Center for Business Intelligence

5.0 Introduction to Classification and Model Evaluation

Eugene Rex L. Jalao, Ph.D.
Associate Professor
Department Industrial Engineering and Operations Research
University of the Philippines Diliman
@thephdataminer

Module 1 of the Business Intelligence and Analytics Track of UP NEC and the UP Center of Business Intelligence

Module 1 Outline

1. Intro to Business Intelligence
   – Case Study on Selecting BI Projects
2. Data Warehousing
   – Case Study on Data Extraction and Report Generation
3. Descriptive Analytics
   – Case Study on Data Analysis
4. Visualization
   – Case Study on Dashboard Design
5. **Classification Analysis**
   – **Case Study on Classification Analysis**
6. Regression and Time Series Analysis
   – Case Study on Regression and Time Series Analysis
7. Unsupervised Learning and Modern Data Mining
   – Case Study on Text Mining
8. Optimization for BI

Outline for this Session

- Introduction to Classification
- Decision Trees
- Software Use
- Alternative Classification Models
- Model Evaluation and Validation
- Case Study

Dataset Structure

Attributes/Columns/Variables $(p+1)$

Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

Rows/ Instances /Tuples /Objects (n)

Predictor Variables/Independent Variables/Control Variables

Response Variable/ Dependent Variable/ Class Variable/ Label Variable/ Target Variable

eljalao@up.edu.ph

Introduction to Classification

Definition 5.1: Classification

- Given a collection of records
  - Multiple **predictor variables** usually $x_1, x_2, \dots x_p$
  - One **categorical response** variable usually y
- Find a model for **predicting** the class variable from the predictor variables.
  - Use historical "training data" to build the model
- Goal: **previously unseen records** should be predicted a class as **accurately as possible.**
  - Use "testing data" to test the accuracy of the model

Introduction to Classification

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

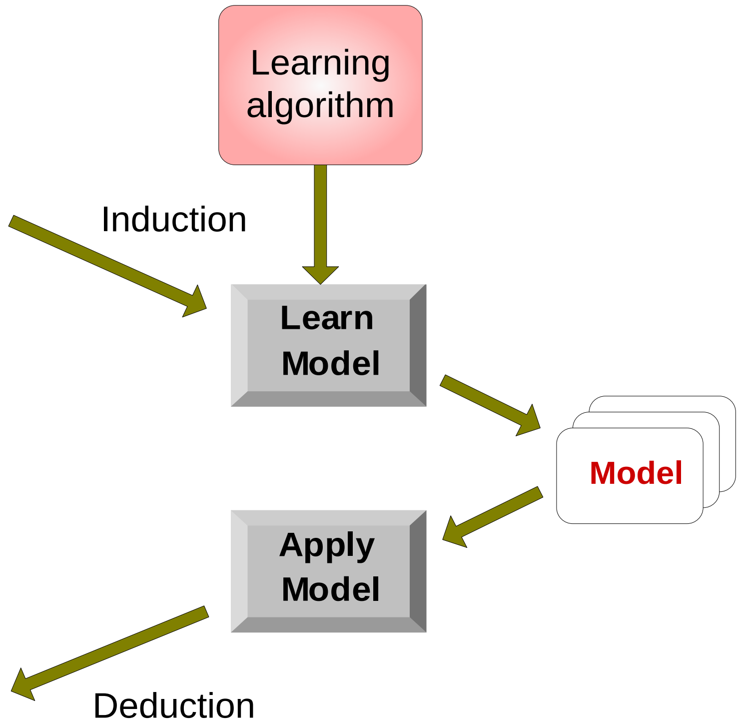


Figure 5.1: A Classification Task

Introduction to Classification

- Some Classification Techniques
  - Decision Tree and Rule-Based Methods
  - Similarity Based Reasoning
  - Neural Networks
  - Support Vector Machines
  - Ensembles

Outline for this Session

- Introduction to Classification
- **Decision Trees**
- Software Use
- Alternative Classification Models
- Model Evaluation and Validation
- Case Study

Decision Trees

Definition 5.2: Decision Trees

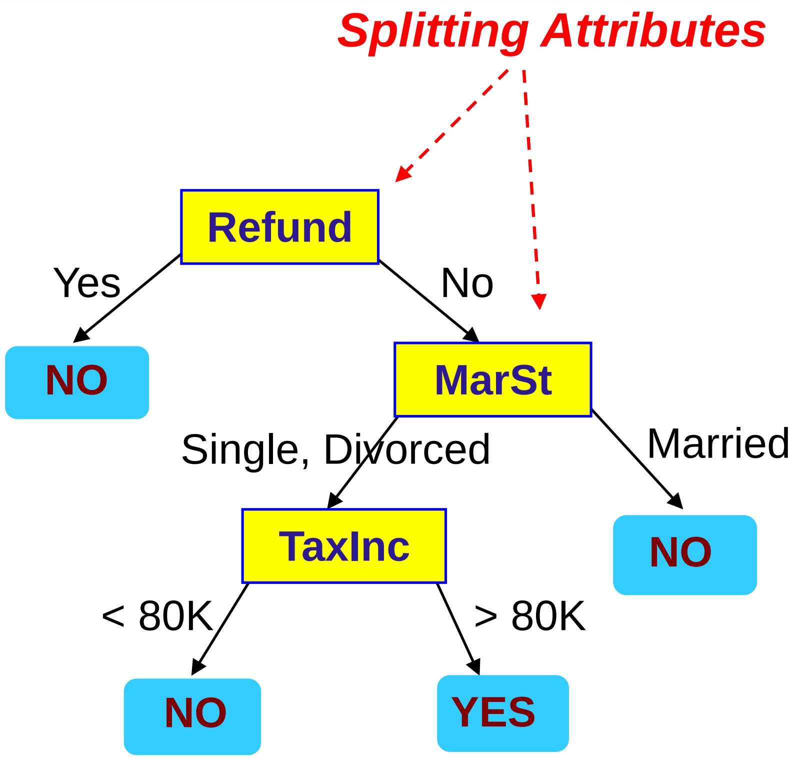
- Decision tree builds classification models in the form of a **tree structure**.
- It breaks down a dataset into **smaller and smaller** subsets while at the same time an associated decision tree is **incrementally developed**.
- The final result is a tree with decision nodes and leaf nodes. A **decision node** (e.g., Outlook) has two or more branches (e.g., Sunny, Overcast and Rainy). **Leaf node** (e.g., Play) represents a classification or decision.

