

# CS 412 Intro. to Data Mining

## Chapter 8. Classification: Basic Concepts

Jiawei Han, Computer Science, Univ. Illinois at Urbana-Champaign, 2017

ມະນີ ດາຕ່າ ມິນ ນັ້ນ ຂອງ ອະນຸຍາ ຮູ່ ຖະນົາ ສຳເນົາ ກົດລົງ 1,2  
① ②

ການຮັບສິນຂອງ  
ພົມພັນ

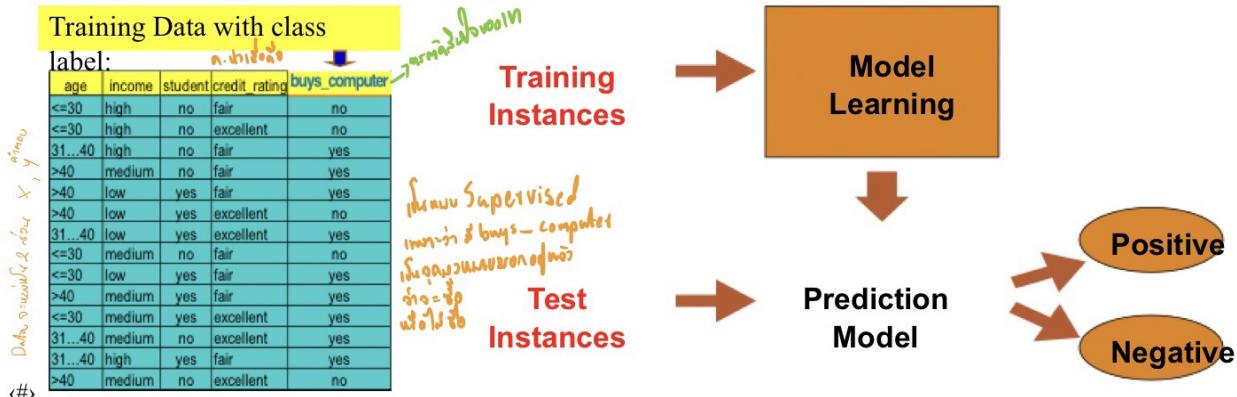
## Supervised vs. Unsupervised Learning (1)

ສິນພາດລາຍໄປລົງສອນ

- Supervised learning (classification)
  - Supervision: The training data such as observations or measurements are accompanied by **labels** indicating the classes which they belong to
  - New data is classified based on the models built from the training set

[In Binary]

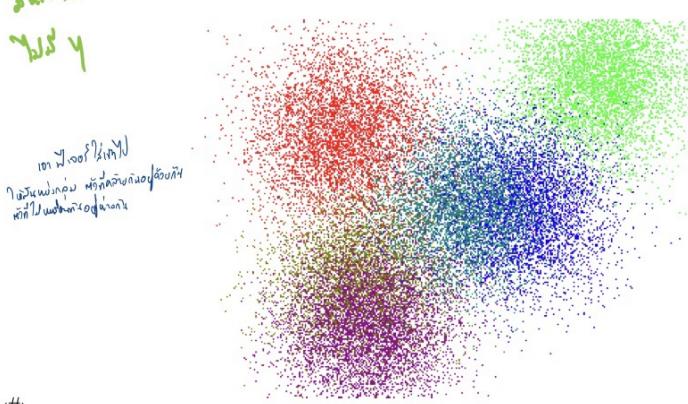
ຊື່ລົງທຶນທີ່ໃຫຍ່



## Supervised vs. Unsupervised Learning (2)

ປັບປຸງ ກົດໄວ້ມີຄວາມ ນາໂລ ລາຍລະອຽດ

- Unsupervised learning (clustering)
  - The class labels of training data are unknown
  - Given a set of observations or measurements, establish the possible existence of classes or clusters in the data



