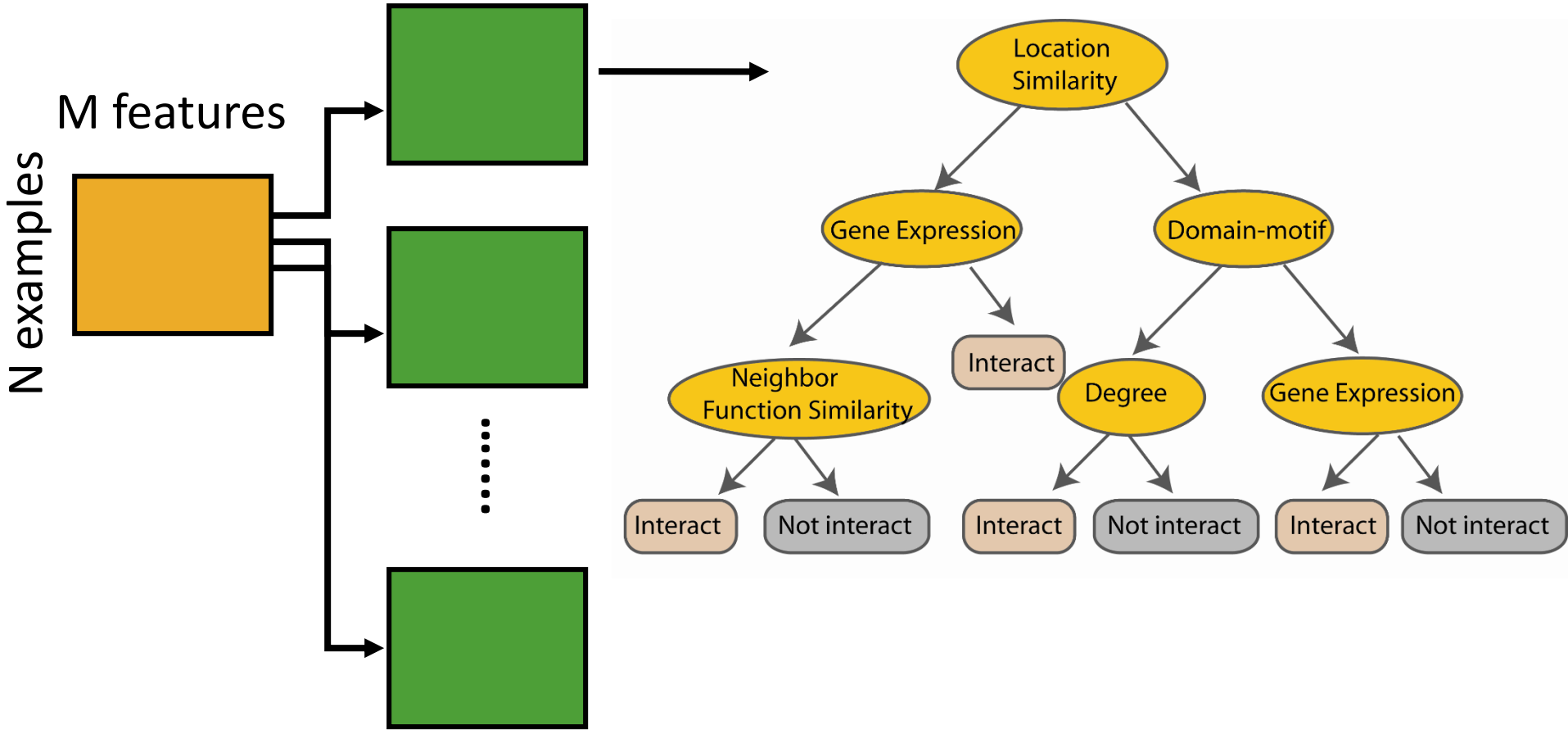
RANDOM FOREST CLASSIFIER

Construct a decision tree



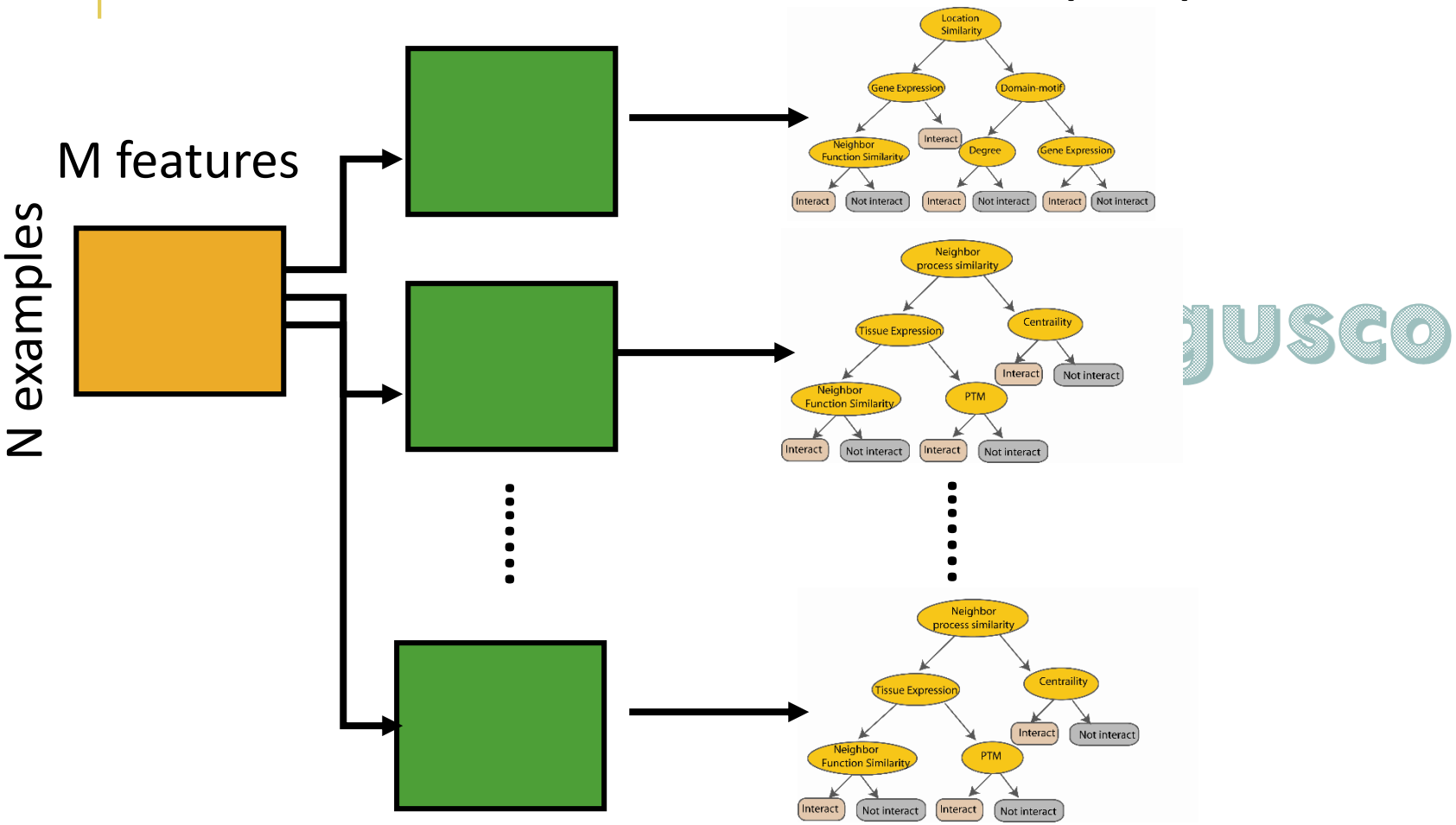
RANDOM FOREST CLASSIFIER

At each node in choosing the split feature choose only among $m<M$ features

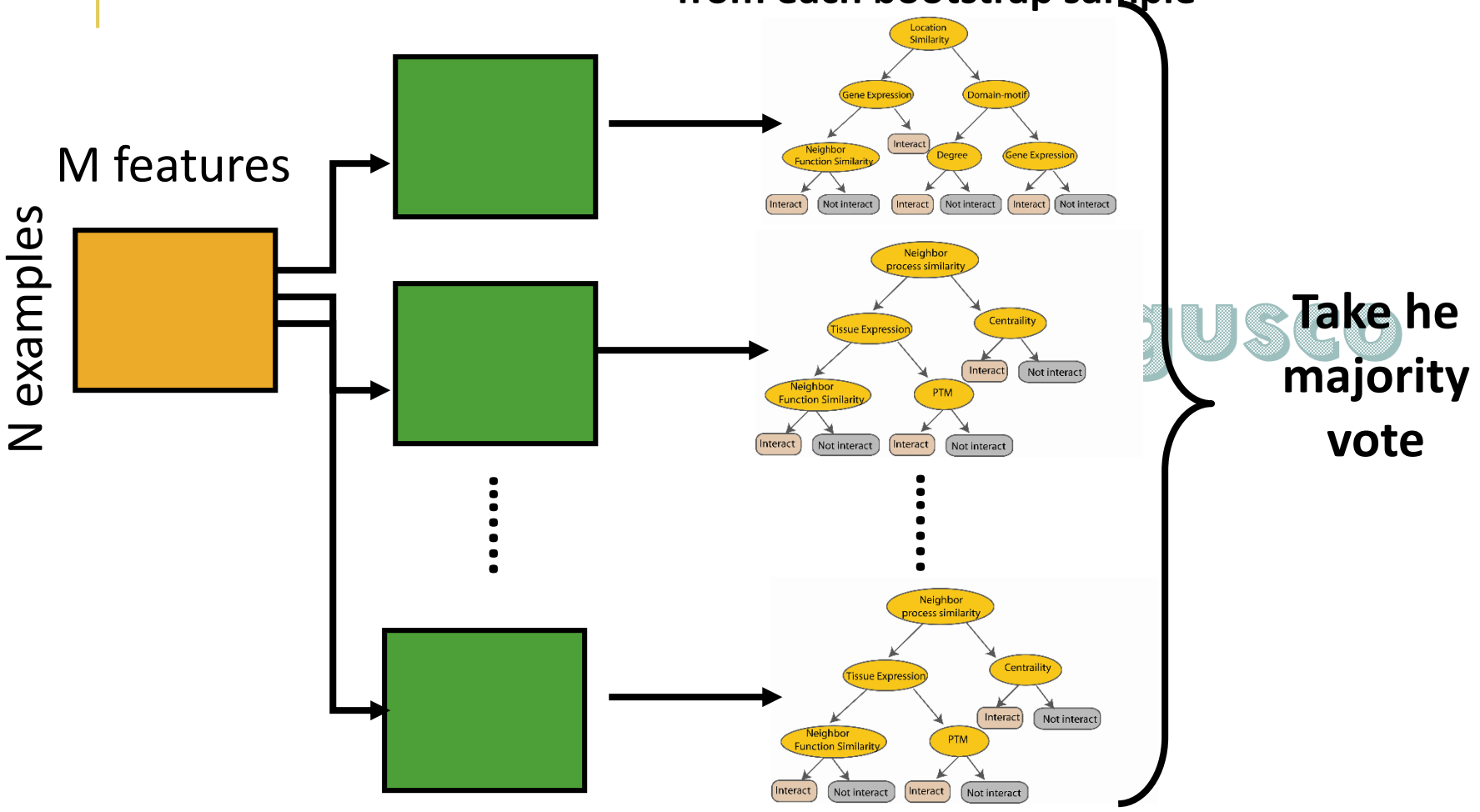


RANDOM FOREST CLASSIFIER

Create decision tree from each bootstrap sample



RANDOM FOREST CLASSIFIER



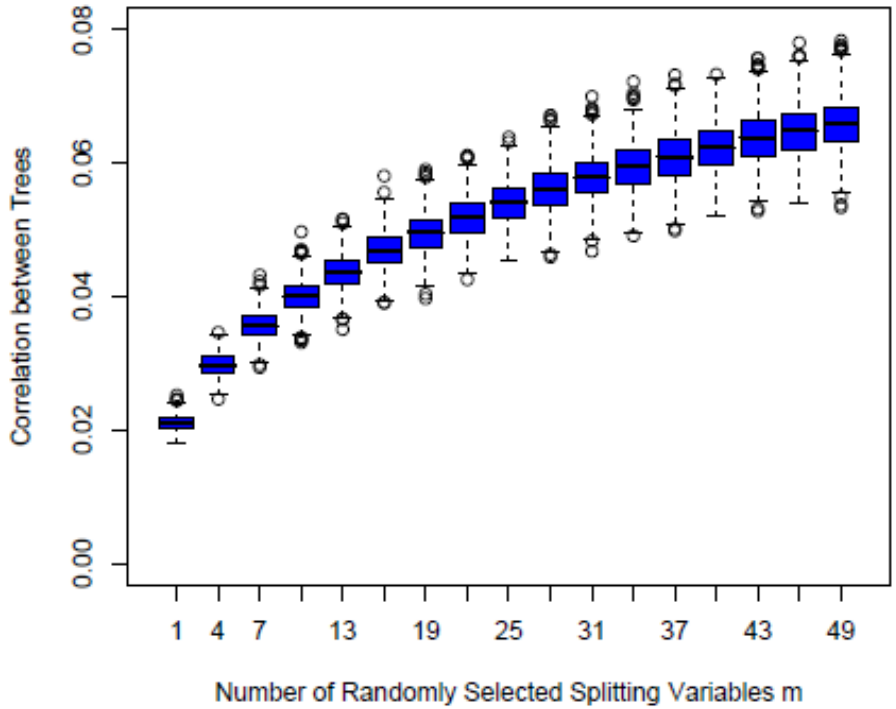


FIGURE 15.9. *Correlations between pairs of trees drawn by a random-forest regression algorithm, as a function of m. The boxplots represent the correlations at 600 randomly chosen prediction points x.*