Decision Trees

Example 5.1: Example of a Decision Tree

Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Data



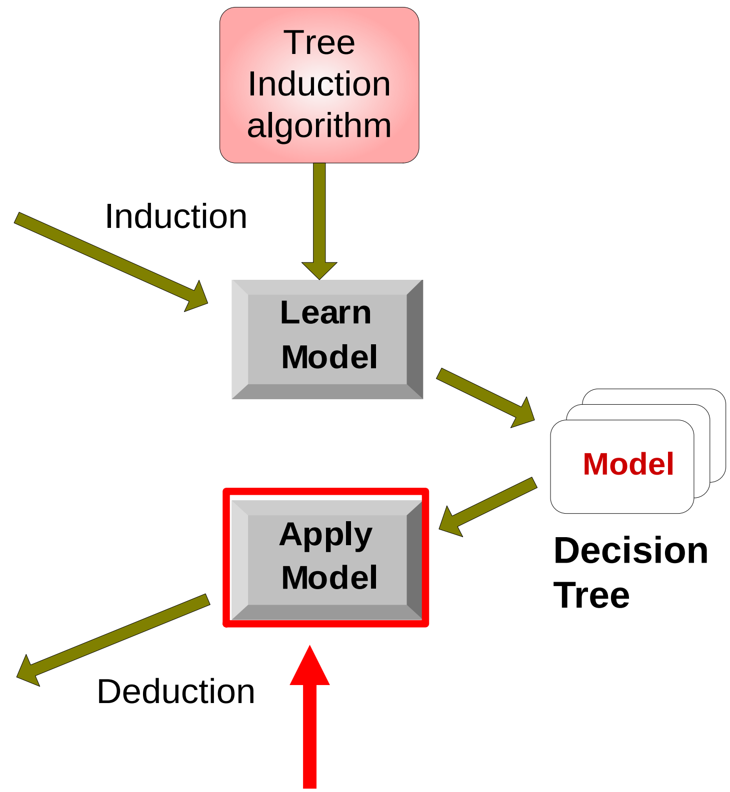
Model: Decision Tree

Decision Trees

Example 5.1 (Cont.): Example of a Decision Tree

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set



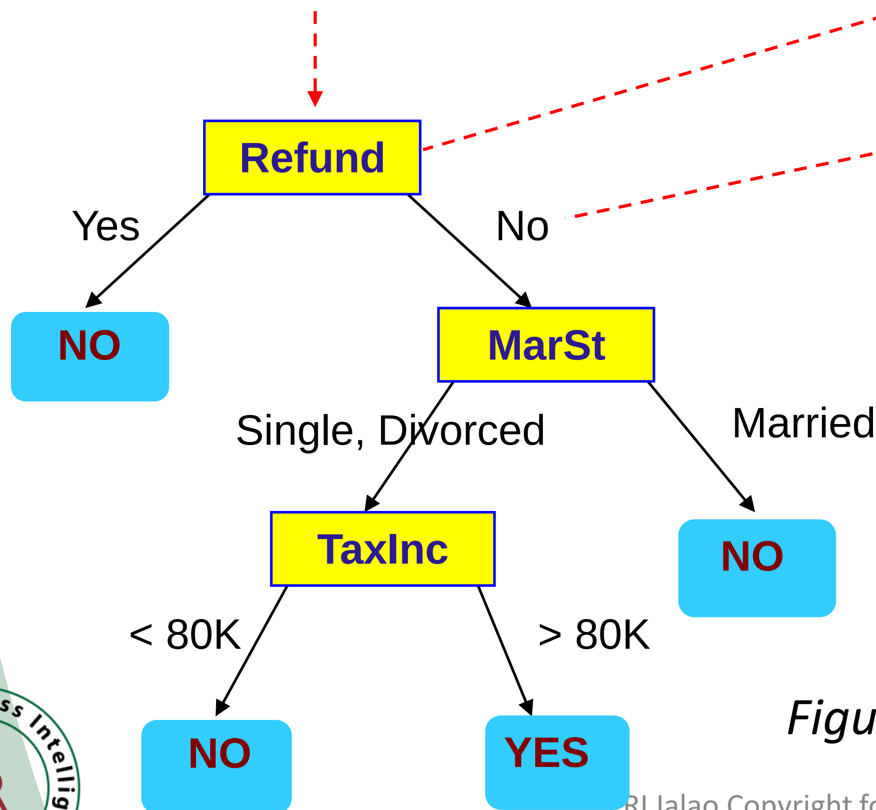
Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

Decision Trees

Example 5.1 (Cont.): Example of a Decision Tree

Start from the root of tree.



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Predict Cheat = “No”

Figure 5.2: A Sample Decision Tree

Decision Trees

Definition 5.3: Prediction Confidence

- Prediction Confidence: Level of **Confidence** we get for each prediction rule
- **Usually computed** on every classification algorithm

Decision Trees

Example 5.2 Calculating Prediction Confidence

- IF Refund = No and Marital Status = Married THEN Cheat= No (10/3)
- The rule is correct 10/13 times
  - There are 13 people that have profile: Refund = No and Marital Status = Married
  - Out of the 13 people that have profile Refund = No and Marital Status = Married, 10 of them have Cheat= No and 3 have Cheat = Yes

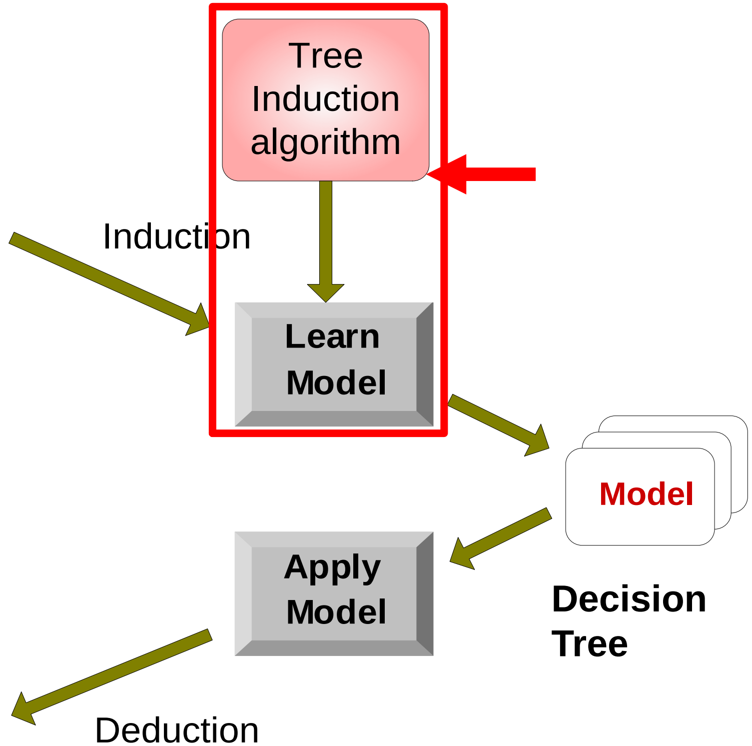
Decision Trees

- We always predict the **majority**
  - 100 yes, 0 no = Prediction is Yes
  - 10 yes, 9 no = Prediction is Yes
  - 10 yes, 10 no = toss coin, Decision Tree is **no good** in predicting class

Decision Trees

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

Decision Trees

- Decision Tree Generation **Algorithms**
  - Hunt's Algorithm (one of the earliest)
  - CART
  - ID3, C4.5
  - SLIQ,SPRINT

Outline for this Session

- Introduction to Classification
- Decision Trees
- **Software Use**
- Alternative Classification Models
- Model Evaluation and Validation
- Case Study

Software Use

- Importing Data
- Learn a Decision Tree from the Data
- Interpret the Results
  - Bank Data.csv
  - Independent Variables
    - Age, region, income, sex, married, children, car, save_act, current_act, and mortgage
  - Response
    - did the customer buy a PEP (Personal Equity Plan) after the last mailing (YES/NO)?

Software Use

- A leading bank's marketing department would like to profile its clients to know which factors lead to the purchase of one of its flagship products: PEP (Personal Equity Plan)
- 600 Clients where gathered from the company's various databases each having variables such as: age, region, income, sex, married, children, car, save_act, current_act, and mortgage