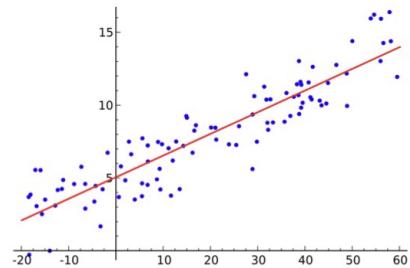
# Prediction Problems: Classification vs. Numeric Prediction

- **Classification**
  - Predict categorical class labels (discrete or nominal)
  - Construct a model based on the training set and the **class labels** (the values in a classifying attribute) and use it in classifying new data
- **Numeric prediction**
  - Model continuous-valued functions (i.e., predict unknown or missing values)

இனாலும் ஒரு வகையில்  
ஏவும் Regression

- Typical applications of classification

- Credit/loan approval
- Medical diagnosis: if a tumor is cancerous or benign
- Fraud detection: if a transaction is fraudulent
- Web page categorization: which category it is



## Classification—Model Construction, Validation and Testing

- **Model construction:** இன் படி நிர்ணயித்து மதிப்பீடு செய்யப்படும் படி என்று கூறப்படுகிறது.
  - Each sample is assumed to belong to a predefined class (shown by the **class label**)
  - The set of samples used for model construction is **training set**
  - Model: Represented as decision trees, rules, mathematical formulas, or other forms
- **Model Validation and Testing:** வெளியே விடப்படும் படி நிர்ணயித்து மதிப்பீடு செய்யப்படும் படி என்று கூறப்படுகிறது.
  - **Test:** Estimate accuracy of the model
    - The known label of test sample is compared with the classified result from the model
    - *Accuracy:* % of test set samples that are correctly classified by the model
    - Test set is independent of training set
  - **Validation:** If the test set is used to select or refine models, it is called **validation** (or development) (**test**) set
- **Model Deployment:** If the accuracy is acceptable, use the model to classify new data

## Chapter 8. Classification: Basic Concepts

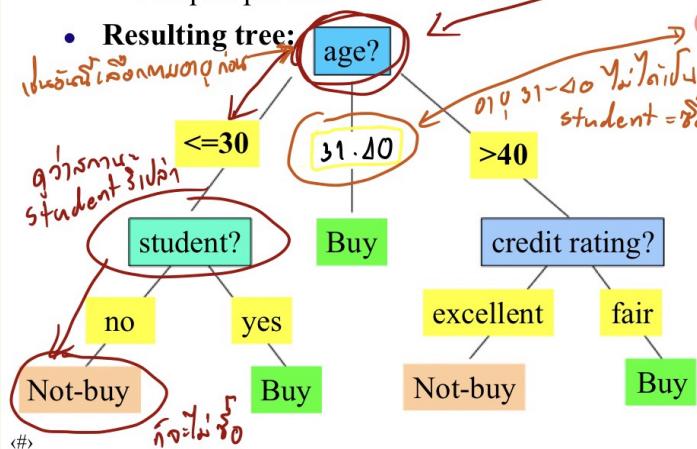
- Classification: Basic Concepts
- Decision Tree Induction திட்டமிழுப்பு
- Bayes Classification Methods
- Linear Classifier
- Model Evaluation and Selection
- Techniques to Improve Classification Accuracy: Ensemble Methods
- Additional Concepts on Classification
- Summary

