Machine Learning
W5 Tutorial

COMP30027 | Sandy Luo

Overview

Model Evaluation

Metrics, baselines

Decision Tree

Theory, code

Splitting data

code!

Baseline

Model evaluation:

1. Analyse the good & bad of this model's performance
    - Accuracy, precision, recall...
2. Compare this model's performance with other models' performances
    - **Baseline comparison !**

Baseline

1. **Random**: Guess labels randomly based on class distribution in training
2. **0-R**: Always guess the most common label in the training set
3. **1-R**: Choose a single attribute to represent the entire decision-making process
   a. For each feature, assign the most frequent class to each of its unique values
   b. Select the feature with the lowest classification error as the final rule
4. (some) others:
   a. Regression: Always guess the mean value
   b. Object detection: Always guess the middle of the image

Q1:

Classify the test instances using the method of <u>0-R</u>

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q1:

Classify the test instances

using the method of <u>0-R</u>

0-R: Always guess the most common label in the training set

3 vs 3 → tiebreaker?

Just choose one that is representative, which in this case can be either Y or N

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q2:

Classify the test instances using the method of 1-R

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q2:

1-R:

1. For each feature, assign the most frequent class to each of its unique values
2. Select the feature with the lowest classification error as the final rule

outlook = S:

- **majority class: Play = N**
- **error = 0**

outlook = O:

- **majority class: Play = Y**
- **error = 0**

outlook = R:

- **majority class: Play = Y**
- **error = 1**

→ outlook total error = 1

outlook

↓

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q2:

1-R:

1. For each feature, assign the most frequent class to each of its unique values
2. Select the feature with the lowest classification error as the final rule

temp = H:

- **majority class: Play = N**
- **error = 1**

temp = M:

- **majority class: Play = Y**
- **error = 0**

temp = C:

- **majority class: Play = Y / N**
- **error = 1**

→ temp total error = 2

temp

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q2:

1-R:

1. For each feature, assign the most frequent class to each of its unique values
2. Select the feature with the lowest classification error as the final rule

Humid = N:

- **majority class: Play = N**
- **error = 1**

Humid = H:

- **majority class: Play = Y**
- **error = 1**

→ Humid total error = 2

Humid

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q2:

1-R:

1. For each feature, assign the most frequent class to each of its unique values
2. Select the feature with the lowest classification error as the final rule

Wind = F:

- **majority class: Play = Y**
- **error = 1**

Wind = T:

- **majority class: Play = N**
- **error = 0**

→ Wind total error = 1

Wind

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q2:

1-R:

1. For each feature, assign the most frequent class to each of its unique values
2. Select the feature with the lowest classification error as the final rule

- **Both Outlook and Wind have the lowest classification error**
- **Pick one as final rule**

- e.g. outlook:
  - ID=G: outlook=O, play = Y
  - ID=H: outlook=S, play = N

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q3:

Classify the test instances using the ID3 Decision Tree method:

1. Using ***information gain*** as the splitting criterion
2. Using the ***gain ratio*** as the splitting criterion

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q3: ID3 Decision Tree: *information gain*

1. Measure **initial uncertainty** (entropy)
2. Calculate ***entropy*** for each feature
3. Calculate ***mean information*** (MI)
4. Calculate ***Information Gain***
   - IG(A) = H(R) - MI(O)
5. Select **Best Splitting Feature**:
   - Choose the attribute with the highest Information Gain

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q3: ID3 Decision Tree: *information gain*

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

1. **Measure initial uncertainty** (entropy)

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x)$$

- Initial = no attributes considered
- Class label distribution: 3*N, 3*Y
  - → P(Play=N) = P(Play=Y) = 3/6
- **H(P) = - [(3/6)log(3/6) + (3/6)log(3/6)] = 1**
- Even distribution → high entropy → high uncertainty, want to reduce