Software Use

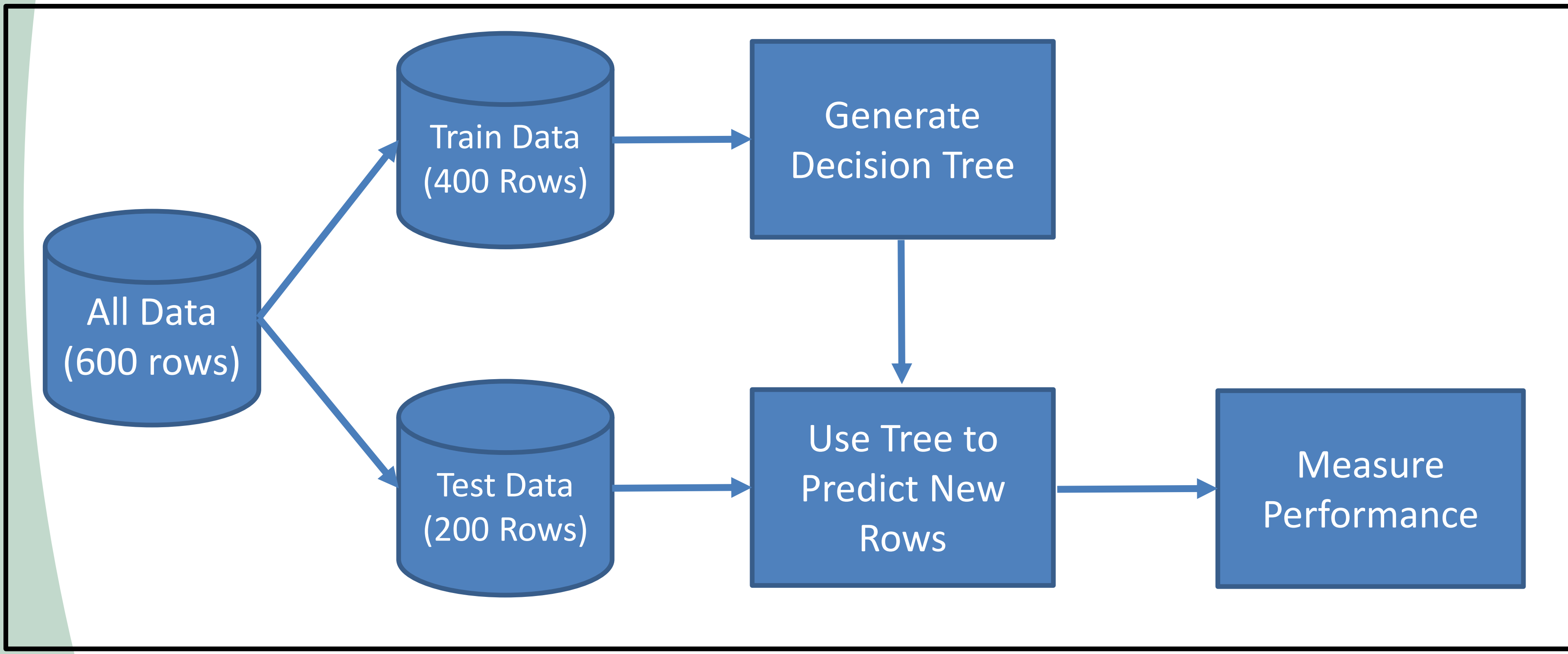


Figure 5.3: Classification Modelling Workflow

Software Use

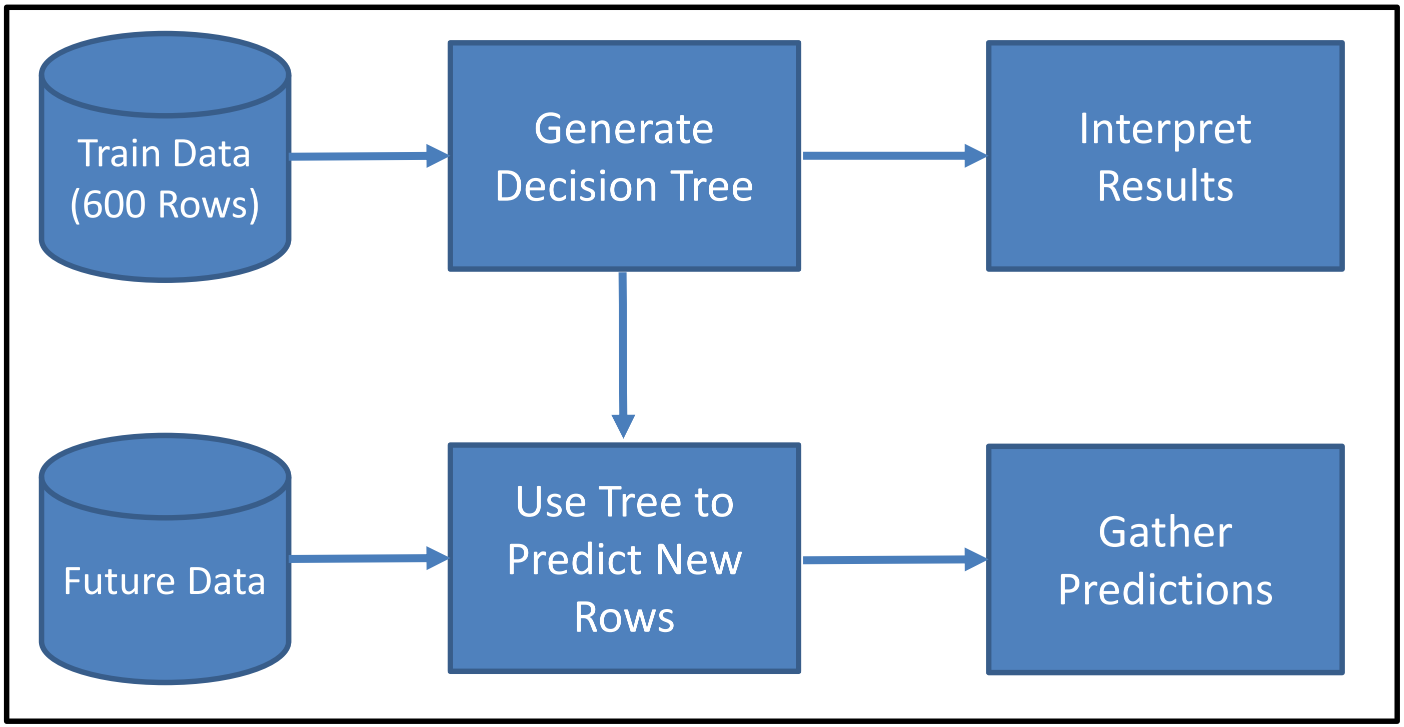


Figure 5.4: Classification Application Workflow

Software Use

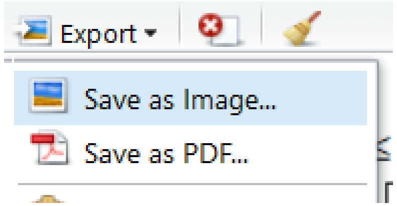
- Type the following lines of code in RStudio and run.

```
bankdata = read.csv("bankdata.csv")
J48Model <- J48(pep ~ age + sex+  region  + income
                + married + children  + car
                + save_act+ current_act+  mortgage
              , data=bankdata)
J48Model
plot(J48Model)
```

```
> J48Model
J48 pruned tree
------------------

children <= 1
|   children <= 0
|   |   married = NO
|   |   |   mortgage = NO: YES (48.0/3.0)
|   |   |   mortgage = YES
|   |   |   |   save_act = NO: YES (12.0)
|   |   |   |   save_act = YES: NO (23.0)
|   |   married = YES
|   |   |   save_act = NO
|   |   |   |   mortgage = NO
|   |   |   |   |   income <= 21506.2
|   |   |   |   |   |   age <= 41: NO (11.0/1.0)
|   |   |   |   |   |   age > 41: YES (5.0/1.0)
|   |   |   |   |   income > 21506.2: NO (20.0)
|   |   |   |   mortgage = YES: YES (25.0/3.0)
|   |   |   save_act = YES: NO (119.0/12.0)
|   children > 0
|   |   income <= 15538.8
|   |   |   age <= 41: NO (22.0/2.0)
|   |   |   age > 41: YES (2.0)
|   |   income > 15538.8: YES (111.0/5.0)
children > 1
|   income <= 30404.3: NO (124.0/12.0)
|   income > 30404.3
|   |   children <= 2: YES (51.0/5.0)
|   |   children > 2
|   |   |   income <= 44288.3: NO (19.0/2.0)
|   |   |   income > 44288.3: YES (8.0)

Number of Leaves  :     15

Size of the tree :      29
```

