Classifier Accuracy Measures

	C_1	C_2
C_1	True positive	False negative
C_2	False positive	True negative

classes	buy_computer = yes	buy_computer = no	total	recognition(%)
buy_computer = yes	6954	46	7000	99.34
buy_computer = no	412	2588	3000	86.27
total	7366	2634	10000	95.52

- Accuracy of a classifier M, acc(M): percentage of test set tuples that are correctly classified by the model M
  - Error rate (misclassification rate) of M = 1 – acc(M)=1-95.411=4.589
  - Given *m* classes, $CM_{i,j}$, an entry in a **confusion matrix**, indicates # of tuples in class *i* that are labeled by the classifier as class *j*
- Alternative accuracy measures (e.g., for cancer diagnosis)

  sensitivity = t-pos/pos /* true positive recognition rate */

  specificity = t-neg/neg /* true negative recognition rate */

  precision = t-pos/(t-pos + f-pos)=94.44

  accuracy = sensitivity * pos/(pos + neg) + specificity * neg/(pos + neg)

  =99.34*7000/10000+86.27*3000/10000=69.53+25.88=95.411

  - This model can also be used for cost-benefit analysis

Predictor Error Measures

- Measure predictor accuracy: measure how far off the predicted value is from the actual known value
- **Loss function**: measures the error betw. y_i and the predicted value y_i'
  - Absolute error: $| y_i - y_i' |$
  - Squared error: $(y_i - y_i')^2$
- Test error (generalization error): the average loss over the test set
  - Mean absolute error: $\frac{\sum_{i=1}^{d} | y_i - y_i' |}{d}$ Mean squared error: $\frac{\sum_{i=1}^{d} (y_i - y_i')^2}{d}$
  - Relative absolute error: $\frac{\sum_{i=1}^{d} | y_i - y_i' |}{\sum_{i=1}^{d} | y_i - \bar{y} |}$ Relative squared error: $\frac{\sum_{i=1}^{d} (y_i - y_i')^2}{\sum_{i=1}^{d} (y_i - \bar{y})^2}$

The mean squared-error exaggerates the presence of outliers

Popularly use (square) root mean-square error, similarly, root relative squared error

CLASSIFICATION METHODS

Example: The following table 1 gives the profile of customers (Refund, Marital Status & Taxable Income) who has taken loan from a bank. The table also shows how many of them really cheated the bank.

1. Can you develop a decision rule to classify the customer as whether they will cheat or not based on the value of 3 attributes (Refund, Marital Status & Taxable Income)
2. Validate the model using the test data given in table 2

Table 2: Test Data

SL No	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Married	> 80 K	No
2	No	Single	> 80 K	No
3	No	Single	< 80 K	No
4	No	Married	> 80 K	No
5	No	Divorced	> 80 K	Yes

CLASSIFICATION METHODS

Table 1: Training Data Set

SL No	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	> 80 K	No
2	No	Married	> 80 K	No
3	No	Single	< 80 K	No
4	Yes	Married	> 80 K	No
5	No	Divorced	> 80 K	Yes
6	No	Married	< 80 K	No
7	Yes	Divorced	> 80 K	No
8	No	Single	> 80 K	Yes
9	No	Married	> 80 K	No
10	No	Single	> 80 K	Yes

Class variable: Cheat

Number of predefined classes: 2 (Cheat = No & Cheat = Yes)

CLASSIFICATION METHODS

Example:Result

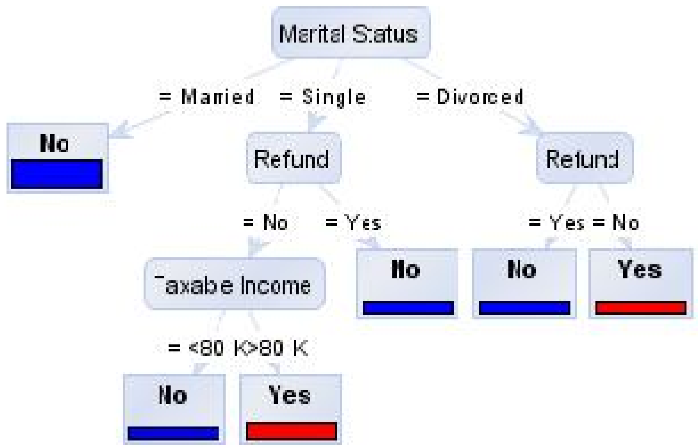
If Marital Status = Married then cheat : No