Q3: ID3 Decision Tree: *information gain*

outlook

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x)$$

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

2. Calculate entropy for each feature

outlook = S → Play = N

- H(O=s) = - [0*log(0) + (2/2)log(2/2)] = 0

outlook = O → Play = Y

- H(O=O) = - [(1/1)*log(1/1) + 0*log(0)] = 0

outlook = R → 2 * Play=Y, 1 * Play=N

- H(O=R) = - [(2/3)*log(2/3) + (1/3)*log(1/3)]

Q3: ID3 Decision Tree: *information gain*

outlook

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x)$$

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

3. Calculate mean information (MI)

MI = weighted average entropy

H(O=S) = 0, H(O=O) = 0

H(O=R) = - [(2/3)*log(2/3) + (1/3)*log(1/3)]

- MI = (2/6) H(O=s) + (1/6) H(O=O) + (3/6) H(O=R)
- ~= 0.4592

Q3: ID3 Decision Tree: *information gain*

outlook

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x)$$

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

4. Calculate Information Gain

- IG(A) = H(R) - MI(O)
- H(R) = 1
- MI(outlook) = 0.4592
- IG(outlook) = IG(R) - MI(outlook)

  = 1 - 0.4592

  = 0.5408

...then repeat for all attributes

Q3: <u>ID3 Decision Tree: *information gain*</u>

5. Choose the attribute with the highest Information Gain

	R	***Outl***			**Temp**			**H**		***Wind***		**ID**					
		s	**o**	**r**	**h**	**m**	**c**	**h**	**n**	**T**	**F**	**A**	**B**	**C**	**D**	**E**	**F**
Y	3	0	1	2	1	1	1	2	1	0	3	0	0	1	1	1	0
N	3	2	0	1	2	0	1	2	1	2	1	1	1	0	0	0	1
Total	6	2	1	3	3	1	2	4	2	2	4	1	1	1	1	1	1
P(Y)	1/2	0	1	2/3	1/3	1	1/2	1/2	1/2	0	3/4	0	0	1	1	1	0
P(N)	1/2	1	0	1/3	2/3	0	1/2	1/2	1/2	1	1/4	1	1	0	0	0	1
H	1	0	0	0.9183	0.9183	0	1	1	1	0	0.8112	0	0	0	0	0	0
MI		0.4592			0.7924			1		0.5408		0					
IG		0.5408			0.2076			0		0.4592		1					
SI		1.459			1.459			0.9183		0.9183		2.585					
GR		0.3707			0.1423			0		0.5001		0.3868					

why shouldn't we choose ID, where IG = 1?