# Decision Tree Induction: An Example

- Decision tree construction:**

- A top-down, recursive, divide-and-conquer process

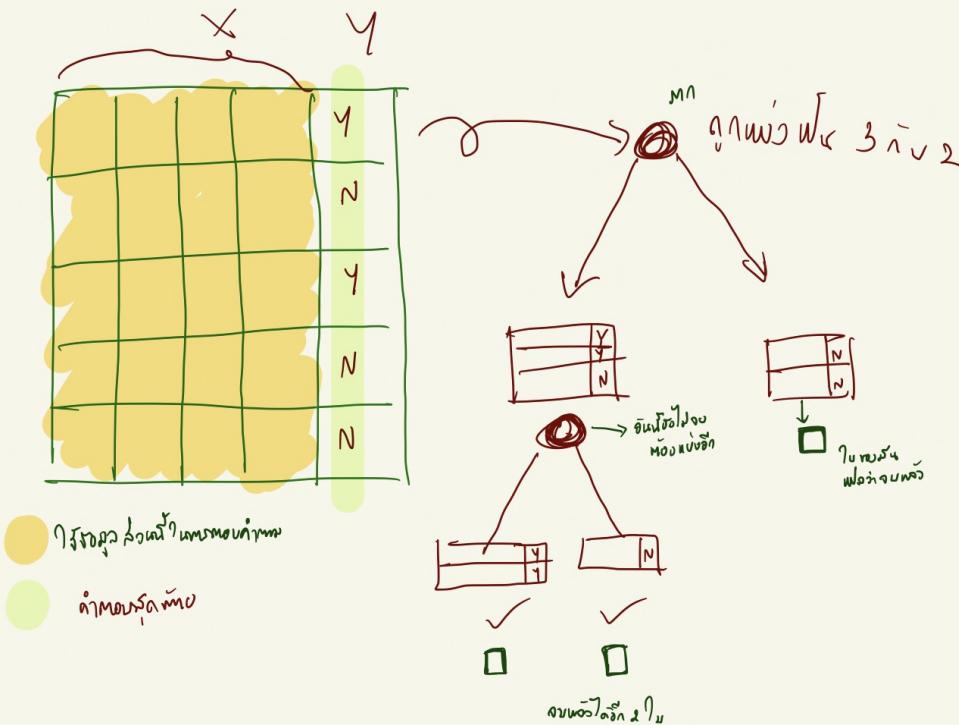
- Resulting tree:**



Training data set: Who buys computer?

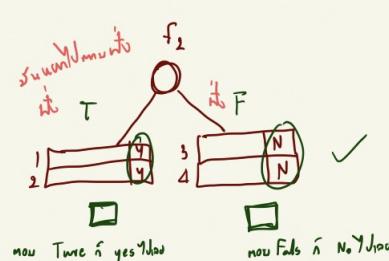
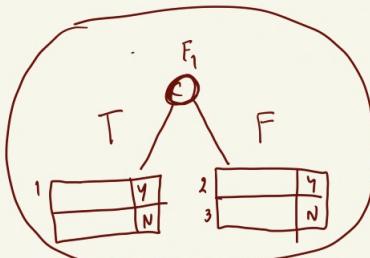
age	income	student	credit rating	buys computer
<=30	high	no	fair	no
<=30	high	no	excellent	yes
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

Note: The data set is adapted from "Playing Tennis" example of R. Quinlan



$F_1$	$F_2$	$F_3$	$y$
T	T	F	Y
F	T	F	Y
F	F	F	N
T	F	T	N

ក្នុងនេះ នឹងតាមលក្ខណៈទីនេះ និងលក្ខណៈទាំងអស់  
រាយការណ៍ នឹងតាមលក្ខណៈទីនេះ និងលក្ខណៈទាំងអស់



និង Association Rule នឹងមែន:

ឱ្យ Data មែន = ក្នុង Y

## Information Gain: An Attribute Selection Measure

- Select the attribute with the highest information gain (used in typical decision tree induction algorithm: ID3/C4.5)
- Let  $p_i$  be the probability that an arbitrary tuple in D belongs to class  $C_i$ , estimated by  $|C_i, D|/|D|$
- Expected information (entropy) needed to classify a tuple in D:

$$Info(D) = - \sum_{i=1}^k p_i \log_2(p_i)$$

- Information needed (after using A to split D into v partitions) to classify D:

$$Info_A(D) = \sum_{j=1}^v \frac{|D_j|}{|D|} \times Info(D_j)$$

- Information gained by branching on attribute A

$$Gain(A) = Info(D) - Info_A(D)$$

#

$$I(A, B, C) = - \frac{4}{9} \log_2 \frac{4}{9} - \frac{3}{9} \log_2 \frac{3}{9} - \frac{2}{9} \log_2 \frac{2}{9}$$

