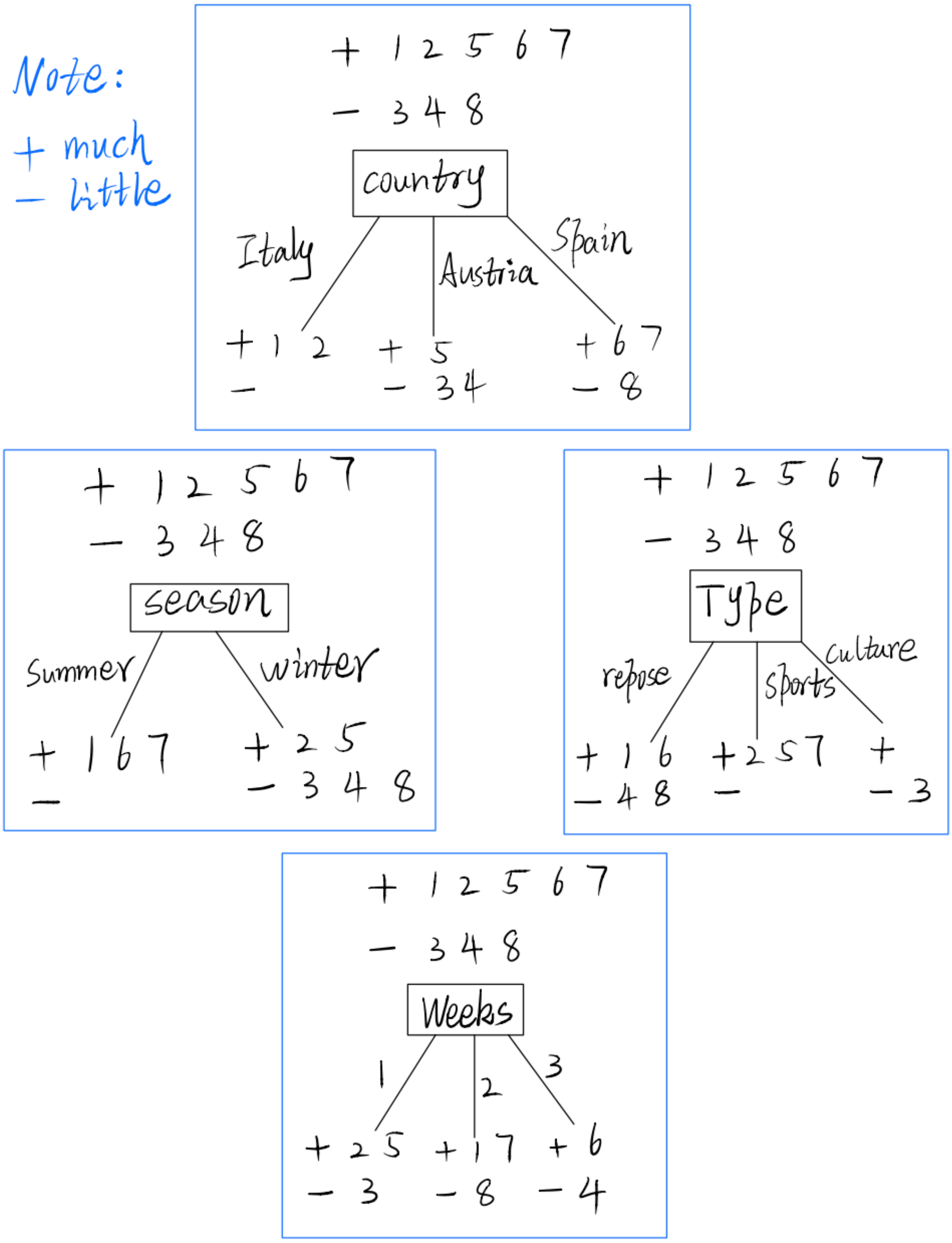
CMPT 310 Assignment 4

Junchen Li
301385486

2021/4/14

Question 1:

The follow picture is graph about an analysis of each of the different attributes “+” sign means “much” and “-” sign means “little”



Entropy(Fun) = $-\mathrm{P(much)} * \log_2(much) - P(little) * \log_2(little)$

+ 1 2 5 6 7 $= -(\frac{5}{8}) * \log_2(\frac{5}{8}) - \frac{3}{8} * \log_2(\frac{3}{8})$

− 3 4 8 ≈ 0.954

Country: Italy, Austria, Spain

Due to the Gain(D,W) = Entropy(D) – $\sum[P\left(\frac{D}{W}\right) * Entropy\left(\frac{D}{W}\right)]$. We need to calculating these three variables respectively.

Entropy (Fun | Country = Italy) = 0 because of there are two value in "much" box nothing in "little" and the answer is 0.