Q3: <u>ID3 Decision Tree: *information gain*</u>

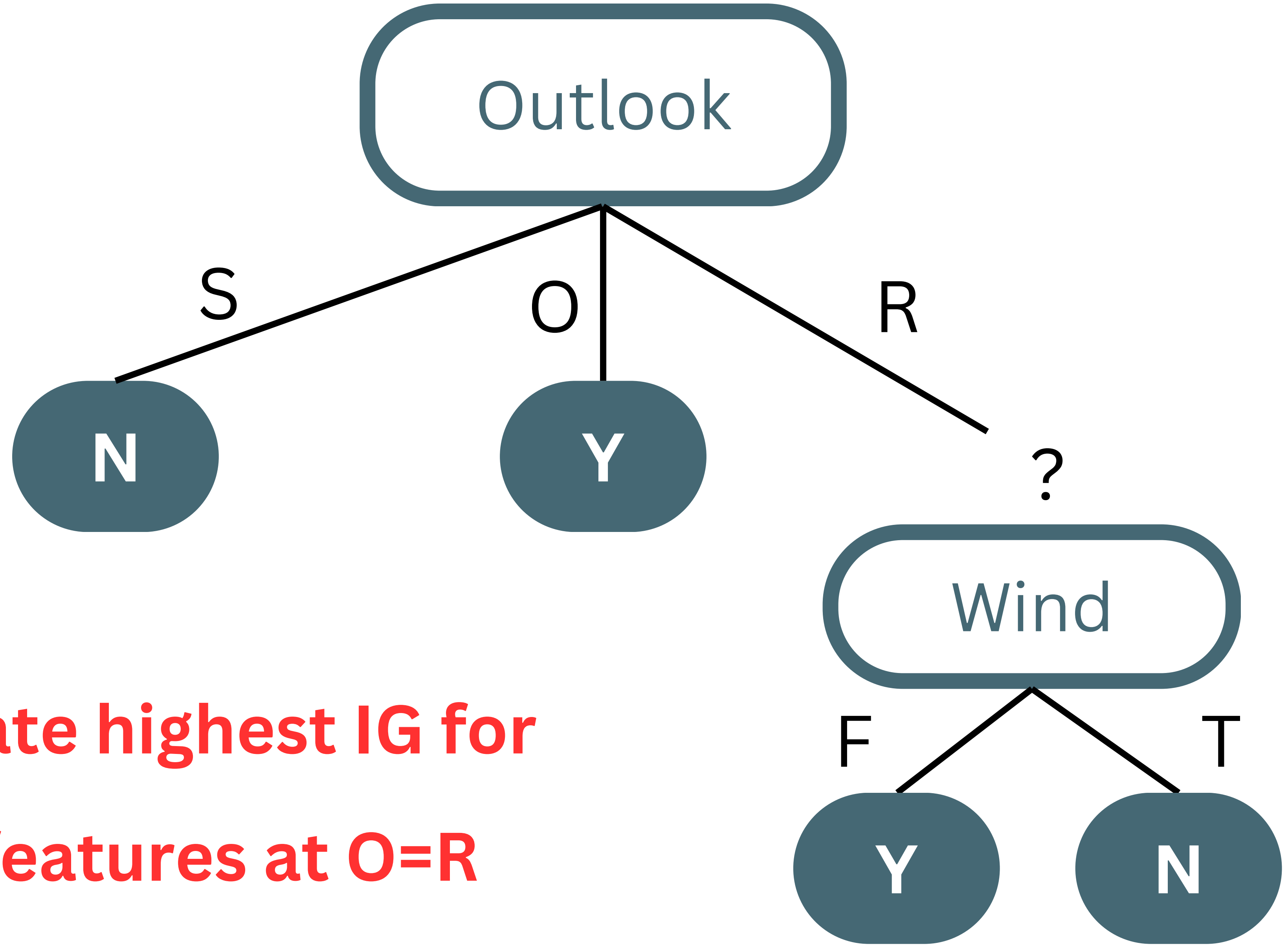
5. Choose the attribute with the highest Information Gain

	R	***Outl***			**Temp**			**H**		***Wind***		**ID**					
		s	**o**	**r**	**h**	**m**	**c**	**h**	**n**	**T**	**F**	**A**	**B**	**C**	**D**	**E**	**F**
Y	3	0	1	2	1	1	1	2	1	0	3	0	0	1	1	1	0
N	3	2	0	1	2	0	1	2	1	2	1	1	1	0	0	0	1
Total	6	2	1	3	3	1	2	4	2	2	4	1	1	1	1	1	1
P(Y)	1/2	0	1	2/3	1/3	1	1/2	1/2	1/2	0	3/4	0	0	1	1	1	0
P(N)	1/2	1	0	1/3	2/3	0	1/2	1/2	1/2	1	1/4	1	1	0	0	0	1
H	1	0	0	0.9183	0.9183	0	1	1	1	0	0.8112	0	0	0	0	0	0
MI		0.4592			0.7924			1		0.5408		0					
IG		0.5408			0.2076			0		0.4592		1					
SI		1.459			1.459			0.9183		0.9183		2.585					
GR		0.3707			0.1423			0		0.5001		0.3868					

why shouldn’t we choose ID, where IG = 1?