### Example: Attribute Selection with Information Gain

• Class P: buys_computer = "yes"	$Info(D) = I(9,5) = -\frac{9}{14} \log_2 \left(\frac{9}{14}\right) - \frac{5}{14} \log_2 \left(\frac{5}{14}\right) = 0.940$
• Class N: buys_computer = "no"	$I(p_i, n_i)$
	$\begin{array}{ c c c c } \hline age & p_i & n_i & I(p_i, n_i) \\ \hline \leq 30 & 2 & 3 & 0.971 \\ \hline 31...40 & 4 & 0 & 0 \\ \hline > 40 & 3 & 2 & 0.971 \\ \hline \end{array}$
	$\leq 30 = 5 \text{ no}$
	$\frac{5}{14}$
	$\begin{array}{ c c c c c c } \hline age & income & student & credit\_rating & buys\_computer \\ \hline \leq 30 & high & no & fair & no \\ \hline \leq 30 & high & no & excellent & no \\ \hline 31...40 & high & no & fair & yes \\ \hline > 40 & medium & no & fair & yes \\ \hline > 40 & low & yes & fair & yes \\ \hline > 40 & low & yes & excellent & no \\ \hline > 40 & medium & no & fair & no \\ \hline < 30 & low & yes & fair & yes \\ \hline < 30 & medium & yes & excellent & yes \\ \hline < 30 & medium & no & excellent & yes \\ \hline 31...40 & medium & yes & fair & yes \\ \hline 31...40 & high & yes & fair & yes \\ \hline > 40 & medium & no & excellent & yes \\ \hline > 40 & medium & yes & excellent & yes \\ \hline \end{array}$
	$\leq 30 = 5 \text{ yes}$
	$\frac{5}{14}$

$$Info_{age}(D) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0)$$

$$> 40 + \frac{5}{14} I(3,2) = 0.694$$

$\frac{5}{14} I(2,3)$  means "age  $\leq 30$ " has 5 out of 14 samples, with 2 yes's and 3 no's.

Hence

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

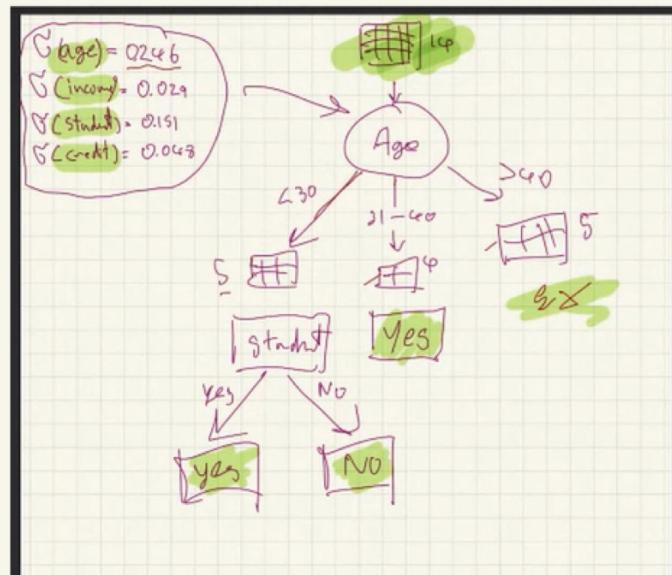
Similarly, we can get

$$Gain(income) = 0.029$$

$$Gain(student) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

#



$$Info(D) = I(2,3) = -\frac{2}{5} \log_2 \frac{2}{5} - \frac{3}{5} \log_2 \frac{3}{5}$$

$$Info_{income}(D) = \frac{2}{5} I(0,2) + \frac{2}{5} I(1,1) + \frac{1}{5} I(1,0)$$