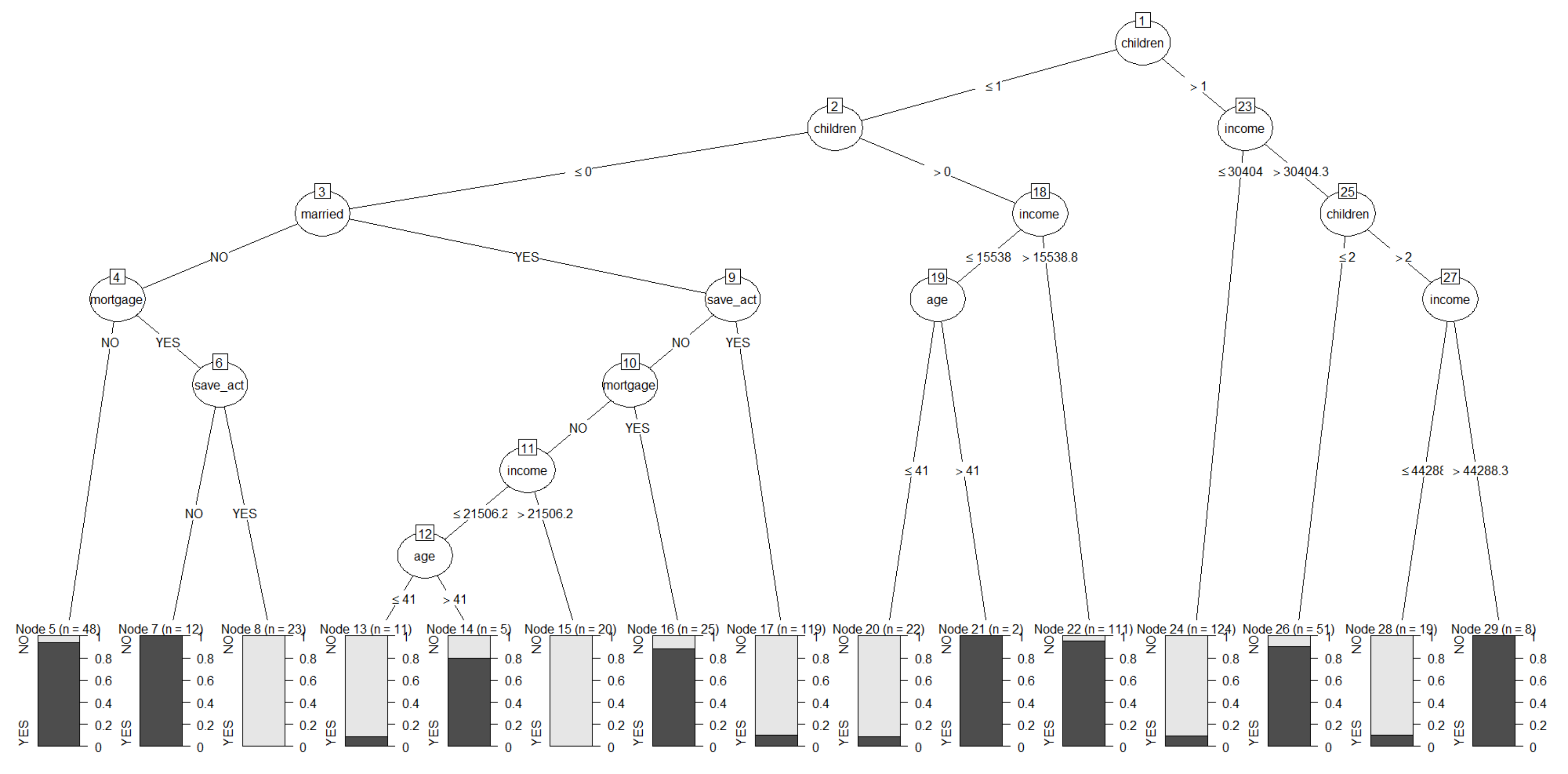
Software Use

- To export the plot, click on Export –>Save as Image



- Set the Width to 2000 and Height to 1000.
- Click on Save.
- An image of the plot is saved in the working directory.

Software Use



Outline for this Session

- Introduction to Classification
- Decision Trees
- Software Use
- **Alternative Classification Models**
- Model Evaluation and Validation
- Case Study

Alternative Classification Models

- Rule Based Methods
- Support Vector Machines
- Ensembles

Alternative Classification Models

Definition 5.3: Rule-Based Classifier

- Classify records by using a collection of **"if…then…"** rules
- Rule: (*Condition*) $\rightarrow y$
  - where
    - **Condition** is a conjunctions of attributes
    - *y* is the class label
  - **LHS**: rule antecedent or condition
  - **RHS:** rule consequent
  - Examples of classification rules:
    - (Blood Type=Warm) $\wedge$ (Lay Eggs=Yes) $\rightarrow$ Birds
    - (Taxable Income < 50K) $\wedge$ (Refund=Yes) $\rightarrow$ Evade=No

Alternative Classification Models

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
human	warm	yes	no	no	mammals
python	cold	no	no	no	reptiles
salmon	cold	no	no	yes	fishes
whale	warm	yes	no	yes	mammals
frog	cold	no	no	sometimes	amphibians
komodo	cold	no	no	no	reptiles
bat	warm	yes	yes	no	mammals
pigeon	warm	no	yes	no	birds
cat	warm	yes	no	no	mammals
leopard shark	cold	yes	no	yes	fishes
turtle	cold	no	no	sometimes	reptiles
penguin	warm	no	no	sometimes	birds
porcupine	warm	yes	no	no	mammals
eel	cold	no	no	yes	fishes
salamander	cold	no	no	sometimes	amphibians
gila monster	cold	no	no	no	reptiles
platypus	warm	no	no	no	mammals
owl	warm	no	yes	no	birds
dolphin	warm	yes	no	yes	mammals
eagle	warm	no	yes	no	birds

R1: (Give Birth = no) $\wedge$ (Can Fly = yes) $\rightarrow$ Birds

R2: (Give Birth = no) $\wedge$ (Live in Water = yes) $\rightarrow$ Fishes

R3: (Give Birth = yes) $\wedge$ (Blood Type = warm) $\rightarrow$ Mammals

R4: (Give Birth = no) $\wedge$ (Can Fly = no) $\rightarrow$ Reptiles

R5: (Live in Water = sometimes) $\rightarrow$ Amphibians

eljalao@up.edu.ph

Alternative Classification Models

- Application of Rule-Based Classifier:
  - A rule *r* **covers** an instance $\boldsymbol{x}$ if the attributes of the instance satisfy the condition of the rule

R1: (Give Birth = no) $\wedge$ (Can Fly = yes) $\rightarrow$ Birds

R2: (Give Birth = no) $\wedge$ (Live in Water = yes) $\rightarrow$ Fishes

R3: (Give Birth = yes) $\wedge$ (Blood Type = warm) $\rightarrow$ Mammals

R4: (Give Birth = no) $\wedge$ (Can Fly = no) $\rightarrow$ Reptiles

R5: (Live in Water = sometimes) $\rightarrow$ Amphibians

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
hawk	warm	no	yes	no	?
grizzly bear	warm	yes	no	no	?

The rule R1 covers a hawk => Bird

The rule R3 covers the grizzly bear => Mammal

Alternative Classification Models

- How does Rule-based Classifier Work?

  R1: (Give Birth = no) $\wedge$ (Can Fly = yes) $\rightarrow$ Birds

  R2: (Give Birth = no) $\wedge$ (Live in Water = yes) $\rightarrow$ Fishes

  R3: (Give Birth = yes) $\wedge$ (Blood Type = warm) $\rightarrow$ Mammals

  R4: (Give Birth = no) $\wedge$ (Can Fly = no) $\rightarrow$ Reptiles

  R5: (Live in Water = sometimes) $\rightarrow$ Amphibians

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
lemur	warm	yes	no	no	?
turtle	cold	no	no	sometimes	?
dogfish shark	cold	yes	no	yes	?