Q3: ID3 Decision Tree: *information gain*

5. Choose the attribute with the highest Information Gain



calculate highest IG for

other features at O=R

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q3: ID3 Decision Tree: *Gain Ratio*

Gain ratio:

- GR = IG / **SI**
- Split information (SI):
  - Measure of how evenly the data is split by a feature
  - $$SplitInfo = -\sum \frac{N_i}{N} log_2 \frac{N_i}{N}$$

where Ni is the number of data points containing each value of the variable and N is the total number of data points

equation looks familiar?

entropy → distribution of class labels, SI → distribution of feature values

ID	Outlook	Temp	Humid	Wind	Play
A	S	H	N	F	N
B	S	H	H	T	N
C	O	H	H	F	Y
D	R	M	H	F	Y
E	R	C	N	F	Y
F	R	C	N	T	N

Test set:

ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q3: ID3 Decision Tree: *Gain Ratio*

- IG(Outlook) = 0.5408
- SI(Outlook) =
  - - [(2/6)log(2/6) + (1/6)log(1/6) + (3/6)log(3/6)]
  - ~= 1.459

- GR(Outlook) = IG(Outlook) / SI(Outlook)
  - = 0.5408 / 1.459
  - ~= 0.3707

outlook

$$SplitInfo = -\sum \frac{N_i}{N} log_2 \frac{N_i}{N}$$

ID	Outlook	Temp	Humid	Wind	Play	
A	S	H	N	F	N	2/6
B	S	H	H	T	N	
C	O	H	H	F	Y	1/6
D	R	M	H	F	Y	3/6
E	R	C	N	F	Y	
F	R	C	N	T	N	

Test set:

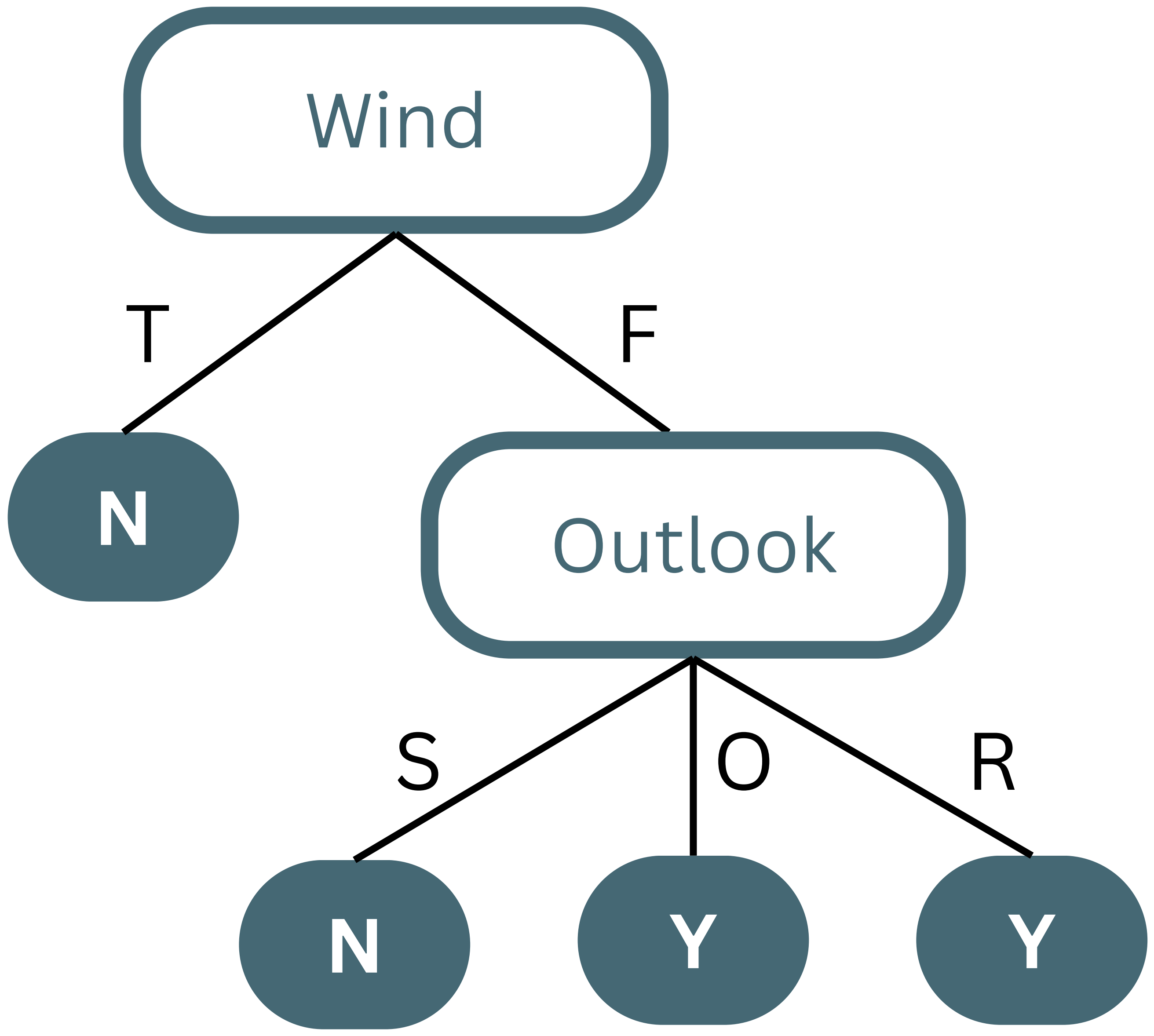
ID	Outlook	Temp	Humid	Wind	Play
G	O	M	N	T	?
H	S	H	H	F	?

