Reminder: Entropy & Information Gain

- Entropy provides a measure of impurity - how uncertain we are about the decision for a given set of examples.

Entropy of a set of examples S with class labels $\{C_1, \ldots, C_n\}$:

$$H(S) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

where p_i is the relative frequency (probability) of class C_i.

- Information Gain measures reduction in entropy when a feature is used to split a set into two or more subsets

IG for feature A that splits a set of examples S into $\{S_1, \ldots, S_m\}$:

$$IG(S, A) = \text{(original entropy) - (entropy after split)}$$

$$IG(S, A) = H(S) - \sum_{i=1}^{m} \frac{|S_i|}{|S|} H(S_i)$$

Each subset is weighted in proportion to its size

Q1(a)

a What is the entropy of this data set with respect to the target class label *Result*?

	Name	Hair	Height	Build	Lotion	Result
1	Sarah	blonde	average	light	no	sunburned
2	Dana	blonde	tall	average	yes	none
3	Alex	brown	short	average	yes	none
4	Annie	blonde	short	average	no	sunburned
5	Emily	red	average	heavy	no	sunburned
6	Pete	brown	tall	heavy	no	none
7	John	brown	average	heavy	no	none
8	Katie	brown	short	light	yes	none

Entropy(Dataset)

= $-(3/8)*\log_2(3/8) -(5/8)*\log_2(5/8)$

= 0.5306 + 0.4238 = 0.9544

$$H(S) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

Q1(b)

b Construct the decision tree that would be built with Information Gain for this data set. Show your work for selection of the root feature in your tree.

	Name	Hair	Height	Build	Lotion	Result
1	Sarah	blonde	average	light	no	sunburned
2	Dana	blonde	tall	average	yes	none
3	Alex	brown	short	average	yes	none
4	Annie	blonde	short	average	no	sunburned
5	Emily	red	average	heavy	no	sunburned
6	Pete	brown	tall	heavy	no	none
7	John	brown	average	heavy	no	none
8	Katie	brown	short	light	yes	none

Steps:

1. Calculate overall dataset entropy.
2. Calculate entropy for each feature.
3. Calculate Information Gain for each feature.

Q1(b)

Calculate entropy values for all features:

Entropy(Hair=blonde) = Entropy(1/3,2/3) = 0.9183
Entropy(Hair=brown) = Entropy(0/4,4/4) = 0
Entropy(Hair=red) = Entropy(1/1,0/1) = 0

Entropy(Height=average) = Entropy(1/3,2/3) = 0.9183
Entropy(Height=tall) = Entropy(2/2,0/2) = 0
Entropy(Height=short) = Entropy(1/3,2/3) = 0.9183

Entropy(Build=light) = Entropy(1/2,1/2) = 1
Entropy(Build=average) = Entropy(1/3,2/3) = 0.9183
Entropy(Build=heavy) = Entropy(1/3,2/3) = 0.9183

Entropy(Lotion=no) = Entropy(2/5,3/5) = 0.9710
Entropy(Lotion=yes) = Entropy(3/3,0/3) = 0

Hair	Result
blonde	sunburned
blonde	none
brown	none
blonde	sunburned
red	sunburned
brown	none
brown	none
brown	none

Height	Result
average	sunburned
tall	none
short	none
short	sunburned
average	sunburned
tall	none
average	none
short	none

Q1(b)

Use Information Gain to choose best feature to split for root node. Try each feature in turn…

IG(Hair) = Entropy(Dataset)
- p(Hair=blonde) * Entropy(Hair=blonde)
- p(Hair=brown) * Entropy(Hair=brown)
- p(Hair=red) * Entropy(Hair=red)

= 0.9544 - (3/8)*0.9183 - (4/8)*0 - (1/8)*0
= 0.610

IG(Height) = 0.9544 - (3/8)*0.9183-(2/8)*0-(3/8)*0.9183 = 0.2657

IG(Build) = 0.9544 - (2/8)*1-(3/8)*0.9183-(3/8)*0.9183 = 0.0157

IG(Lotion) = 0.9544 - (5/8)*0.9710-(3/8)*0 = 0.3475

➞ "Hair" will be selected as the feature with the highest IG value. It perfectly classifies the data for *Hair=brown* & *Hair=red*