A lemur **triggers** rule R3, so it is classified as a mammal

A turtle **triggers** both R4 and R5

A dogfish shark **triggers none of the rules**

Alternative Classification Models

- Type the following lines of code in RStudio and run.

```
JRipModel <- JRip(pep ~ age + sex+  region  + income
                  + married + children  + car
                  + save_act+ current_act+  mortgage
                  , data=bankdata)
JRipModel
```

Alternative Classification Models

```
> JRipModel
JRIP rules:
===========

(income >= 29714.4) and (children >= 1) and (children <= 2) => pep=YES (102.0/7.0)
(children <= 1) and (save_act = NO) and (mortgage = YES) => pep=YES (46.0/7.0)
(children <= 1) and (children >= 1) and (income >= 15735.8) => pep=YES (53.0/1.0)
(married = NO) and (children <= 0) and (mortgage = NO) => pep=YES (48.0/3.0)
(children >= 1) and (income >= 45031.9) => pep=YES (8.0/0.0)
 => pep=NO (343.0/35.0)

Number of Rules : 6
```

Alternative Classification Models

- Rule Based Methods
- Support Vector Machines
- Ensembles

eljalao@up.edu.ph

Alternative Classification Models

Definition 5.4: Support Vector Machine

- A Support Vector Machine (SVM) performs classification by finding a plane that **maximizes the margin** between the two classes. The vectors (cases) that define the plane are the support vectors.

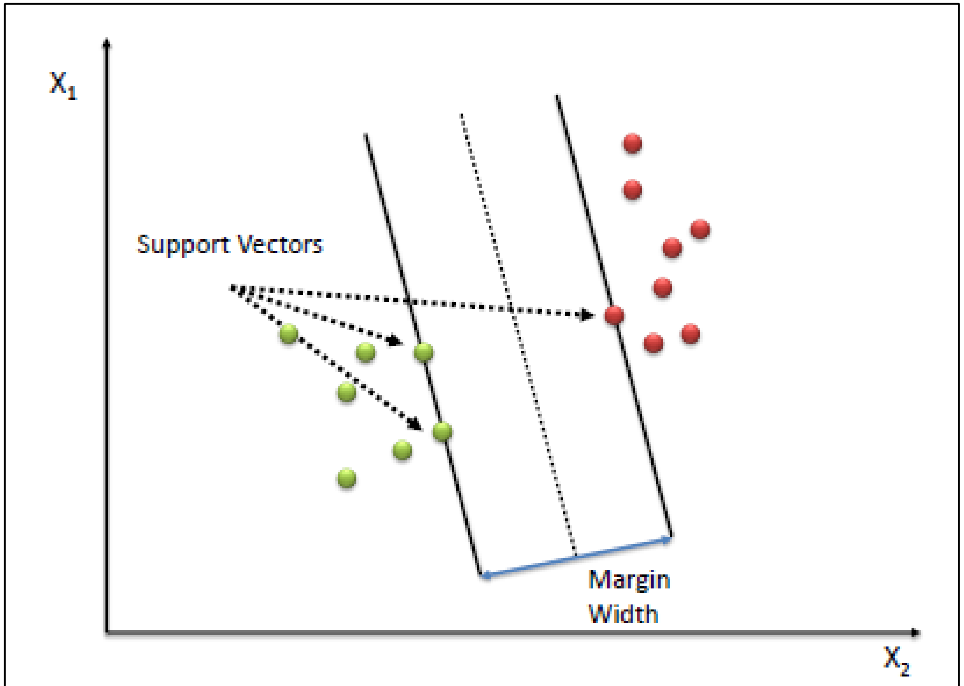


Figure 5.5: SVM

Alternative Classification Models

- SVM applications
  - SVMs were originally proposed by Boser, Guyon and Vapnik in **1992** and gained increasing popularity in late 1990s.
  - SVMs are currently among the **best performers** for a number of classification tasks ranging from text to genomic data.
  - Most popular optimization algorithms for SVMs use decomposition methodologies
  - Tuning SVMs remains a **black art**: selecting a specific kernel and parameters is usually done in a try-and-see manner.

Business Scenario: Credit Scoring

- Credit scoring is the practice of analyzing a persons background and credit application in order to assess the creditworthiness of the person
- The variables *income* (yearly), *age*, *loan* (size in euros) and *LTI*(the loan to yearly income ratio) are available.
- The goal is to devise a model which *predicts,* whether or not a default will occur within 10 years.

http://www.r-bloggers.com/using-neural-networks-for-credit-scoring-a-simple-example/

Business Scenario: Credit Scoring

- Type the following lines of code in RStudio and run.

```
creditsetnumericSMO = read.csv("creditsetnumeric.csv")
creditsetnumericSMO$default10yr =
  as.factor(creditsetnumericSMO$default10yr)
SVMModel <- SMO(default10yr ~ income + age +  loan  + LTI
                 , data=creditsetnumericSMO)
creditsettestSMO = read.csv("creditsettest.csv")
creditsettestSMO$default10yr =
  as.factor(creditsettestSMO$default10yr)
creditsettestSMO$predictions = predict(SVMModel,
                                       creditsettestSMO)
creditsettestSMO
```

Business Scenario: Credit Scoring

```
> creditsettestSMO
  income age loan   LTI default10yr predictions
1  42710  46 6104 0.143        <NA>           0
2  66953  19 8770 0.131        <NA>           1
3  24904  57   15 0.001        <NA>           0
```

Alternative Classification Models

- Rule Based Methods
- Support Vector Machines
- **Ensembles**

Alternative Classification Models

Definition 5.5: Ensembles

- Construct a **set of classifiers** from the training data
- Predict class label of previously unseen records by **aggregating predictions** made by multiple classifiers
- Wisdom of the Crowd

Alternative Classification Models

Definition 5.6: Parallel Ensembles

- Parallel: Combines approximately **independent**, diverse base learners
  - Different learners should make **different errors**
  - **Ensemble can outperform** any one of its components
  - **Variance reduction method** useful for unstable, high-variance learners (such as trees)
  - Bagging, Random Forest (RF) examples