```
bankloan <- read.csv("D:/bankloan.csv", header=TRUE)
head(bankloan)

#set.seed(20)
acak <- sample(1:nrow(bankloan), 450, replace=FALSE)
bankloan.training <- bankloan[acak,]
bankloan.testing <- bankloan[-acak,]

library(rpart)
model.pohon <- rpart(as.factor(default) ~ age + ed + employ + address
                     + income + debtinc + creddebt + othdebt,
                     data=bankloan.training)
prob.prediksi <- predict(model.pohon, bankloan.testing)
prediksi <- ifelse(prob.prediksi[,2] > 0.5, 1, 0)

tabel <- table(bankloan.testing$default, prediksi)
akurasi <- (tabel[1,1] + tabel[2,2])/sum(tabel) * 100
akurasi
```

```
library(randomForest)
#set.seed(100)
model.forest <- randomForest(as.factor(default) ~ age + ed + employ + address
                  + income + debtinc + creddebt + othdebt,
                  data=bankloan.training, importance=TRUE, ntree=2000, mtry=3)
prediksi.rf <- predict(model.forest, bankloan.testing)

tabel.rf <- table(bankloan.testing$default, prediksi.rf)
akurasi.rf <- (tabel.rf[1,1] + tabel.rf[2,2])/sum(tabel.rf) * 100
akurasi.rf

#importance(model.forest)
varImpPlot(model.forest)
#getTree(model.forest, labelVar=TRUE, k=2)
```

Perbandingan Akurasi Pohon Tunggal dan Random Forest

```
for(i in 1:100){
acak <- sample(1:nrow(bankloan), 450, replace=FALSE)
bankloan.training <- bankloan[acak,]
bankloan.testing <- bankloan[-acak,]


model.pohon <- rpart(as.factor(default) ~ age + ed + employ + address
                     + income + debtinc + creddebt + othdebt,
                     data=bankloan.training)
prob.prediksi <- predict(model.pohon, bankloan.testing)
prediksi <- ifelse(prob.prediksi[,2] > 0.5, 1, 0)

tabel <- table(bankloan.testing$default, prediksi)
akurasi[i] <- (tabel[1,1] + tabel[2,2])/sum(tabel) * 100
```