Entropy(Hair=blonde) = 0.9183
Entropy(Hair=brown) = 0
Entropy(Hair=red) = 0

Entropy(Height=average) = 0.9183
Entropy(Height=tall) = 0
Entropy(Height=short) = 0.9183

Entropy(Build=light) = 1
Entropy(Build=average) = 0.9183
Entropy(Build=heavy) = 0.9183

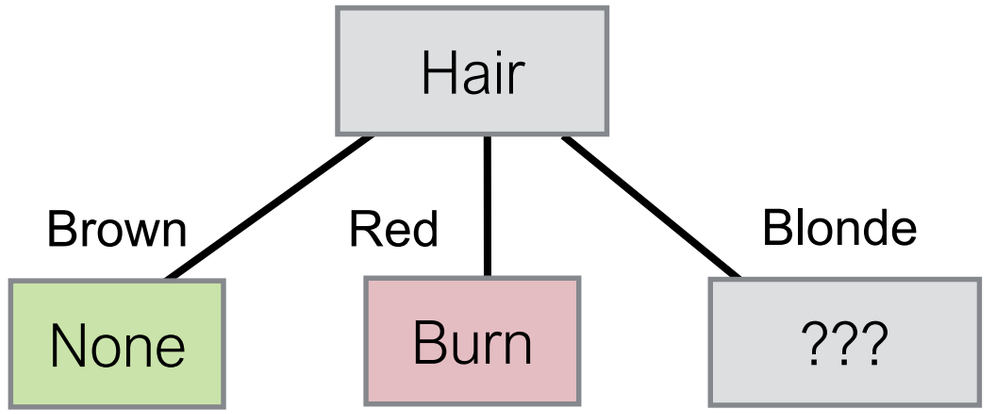
Entropy(Lotion=no) = 0.9710
Entropy(Lotion=yes) = 0

Q1(b)

- “Hair” selected as the feature with the highest IG value ⇒ used to split the root node of the tree.

Child node *Hair=blonde:*

	Name	Hair	Height	Build	Lotion	Result
1	Sarah	blonde	average	light	no	sunburned
2	Dana	blonde	tall	average	yes	none
4	Annie	blonde	short	average	no	sunburned