Alternative Classification Models

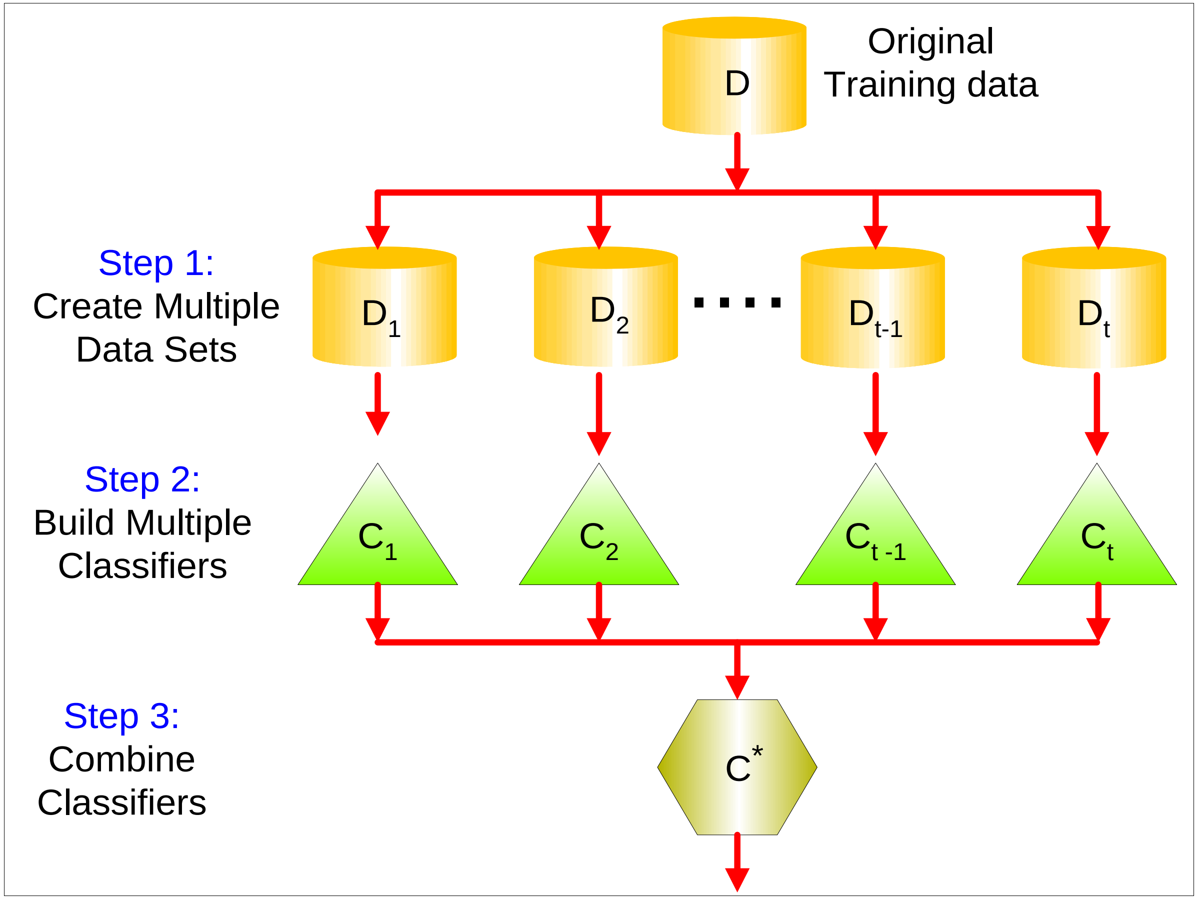


Figure 5.5: Parallel Ensemble

Alternative Classification Models

Definition 5.7: Random Forests

- The random forest (Breiman, 2001) is an **ensemble of decision trees**
- Trees are **combined** by average (regression) or voting (classification)
- RF injects additional **randomness**
- Averaging minimizes **overfitting** (no prunning)
- Tree provides a class probability estimates so that **weighted votes** can be used

Alternative Classification Models

- The Random Forests Algorithm
  - For some number of trees T
  - Sample N cases at random with replacement to create a subset of the data at each node:
    - For some number p, p predictor variables are selected at random from all the predictor variables.
    - The predictor variable that provides the best split, according to some objective function, is used to do a binary split on that node.
    - At the next node, choose another p variables at random from all predictor variables and do the same.

Alternative Classification Models

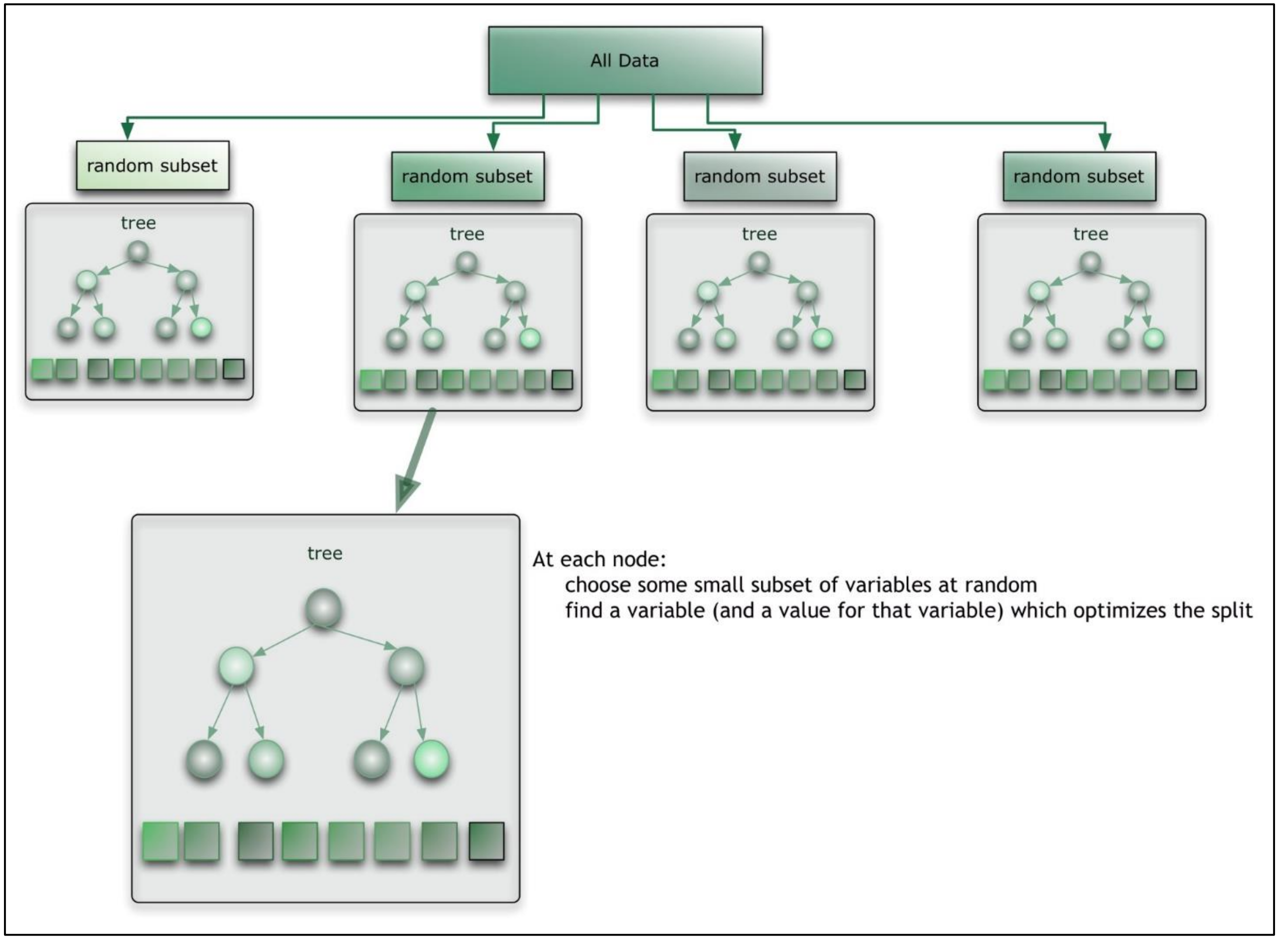


Figure 5.6: Random Forest Model

Alternative Classification Models

- Selecting p variables randomly reduces correlation between trees
  - Random splitter selection: $p = 1$
  - Breiman's bagger: p = total number of predictor variables
  - Random forest: p << number of predictor variables. Brieman suggests three possible values for m: ½$\sqrt{p}$, $\sqrt{p}$, and $2\sqrt{p}$

Business Scenario: Income Data

- Type the following lines of code in RStudio and run.

```
adult = read.csv("adult.csv")
RF <- make_Weka_classifier("weka/classifiers/trees/RandomForest")
RFModel <- RF(income ~ age+workclass
              +finalweight+education+education.num
              +marital.status+occupation+relationship
              +race+sex+capitalgain+capitalloss+hoursperweek
              +country,
              data=adult, control = Weka_control(K =1))
adult$predictions = predict(RFModel, adult, se.fit=TRUE)
adult[1:10,]
```

Some Notes on Choosing Classification Models

- Need to go back to the initial problem/objective
- If problem is to **find rules for data** summarization
  - Use Decision Trees, Rule Classifiers
  - Weakness: Relative weak predictive power
- If problem is to **predict accurately**
  - Use SVM or Random Forests
  - Weakness: Black box

Outline for this Session

- Introduction to Classification
- Decision Trees
- Software Use
- Alternative Classification Models
- **Model Evaluation and Validation**
- Case Study

Model Evaluation and Validation

Definition 5.8: Model Evaluation

- Model Evaluation is a methodology that helps to find the best model that represents our data and how well the chosen model will **work in the future.**