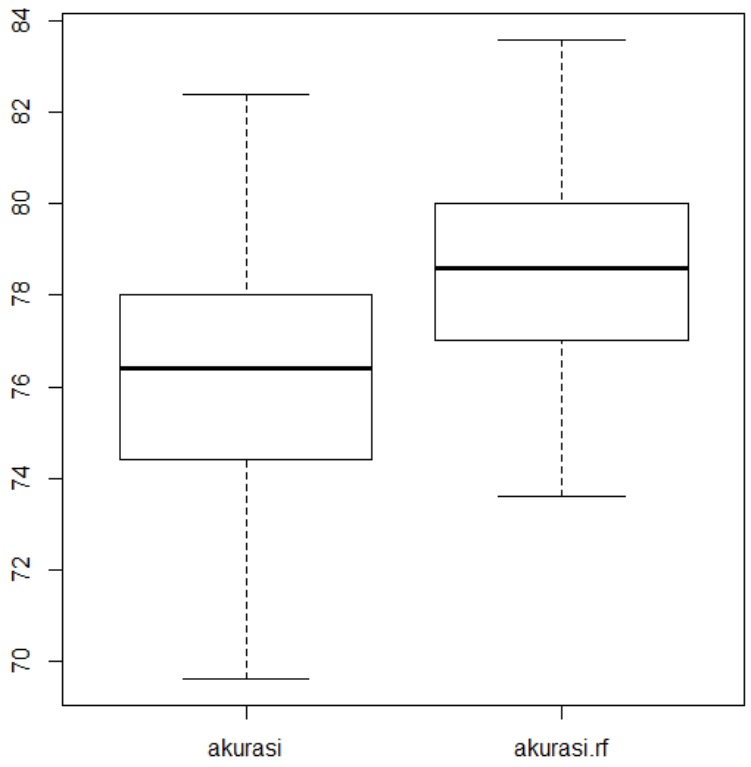
bagusco

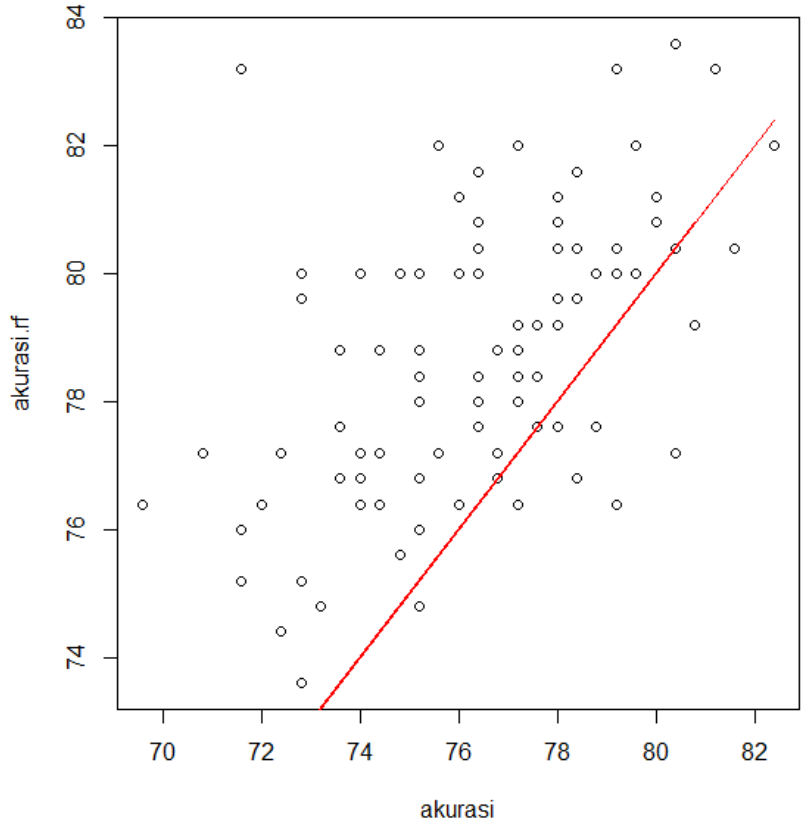
```
model.forest <- randomForest(as.factor(default) ~ age + ed + employ + address
                             + income + debtinc + creddebt + othdebt,
                             data=bankloan.training, importance=TRUE, ntree=2000, mtry=3)
prediksi.rf <- predict(model.forest, bankloan.testing)

tabel.rf <- table(bankloan.testing$default, prediksi.rf)
akurasi.rf[i] <- (tabel.rf[1,1] + tabel.rf[2,2])/sum(tabel.rf) * 100
}

boxplot(cbind(akurasi, akurasi.rf))
plot(akurasi, akurasi.rf)
points(akurasi, akurasi, type="l")
```

Perbandingan Akurasi Pohon Tunggal dan Random Forest





Features and Advantages

The advantages of random forest are:

It is one of the most accurate learning algorithms available. For many data sets, it produces a highly accurate classifier.

It runs efficiently on large databases.

It can handle thousands of input variables without variable deletion.

It gives estimates of what variables are important in the classification.

Disadvantages

Random forests have been observed to overfit for some datasets with noisy classification/regression tasks.

For data including categorical variables with different number of levels, random forests are biased in favor of those attributes with more levels. Therefore, the variable importance scores from random forest are not reliable for this type of data.

RM - Additional information

Estimating the test error:

While growing forest, estimate test error from training samples

For each tree grown, 33-36% of samples are not selected in bootstrap, called out of bag (OOB) samples

Using OOB samples as input to the corresponding tree, predictions are made as if they were novel test samples

Through book-keeping, majority vote (classification), average (regression) is computed for all OOB samples from all trees.

Such estimated test error is very accurate in practice, with reasonable N

Conclusions & summary:

Fast fast fast!

- RF is fast to build. Even faster to predict!
- Practically speaking, not requiring cross-validation alone for model selection significantly speeds training by 10x-100x or more.
- Fully parallelizable … to go even faster!

Provide information for predictor selection from large number of candidates

Ability to handle data without preprocessing

- data does not need to be rescaled, transformed, or modified
- resistant to outliers
- automatic handling of missing values

bagusco

terima kasih