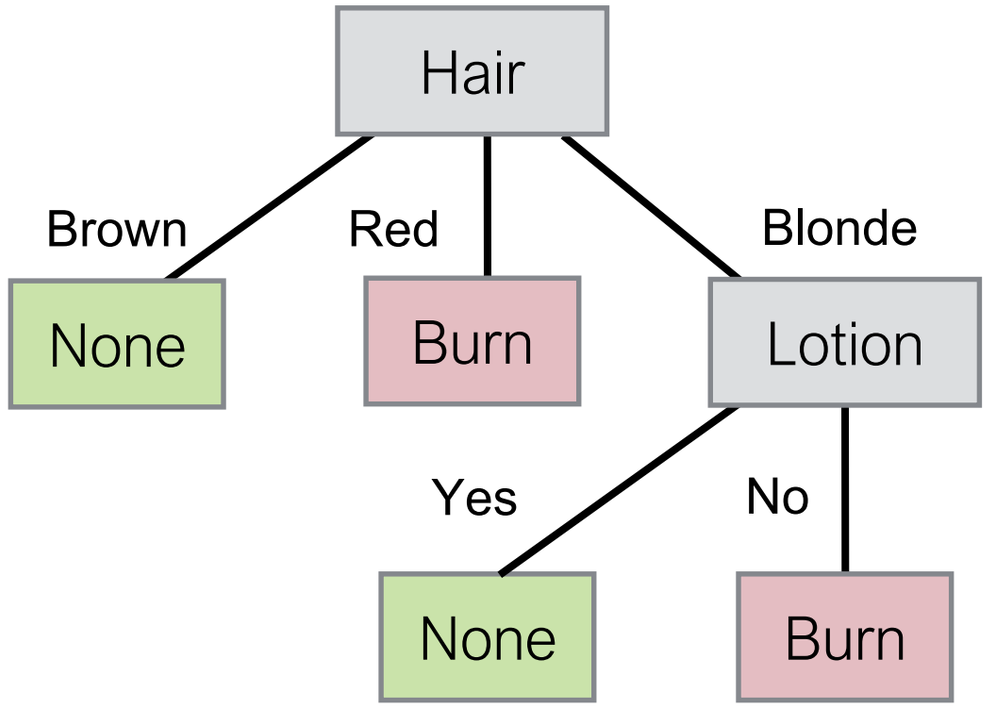


Q1(b)

- "Hair" selected as the feature with the highest IG value ⇒ used to split the root node of the tree.

Child node *Hair=blonde:*

	Name	Hair	Height	Build	Lotion	Result
1	Sarah	blonde	average	light	no	sunburned
2	Dana	blonde	tall	average	yes	none
4	Annie	blonde	short	average	no	sunburned



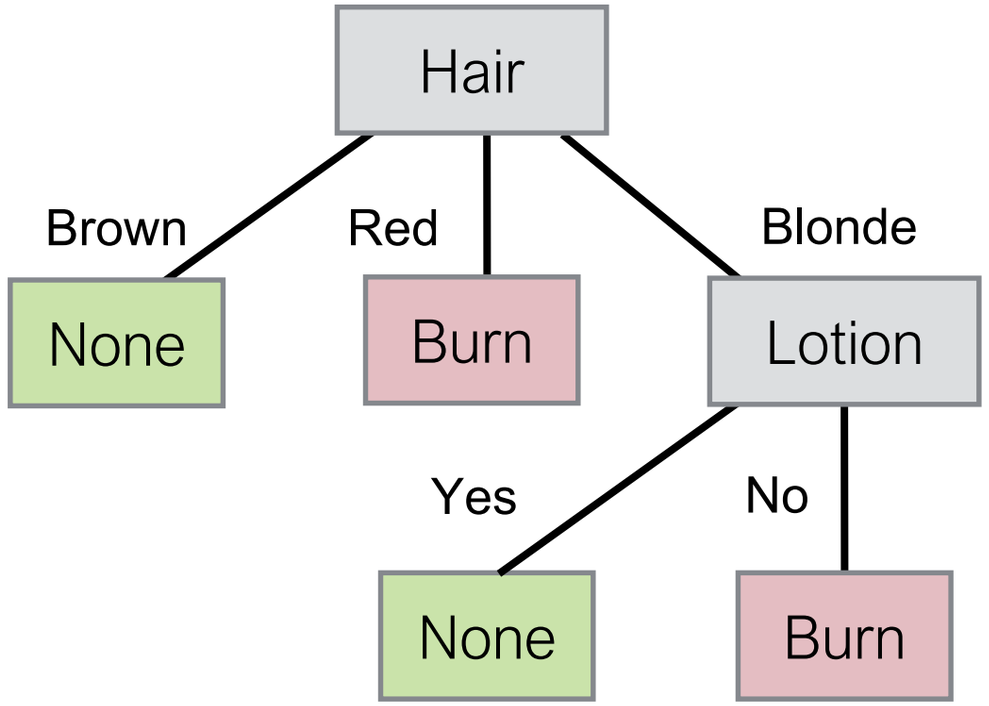
➞ The case for *Hair=blonde* contains (2 sunburned, 1 none). Can split these into pure child nodes using feature "Lotion".

Q1(c)

c Using your decision tree from (b), how would you classify the following example?

	Hair	**Height**	**Build**	**Lotion**	**Result**
X	blonde	average	heavy	no	???

- First, check *Hair=Blonde*
- Next, check *Lotion=No*
- **Output: Sunburned**



Q2(a)

a What is the entropy of this dataset with respect to the target class label Result based on the 14 examples above?