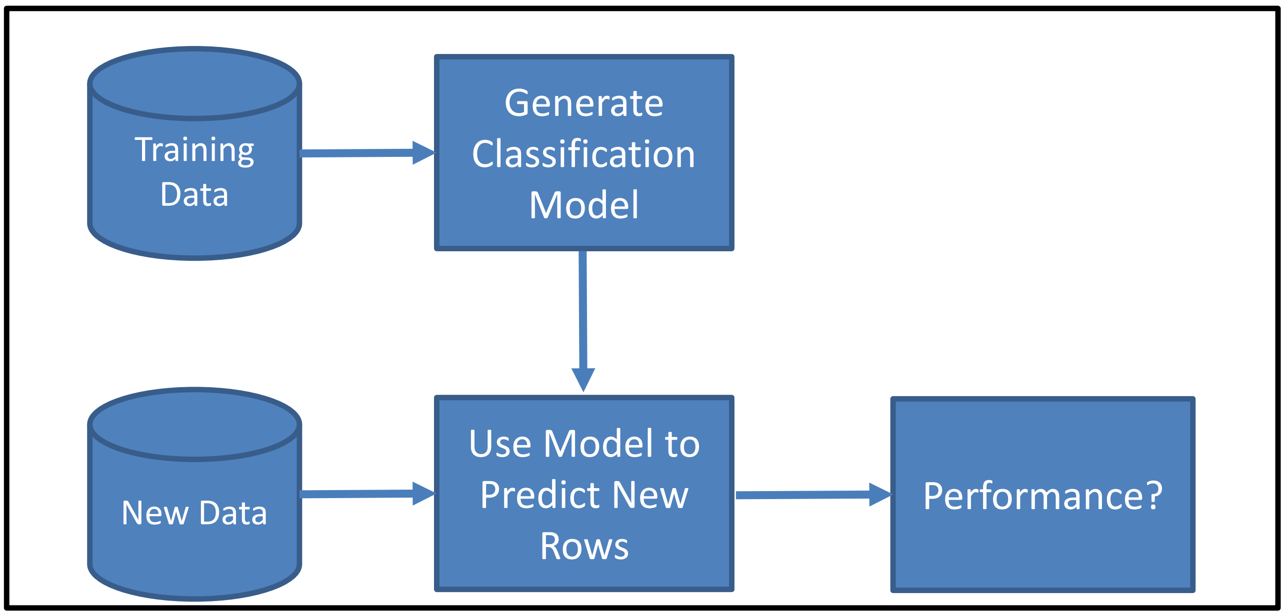


Figure 5.7: Model Evaluation Workflow

eljalao@up.edu.ph

Model Evaluation and Validation

Definition 5.9: Error

- Error: if predicted class is not equal to actual class

$$e(t) = 1 \; if \; f(x) \neq y \tag{5.1}$$

- Re-substitution errors:
  - error on training data
- Generalization errors:
  - error on unseen data