If Marital Status = Single & Refund = Yes then cheat : No
If Marital Status = Single, Refund = No & Taxable Income < 80K then cheat: No
If Marital Status = Single, Refund = No & Taxable Income > 80K then cheat: Yes
If Marital Status = Divorced & Refund = Yes then cheat : No
If Marital Status = Divorced & Refund = No then cheat : Yes

CLASSIFICATION METHODS

Example: Decision Tree

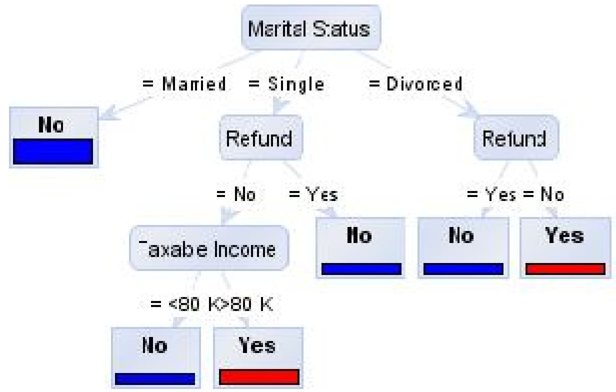
SL No	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	> 80 K	No
2	No	Married	> 80 K	No
3	No	Single	< 80 K	No
4	Yes	Married	> 80 K	No
5	No	Divorced	> 80 K	Yes
6	No	Married	< 80 K	No
7	Yes	Divorced	> 80 K	No
8	No	Single	> 80 K	Yes
9	No	Married	> 80 K	No
10	No	Single	> 80 K	Yes



CLASSIFICATION METHODS

Example: Test Data Set

SL No	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Married	> 80 K	No
2	No	Single	> 80 K	No
3	No	Single	< 80 K	No
4	No	Married	> 80 K	No
5	No	Divorced	> 80 K	Yes



SL No	Refund	Marital Status	Taxable Income	Cheat	Predicted Cheat
1	Yes	Married	> 80K	No	No
2	No	Single	> 80 K	No	Yes
3	No	Single	< 80K	No	No
4	No	Married	> 80 K	No	No
5	No	Divorced	> 80 K	Yes	Yes

CLASSIFICATION METHODS

Performance Evaluation Measures

1. Confusion Matrix

	Predicted Class		
Actual Class		Class = Yes	Class = No
	Class = Yes	a	b
	Class = No	c	d

2. Accuracy

$$(a+d) \;/\; (a + b + c + d)$$

3. Precision

$$a \;/\; (a + c)$$

Note: Accuracy is a better measure

Example: Performance Evaluation Measures

SL No	Cheat	Predicted Cheat
1	No	No
2	No	Yes
3	No	No
4	No	No
5	Yes	Yes

1. Confusion Matrix

	Predicted Class		
Actual Class		Cheat = No	Cheat = Yes
	Cheat = No	3	1
	Cheat = Yes	0	1

CLASSIFICATION METHODS

Example: Performance Evaluation Measures

1. Confusion Matrix

	Predicted Class		
Actual Class		Cheat = No	Cheat = Yes
	Cheat = No	3	1
	Cheat = Yes	0	1

2. Accuracy

(3+1) / (3 + 1 + 0 + 1) = 4 / 5 = 0.8

3. Precision

3 / (3 + 0) = 3 / 3 = 1.0

CLASSIFICATION METHODS

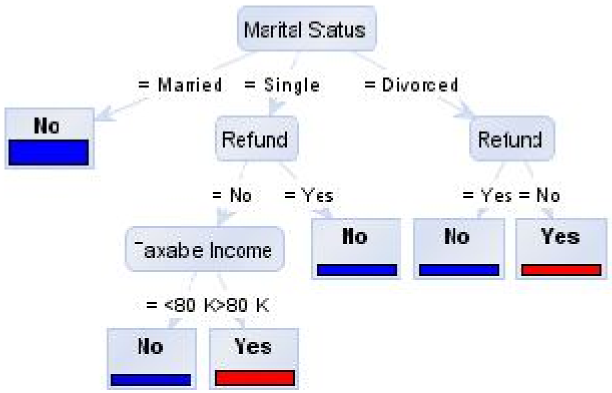
Challenges

How to represent the entire information in the dataset using minimum number of rules?