Example	Credit_History	Debt	Income	Risk
1	bad	low	0to30	high
2	bad	high	30to60	high
3	bad	low	0to30	high
4	unknown	high	30to60	high
5	unknown	high	0to30	high
6	good	high	0to30	high
7	bad	low	over60	medium
8	unknown	low	30to60	medium
9	good	high	30to60	medium
10	unknown	low	over60	low
11	unknown	low	over60	low
12	good	low	over60	low
13	good	high	over60	low
14	good	high	over60	low

$-(6/14)*\log_2(6/14) - (3/14)*\log_2(3/14) -(5/14)*\log_2(5/14)$

$= 0.5239 + 0.4762 + 0.5305$

$= 1.5306$

Q2(b)

b Compute the entropy of each of the 3 descriptive features.

	CH	Debt	Income	Risk
1	bad	low	0to30	high
2	bad	high	30to60	high
3	bad	low	0to30	high
4	unknown	high	30to60	high
5	unknown	high	0to30	high
6	good	high	0to30	high
7	bad	low	over60	medium
8	unknown	low	30to60	medium
9	good	high	30to60	medium
10	unknown	low	over60	low
11	unknown	low	over60	low
12	good	low	over60	low
13	good	high	over60	low
14	good	high	over60	low

Entropy(CH=bad) = $-(1/4)*\log_2(1/4)-(3/4)*\log_2 2(3/4) = 0.8113$

Entropy(CH=unknown) = $-(2/5)*\log_2(2/5)-(1/5)*\log_2(1/5)-(2/5)*\log_2(2/5) = 1.5219$

Entropy(CH=good) = $-(1/5)*\log_2(1/5)-(1/5)*\log_2(1/5)-(3/5)*\log_2(3/5) = 1.3710$

Entropy(Debt=low) = $-(2/7)*\log_2(2/7)-(2/7)*\log_2(2/7)-(3/7)*\log_2(3/7) = 1.5567$

Entropy(Debt=high) = $-(4/7)*\log_2(4/7)-(1/7)*\log_2(1/7)-(2/7)*\log_2(2/7) = 1.3788$

Entropy(Income=0to30) = $-(4/4)*\log_2(4/4) = 0$

Entropy(Income=30to60) = $-(2/4)*\log_2(2/4)-(2/4)*\log_2(2/4) = 1$

Entropy(Income=over60) = $-(1/6)*\log_2(1/6)-(5/6)*\log_2(5/6) = 0.65$

Q2(c)

c Which one of the predicting features would be selected by ID3 at the root of a decision tree? Explain your answer.

Use Information Gain to choose best feature to split for root node…

IG(CH) = Entropy(Dataset)
- p(CH=bad) * Entropy(CH=bad)
- p(CH=unknown) * Entropy(CH=unknown)
- p(CH=good) * Entropy(CH=good)

= 1.5306 - (4/14)*0.8113 - (5/14)*1.5219
- (5/14)*1.3710
= 0.2656

Entropy(CH=bad) = 0.8113
Entropy(CH=unknown) = 1.5219
Entropy(CH=good) = 1.3710

Entropy(Debt=low) = 1.5567
Entropy(Debt=high) = 1.3788

Entropy(Income=0to30) = 0
Entropy(Income=30to60) = 1
Entropy(Income=over60) = 0.65

IG(Debt) = 1.5306 - (7/14)*1.5567 - (7/14)*1.3788 = 0.0628

IG(Income) = 1.5306 - (4/14)*0 - (4/14)*1 - (6/14)*0.65 = 0.9663

→ “Income” will be selected as the feature to split as it has the highest IG value.

Q2(d)

d What is the main problem with the Information Gain criterion for attribute selection in decision trees?

"Although information gain is usually a good measure for deciding the relevance of an attribute, it is not perfect. A notable problem occurs when information gain is applied to attributes that can take on a large number of distinct values. For example, suppose that one is building a decision tree for some data describing the customers of a business. Information gain is often used to decide which of the attributes are the most relevant, so they can be tested near the root of the tree. One of the input attributes might be the customer's credit card number. This attribute has a high mutual information, because it uniquely identifies each customer, but we do not want to include it in the decision tree: deciding how to treat a customer based on their credit card number is unlikely to generalise to customers we haven't seen before (overfitting)."

http://en.wikipedia.org/wiki/Information_gain_in_decision_trees