Model Evaluation and Validation

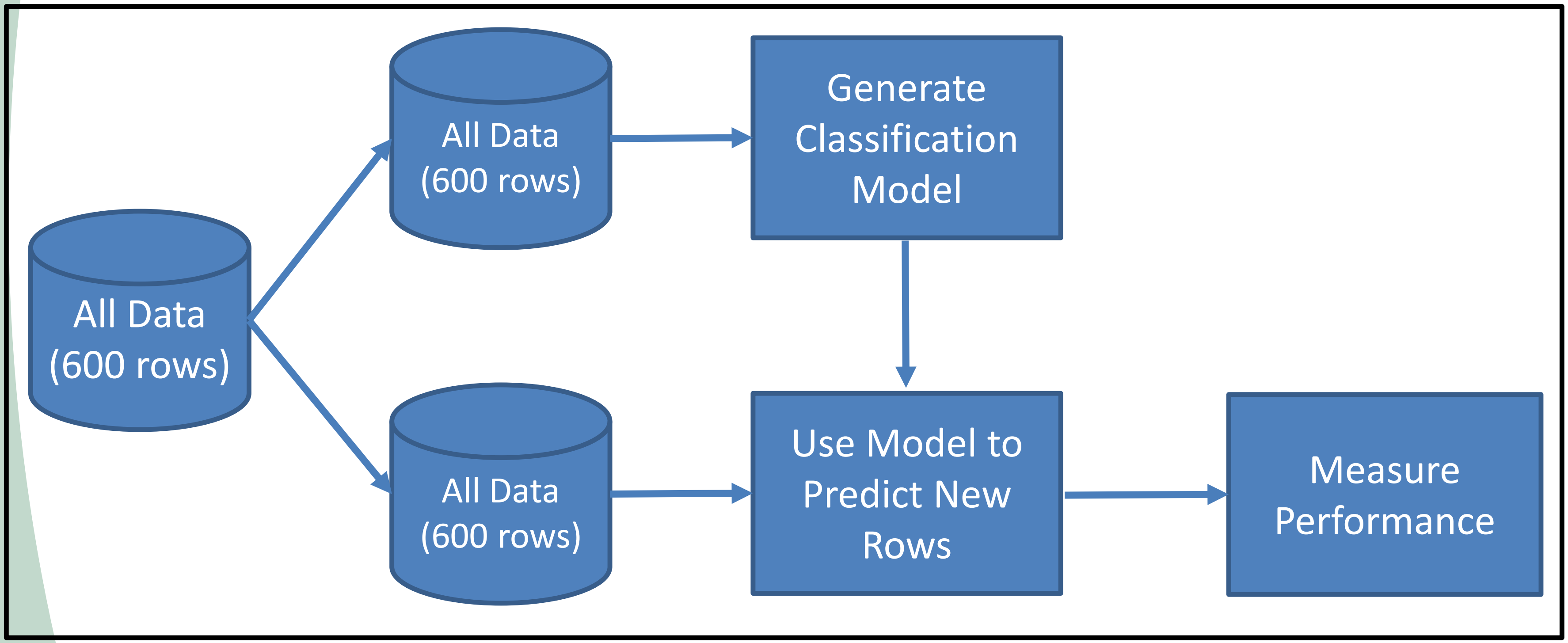


Figure 5.8: Re-Substitution Errors

Model Evaluation and Validation

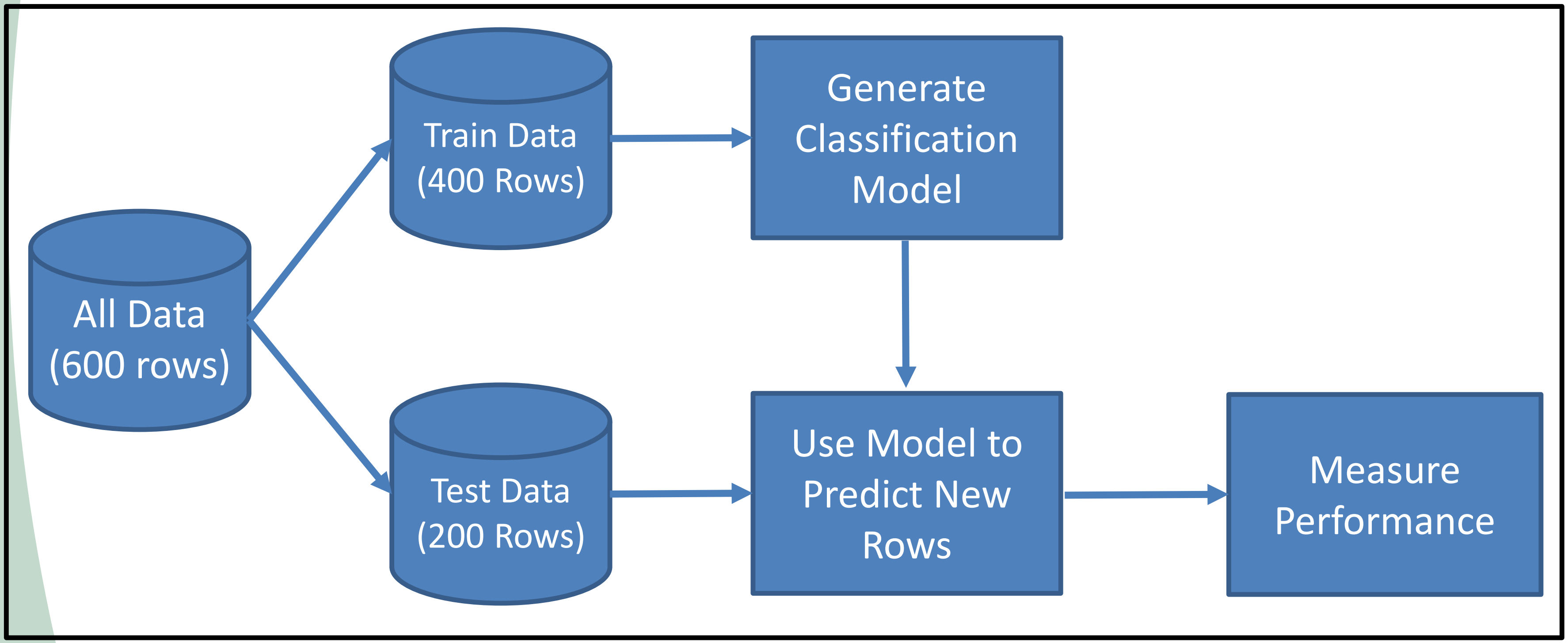


Figure 5.9: Generalization Errors

Model Evaluation and Validation

- **Metrics for Performance Evaluation**
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?

Metrics for Performance Evaluation

- Focus on the predictive capability of a model
  - Rather than how fast it takes to classify or build models, scalability, etc.
- Confusion Matrix:

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

Figure 5.15: Confusion Matrix

Metrics for Performance Evaluation

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- Most widely-used metric:

$$Accuracy = \frac{a + d}{a + b + c + d} \qquad (5.2)$$

eljalao@up.edu.ph

Metrics for Performance Evaluation

- Limitation of Accuracy
  - Consider a **2-class problem**
    - Number of Class 0 examples = 9990
    - Number of Class 1 examples = 10
  - If model predicts everything to be class 0, accuracy is 9990/10000 = 99.9 %
    - Accuracy is **misleading** because model does not detect any class 1 example

Metrics for Performance Evaluation

	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	**Class=Yes**	**Class=No**
	Class=Yes	C(Yes|Yes)	C(No|Yes)
	Class=No	C(Yes|No)	C(No|No)

C(i|j): Cost of misclassifying class j example as class i

Figure 5.16: Cost Matrix

Metrics for Performance Evaluation

Example 5.4: Cost Example

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	+	-
	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	150	40
	-	60	250

Accuracy = 80%

Cost = 3910

Model M_2	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	250	45
	-	5	200

Accuracy = 90%

Cost = 4255

Model Evaluation

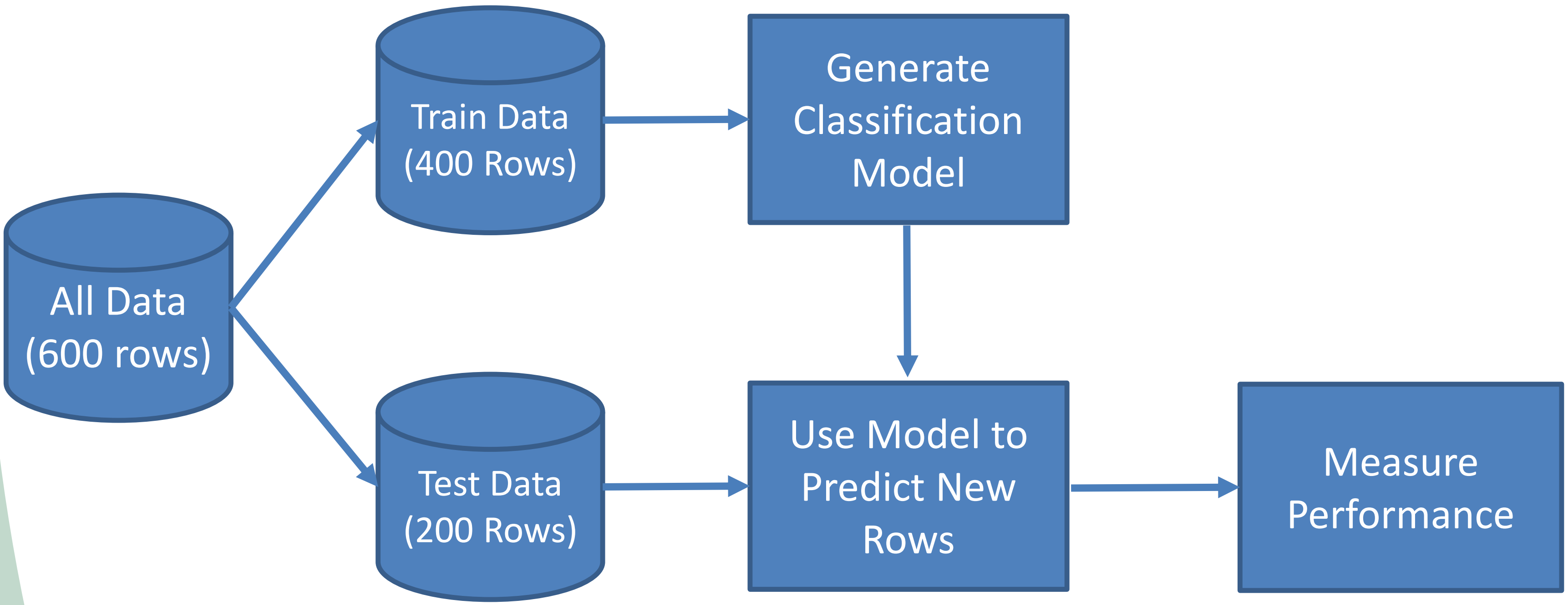
- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- **Methods for Performance Evaluation**
  - How to obtain reliable estimates?

Methods of Performance Evaluation

- Holdout
  - **Reserve** 2/3 for training and 1/3 for testing

Methods of Performance Evaluation

- Generalization errors: error on unseen data



Methods of Performance Evaluation

- Type the following lines of code in RStudio and run.

```
bankdata = read.csv("bankdata.csv")
sample <- floor(0.67 * nrow(bankdata))
set.seed(123)
train_ind <- sample(seq_len(nrow(bankdata)),
                    size = sample)
bankdatatrain <- bankdata[train_ind, ]
bankdatatest <- bankdata[-train_ind, ]
J48ModelHoldout <- J48(pep ~ age + sex+ region  + income
                + married + children  + car
                + save_act+ current_act+  mortgage
                , data=bankdatatrain)
evaluate_Weka_classifier(J48ModelHoldout,
                         newdata = bankdatatest)
```

Methods of Performance Evaluation

```
> evaluate_Weka_classifier(J48ModelHoldout,
+                          newdata = bankdatatest)

=== Summary ===

Correctly Classified Instances         177               89.3939 %
Incorrectly Classified Instances        21               10.6061 %
Kappa statistic                          0.7812
Mean absolute error                      0.1677
Root mean squared error                  0.3201
Relative absolute error                 34.3219 %
Root relative squared error             64.7738 %
Total Number of Instances              198

=== Confusion Matrix ===

   a   b   <-- classified as
 106   8 |   a = NO
  13  71 |   b = YES
```

Outline for this Session

- Introduction to Classification
- Decision Trees
- Software Use
- Alternative Classification Models
- Model Evaluation and Validation
- **Case Study**

Case Study

- ChurnData.xlsx
- Generate a Classification Process that can predict if the customer will churn(y) or not (n)
- Identify business rules that to profile subscribers to **minimize churn.**

Outline for this Session

- Introduction to Classification
- Decision Trees
- Software Use
- Alternative Classification Models
- Model Evaluation and Validation
- Case Study

References

- Tan et al. Intro to Data Mining Notes
- Runger, G. IEE 520 notes
- C4.5 Data: UCI Data Repository