Q3: ID3 Decision Tree: *Gain Ratio*

	R	*Outl*			Temp			H		*Wind*		ID					
		s	o	r	h	m	c	h	n	T	F	A	B	C	D	E	F
Y	3	0	1	2	1	1	1	2	1	0	3	0	0	1	1	1	0
N	3	2	0	1	2	0	1	2	1	2	1	1	1	0	0	0	1
Total	6	2	1	3	3	1	2	4	2	2	4	1	1	1	1	1	1
P(Y)	1/2	0	1	2/3	1/3	1	1/2	1/2	1/2	0	3/4	0	0	1	1	1	0
P(N)	1/2	1	0	1/3	2/3	0	1/2	1/2	1/2	1	1/4	1	1	0	0	0	1
H	1	0	0	0.9183	0.9183	0	1	1	1	0	0.8112	0	0	0	0	0	0
MI		0.4592			0.7924			1		0.5408		0					
IG		0.5408			0.2076			0		0.4592		1					
SI		1.459			1.459			0.9183		0.9183		2.585					
GR		0.3707			0.1423			0		0.5001		0.3868					

Repeat for all features → go to next node → repeat until certain class predictions

Q3: ID3 Decision Tree: *Gain Ratio*



Q4:

A confusion matrix is a summary of the performance of a (supervised) classifier over a set of development ("test") data, by counting the various instances:

	Predicted +	Predicted -
Actual +	10	2
Actual -	5	7

Calculate the precision, recall, and F-score (where beta = 1).

Q4:

	Predicted +	Predicted -
Actual +	10	2
Actual -	5	7

- Precision = TP / (TP + FP)

  = 10 / (10 + 5) = 2/3

- Recall = TP / (TP + FN)

  = 10 / (10 + 2) = 5/6

- F1 score = 2 * Precision * Recall / (Precision + Recall)

  = (2 * 2/3 * 5/6) / (2/3 + 5/6) ~= 0.7407

Q5:

How is holdout evaluation different to cross-validation evaluation?

What are some reasons we would prefer one strategy over the other?

1. Hold out:
   - Split data → training / test
   - Only train on training set, evaluate on unseen “held out” test set

Faster! Good for when efficiency important (e.g. large dataset, limited time)

2. Cross-validation:
   - Splits data into k subsets / folds
   - Trains k models, each using k-1 folds for training & 1 fold for testing

More robust! Reduces variance by averaging performance. Preferred if the dataset is smaller / have enough computation