How to develop the smallest tree?

Solution

Select the attribute with maximum information for first split



Split	Attribute
First	Marital Status
Second	Refund
Third	Taxable Income

	assigned class	money-fx	trade	interest	wheat	corn	grain
true class							
money-fx		95	0	10	0	0	0
trade		1	1	90	0	1	0
interest		13	0	0	0	0	0
wheat		0	0	1	34	3	7
corn		1	0	2	13	26	5
grain		0	0	2	14	5	10

Cost	PREDICTED CLASS	
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	p	q
	Class=No	q	p

$$\begin{aligned}\text{Cost} &= p\,(a + d) + q\,(b + c)\\ &= p\,(a + d) + q\,(N - a - d)\\ &= q\,N - (q - p)(a + d)\\ &= N\,[q - (q\text{-}p) \times \text{Accuracy}]\end{aligned}$$

```
Confusion matrix:                   Cost matrix:
            23  4  0                        -1 5 10
             6 13  3                         5 0 10
             9  2 20                       100 5  0
```

LINEAR REGRESSION

Regression

Correlation helps

To check whether two variables are related

If related

Identify the type & degree of relationship

LINEAR REGRESSION

Regression

Regression helps

- To identify the exact form of the relationship
- To model output in terms of input or process variables

Examples:

Yield = 5 + 3 x Time - 2 x Temperature

$Y = 2 - 5x$

LINEAR REGRESSION

Simple Linear Regression

Output variable is modeled in terms of only one variable

x	y
2	7
1	4
5	16
4	13
3	10
6	19

Regression Model

$$Y = 1 + 3x$$

LINEAR REGRESSION

Simple Linear Regression

General Form:

$$Y = a + bx$$

LINEAR REGRESSION

Simple Linear Regression

Model: $Y = a + bx$

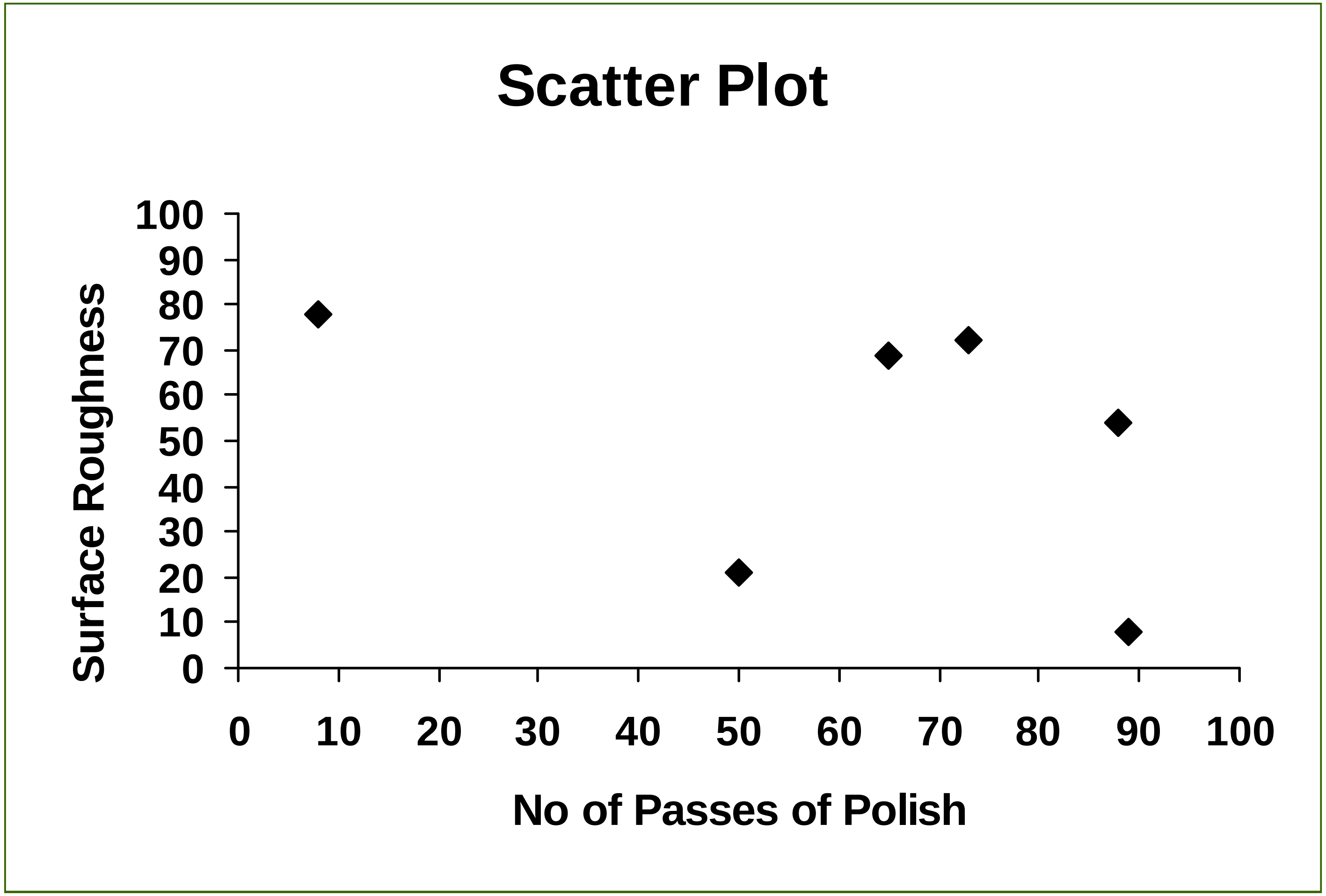
a = Mean y – b.Mean x

$b = Sxy / Sxx$

Regression: Example

x	y
65	69
8	78
89	8
88	21
50	24
73	72

Regression Model $Y = 76.32 - 0.42\,x$



Regression: Issues

For any set of data,

a & b can be calculated

Regression model $Y = a + bx$ can be build

But the model may not be correct

LINEAR REGRESSION

Coefficient of Regression: Measure of degree of Relationship

Symbol : R^2

$$R^2 = SS_R / Syy = b.Sxy / Syy$$

Range of R^2 : 0 to 1

If $R^2 > 0.6$, the Model is reasonably good

LINEAR REGRESSION

Root Mean Square Error:

x	y
65	69
8	78
89	8
88	21
50	24
73	72

Regression Statistics	
Multiple R	0.594159006
R Square	0.353024925
Adjusted R Square	0.191281156
Standard Error	27.80337004
Observations	6

	Coefficients
Intercept	83.00449781
x	-0.605970474

LINEAR REGRESSION

Root Mean Square Error:

x	y	Predicted y	Error	Error Square
65	69	43.62	25.38	644.33
8	78	78.16	-0.16	0.02
89	8	29.07	-21.07	444.08
88	21	29.68	-8.68	75.33
50	24	52.71	-28.71	824.03
73	72	38.77	33.23	1104.32
			Sum	3092.11

Predicted y = 83.0045 – 0.6059 x

Error = y – predicted y

Mean Square Error = 3092.11 / 6 = 515.35

Root Mean Square Error = 22.70

LINEAR REGRESSION

Exercise 1: An IT company wants to develop a model to estimate r the Test Effectiveness in terms of size of the project. The data on size and the corresponding test effectiveness of 14 similar projects is given below. (to facilitate regression analysis, test effectiveness is expressed in square roots)

Can you develop a model for Test Effectiveness in terms of Size?

Size	Test Effectiveness	Size	Test Effectiveness
2.89	0.3464	5.97	0.6300
3.16	0.3606	6.28	0.6403
3.66	0.3560	6.50	0.6481
3.92	0.3400	6.71	0.6700
4.63	0.3900	7.22	0.6800
4.69	0.3845	8.07	0.7400
4.93	0.3500	8.50	0.7700