$$Info_{student}(D) = \frac{2}{5} I(2,0) + \frac{3}{5} I(0,2)$$

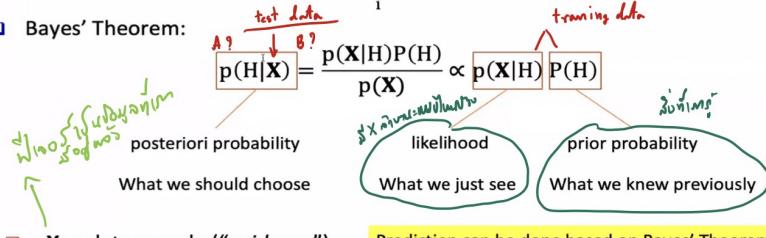
$$Info_{credit\_rating}(D) = \frac{3}{5} I(1,2) + \frac{2}{5} I(1,1)$$

## Bayes' Theorem: Basics

- Total probability Theorem:

$$p(B) = \sum_i p(B|A_i)p(A_i)$$

- Bayes' Theorem:



□ X: a data sample ("evidence")

Prediction can be done based on Bayes' Theorem:

Classification is to derive the maximum posteriori

Thanapong I. ...

## Naïve Bayes Classifier: Training Dataset

Class:

C1:buys\_computer = 'yes'

C2:buys\_computer = 'no'

Data to be classified:

X = (age <=30, Income = medium,  
Student = yes, Credit\_rating =  
Fair)

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

Training Data

4#

## Naïve Bayes Classifier: Training Dataset

Class:

C1:buys\_computer = 'yes'

C2:buys\_computer = 'no'

Data to be classified:

X = (age <=30, Income = medium,  
Student = yes, Credit\_rating = Fair)

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

Training Data

29

$$P^y = P(b^y) P(a^{30}|b^y) P(i^{medium}|b^y) P(s^y|b^y) P(c^y|b^y)$$

$$\begin{aligned} \frac{P(H^{-Y} | x')}{P(H^{-N} | x')} &=? \\ \frac{P(x | H^{-Y}) P(H^{-Y})}{P(x | H^{-N}) P(H^{-N})} &=? \\ &= P(x | H^{-Y}) P(H^{-Y}) \end{aligned}$$

training Data

## Naïve Bayes Classifier: An Example

- P(Ci): P(buys\_computer = "yes") = 9/14 = 0.643  
P(buys\_computer = "no") = 5/14 = 0.357
- Compute P(X|Ci) for each class
  - P(age = "<= 30" | buys\_computer = "yes") = 2/9 = 0.222  
P(age = "<= 30" | buys\_computer = "no") = 3/5 = 0.6
  - P(income = "medium" | buys\_computer = "yes") = 4/9 = 0.444  
P(income = "medium" | buys\_computer = "no") = 2/5 = 0.4
  - P(student = "yes" | buys\_computer = "yes") = 6/9 = 0.667  
P(student = "yes" | buys\_computer = "no") = 1/5 = 0.2
  - P(credit\_rating = "fair" | buys\_computer = "yes") = 6/9 = 0.667  
P(credit\_rating = "fair" | buys\_computer = "no") = 2/5 = 0.4

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

- X = (age <= 30, income = medium, student = yes, credit\_rating = fair)

$$P(X|Ci) : P(X|buys\_computer = "yes") = 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044$$

$$P(X|buys\_computer = "no") = 0.6 \times 0.4 \times 0.2 \times 0.4 = 0.019$$

$$P(X|Ci)*P(Ci) : P(X|buys\_computer = "yes") * P(buys\_computer = "yes") = 0.028$$

$$P(X|buys\_computer = "no") * P(buys\_computer = "no") = 0.007$$

Therefore, X belongs to class ("buys\_computer = yes")

$$\frac{3}{9} \times \frac{6}{9} \times \frac{9}{14} = 0.33$$

$$\hat{X} = \text{age} = 42, \text{student} = \text{yes} ?$$

$$P(H_n | \hat{X}) = ?$$

$$P(H_n | (\text{age} = 42, \text{student} = \text{yes})) = P(\text{age} = 42 | H_n) P(\text{student} = \text{yes} | H_n) P(H_n)$$

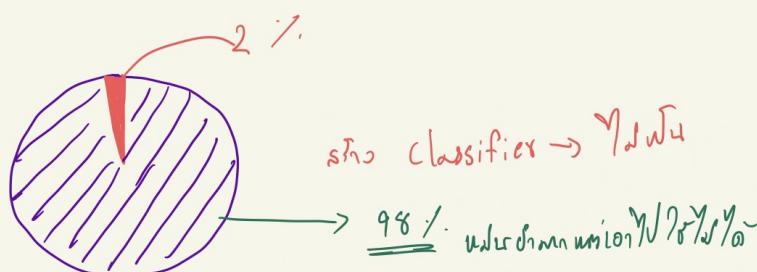
$$\frac{3}{9} \times \frac{6}{9} \times \frac{9}{14}$$

$$P(H_{\text{buy}=\text{N}} | (\text{age} = 42, \text{student} = \text{yes})) =$$

# Model Evaluation and Selection

- Evaluation metrics
  - How can we measure accuracy?
  - Other metrics to consider?
- Use **validation test set** of class-labeled tuples instead of training set when assessing accuracy
- Methods for estimating a classifier's accuracy
  - Holdout method
  - Cross-validation
  - Bootstrap
- Comparing classifiers:
  - ROC Curves

#



47

## Classifier Evaluation Metrics: Precision and Recall, and F-measures

- ❑ **Precision:** Exactness: what % of tuples that the classifier labeled as positive are actually positive?  

$$P = \text{Precision} = \frac{TP}{TP + FP}$$

in Model  $\frac{1}{\text{Pos}}$   
Cada vez que
- ❑ **Recall:** Completeness: what % of positive tuples did the classifier label as positive?
  - ❑ Range:  $[0, 1]$
  - ❑ The “inverse” relationship between precision & recall
  - ❑ **F measure (or F-score):** harmonic mean of precision and recall
    - ❑ In general, it is the weighted measure of precision & recall

$$F_\beta = \frac{1}{\alpha \cdot \frac{1}{P} + (1 - \alpha) \cdot \frac{1}{R}} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

Assigning  $\beta$  times as much weight to recall as to precision)

- ❑ **F1-measure (balanced F-measure)**
  - ❑ That is, when  $\beta = 1$ ,

$$F_1 = \frac{2PR}{P + R}$$

48

precision

$$\frac{O}{O+D}$$

မြတ်လုပ်

Recall

sensitivity  
specificity

$$\frac{O}{O+D} = \frac{98}{98+2} = 0.98 \approx 100\%$$

	P	N	
P	TP	FP	TP + FP
N	FN	TN	FN + TN
O	98	2	100
1	2	98	100

ကိစ္စပါဒ်များ၏ မှန်ချက်

	0	1	
0	30	6	FN
1	2	48	TN
FP			

ကိစ္စပါဒ်၏ မှန်ချက်

O = Malign

1 = Benign

ကိစ္စပါဒ်၏ အားလုံး

	0	1	precision	recall	f1-score	support
0	0.94	0.89	$\frac{44}{44+6}$	$\frac{30}{30+2}$	$\frac{44}{44+6}$	36
1	0.83	0.96	$\frac{44}{44+2}$	$\frac{30}{30+6}$	$\frac{44}{44+6}$	50
accuracy						86
macro avg	0.91	0.90	$\frac{0.94+0.89}{2}$	$\frac{0.83+0.96}{2}$	$\frac{0.91+0.90}{2}$	86
weighted avg	0.91	0.91	$\frac{0.94*36}{86}$	$\frac{0.96*50}{86}$	$\frac{0.91*(36+50)}{86}$	86

$$\text{Sens} = \frac{\text{no pos}}{\text{no pos} + \text{no neg}} = 0.83$$

$$\text{Spec} = \frac{\text{no Neg}}{\text{no pos} + \text{no neg}}$$

$$\left( \frac{44}{44+2} \right)$$

$$\frac{(0.83*36) + (0.96*50)}{36+50}$$

**Confusion Matrix:**

ມີຕົວຢ່າງນີ້ໃຫຍ່ຈີນ  
no Positives Minus  
no = m<sub>i</sub> N<sub>class</sub> ດັນນີ້ຈະ

Actual class\Predicted class	C <sub>1</sub>	¬ C <sub>1</sub>
C <sub>1</sub>	True Positives (TP)	False Negatives (FN)
¬ C <sub>1</sub>	False Positives (FP)	True Negatives (TN)

- In a confusion matrix w.  $m$  classes,  $CM_{i,j}$  indicates # of tuples in class  $i$  that were labeled by the classifier as class  $j$

- May have extra rows/columns to provide totals

**Example of Confusion Matrix:**

/Positives /Negatives

Actual class\Predicted class	buy_computer = yes	buy_computer = no	Total
buy_computer = yes	6954	46	7000
buy_computer = no	412	2588	3000
Total	7366	2634	10000

**Confusion Matrix:**

ມີຕົວຢ່າງນີ້ໃຫຍ່ຈີນ  
no Positives Minus  
no = m<sub>i</sub> N<sub>class</sub> ດັນນີ້ຈະ

Actual class\Predicted class	C <sub>1</sub>	¬ C <sub>1</sub>
C <sub>1</sub>	True Positives (TP)	False Negatives (FN)
¬ C <sub>1</sub>	False Positives (FP)	True Negatives (TN)

- In a confusion matrix w.  $m$  classes,  $CM_{i,j}$  indicates # of tuples in class  $i$  that were labeled by the classifier as class  $j$

- May have extra rows/columns to provide totals

**Example of Confusion Matrix:**

Actual class\Predicted class	buy_computer = yes	buy_computer = no	Total
buy_computer = yes	6954	46	7000
buy_computer = no	412	2588	3000
Total	7366	2634	10000

# Classifier Evaluation Metrics: Accuracy, Error Rate, Sensitivity and Specificity

A\P	C	$\neg C$	
C	TP	FN	P
$\neg C$	FP	TN	N
	P'	N'	All

- **Classifier accuracy**, or recognition rate
- Percentage of test set tuples that are correctly classified  

$$\text{Accuracy} = \frac{(TP + TN)}{\text{All}}$$
- **Error rate**:  $1 - \text{accuracy}$ , or  

$$\text{Error rate} = \frac{(FP + FN)}{\text{All}}$$

- **Class imbalance problem**
  - One class may be *rare*
  - E.g., fraud, or HIV-positive
  - Significant *majority of the negative class* and minority of the positive class
  - Measures handle the class imbalance problem
- **Sensitivity** (recall): True positive recognition rate  

$$\text{Sensitivity} = \frac{\text{TP}}{\text{P}}$$
  - Sensitivity =  $\frac{\text{TP}}{\text{P}}$
- **Specificity**: True negative recognition rate  

$$\text{Specificity} = \frac{\text{TN}}{\text{N}}$$
  - Specificity =  $\frac{\text{TN}}{\text{N}}$

## Classifier Evaluation Metrics: Precision and Recall, and F-measures

- **Precision**: Exactness: what % of tuples that the classifier labeled as positive are actually positive?  

$$P = \text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$
- **Recall**: Completeness: what % of positive tuples did the classifier label as positive?  

$$R = \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$
- Range:  $[0, 1]$
- The “inverse” relationship between precision & recall
- **F measure (or F-score)**: harmonic mean of precision and recall
- In general, it is the weighted measure of precision & recall

### □ **F1-measure (balanced F-measure)**

- That is, when  $\beta = 1$ ,

$$F_1 = \frac{2PR}{P + R}$$

Model must have pos. & neg. classes

most pos. & neg. classes