- Entropy (Fun | country = Austria)

$= -\mathrm{P(much)} * \log_2(much) - P(little) * \log_2(little)$

$= -(\frac{1}{3}) * \log_2\left(\frac{1}{3}\right) - \frac{2}{3} * \log_2\left(\frac{2}{3}\right) \approx 0.91829$

- Entropy (Fun | country = Spain)

$= -\mathrm{P(much)} * \log_2(much) - P(little) * \log_2(little)$

$= -(\frac{2}{3}) * \log_2\left(\frac{2}{3}\right) - \frac{1}{3} * \log_2\left(\frac{1}{3}\right) \approx 0.91829$

Gain (Fun, country)

$= Entropy(Fun) - [P(Fun|country = Italy) * Entropy(Fun|country = Italy)] - [P(Fun|country = Austria) * Entropy(Fun|country = Austria)] - [P(Fun|country = Spain) * Entropy(Fun|country = Spain)]$

$= 0.954 - \left(\frac{2}{8}\right) * 0 - \left(\frac{3}{8}\right)(0.91829) - \left(\frac{3}{8}\right)(0.91829) \approx 0.26528$

Seasons: summer, winter

- Entropy (Fun| season = summer) = 0 because there are three variables in "much" box and nothing in the "little" box.
- Entropy (Fun | season = winter)

$= -\mathrm{P(much)} * \log_2(much) - P(little) * \log_2(little)$

$= -(\frac{2}{5}) * \log_2\left(\frac{2}{5}\right) - \frac{3}{5} * \log_2\left(\frac{3}{5}\right) \approx 0.97095$

Gain (Fun, season)

$= Entropy(Fun) - [P(Fun|season = winter) * Entropy(Fun|season = winter)]$

$= 0.954 - \left[\frac{5}{8} * 0.97095\right] - 0 \approx 0.34715$

Type: repose, sports, culture

- Entropy (Fun | type = repose)

$= -\mathrm{P(much)} * \log_2(much) - P(little) * \log_2(little)$

$= -(\frac{2}{4}) * \log_2\left(\frac{2}{4}\right) - \frac{2}{4} * \log_2\left(\frac{2}{4}\right) = 1$

Sports have all in "much" box and nothing in "little" and culture all in "little" box and nothing in "much" box. So the entropy of both of them are 0.

- Entropy (Fun | type = sports) = 0
- Entropy (Fun | type = culture) = 0

Gain (Fun, type) $= 0.954 - (\frac{4}{8})(1) - 0 - 0 \approx 0.4545$

Week have 1,2 and 3 variables.

- Entropy (Fun | Week = 1)

$= -(\frac{2}{3}) * \log_2\left(\frac{2}{3}\right) - \frac{1}{3} * \log_2\left(\frac{1}{3}\right) \approx 0.91829$

- Entropy (Fun | Week = 2) ≈ 0.91829
- Entropy (Fun | Week = 3)

$= -(\frac{1}{2}) * \log_2\left(\frac{1}{2}\right) - \frac{1}{2} * \log_2\left(\frac{1}{2}\right) = 1$

Gain (Fun | week) $= 0.954 - \frac{3}{8}(0.91829) - \left(\frac{3}{8}\right)(0.91829) - \left(\frac{2}{8}\right)(1) \approx 0.01528$

Gain (Fun | country) $= 0.26528$

Gain (Fun | season) $= 0.34715$

Gain (Fun | type) $= 0.454$

Gain (Fun | week) $= 0.01528$

The type factor on decision produces gain the highest score. Then it will be the root rode. Note the positive sign means "much" and negative sign means "little"

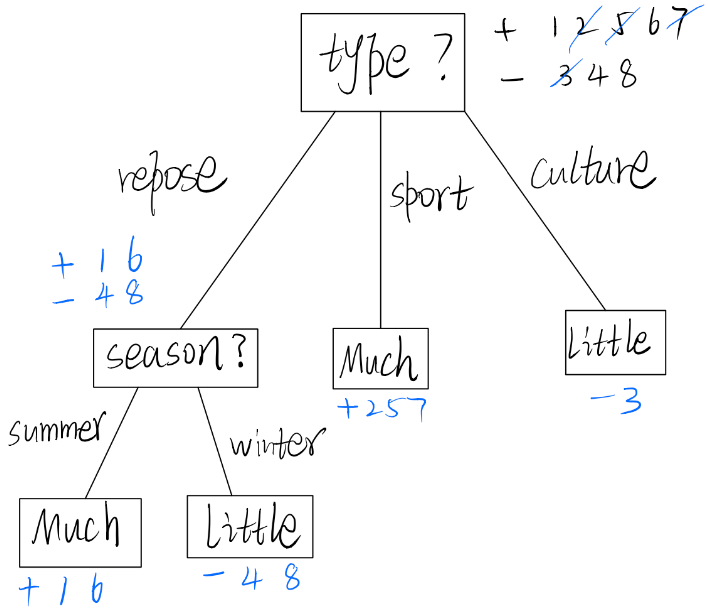
The sport and culture branch will reach the result. Since we can distinguish part of situations, we need to think about the rest of situation (+ 1,6 and – 4,8). We could use the first analysis graph. We need to consider the less branch in the attributes will be the better choice. Also in season part, the 1, 6 and 4, 8 perfectly separated into different choices and different outcomes. Also, we can use some calculations to prove that.

Gain (Type = repose | country) = 0.5

Gain (Type = repose | season) = 1

Gain (Type = repose | week) = 0

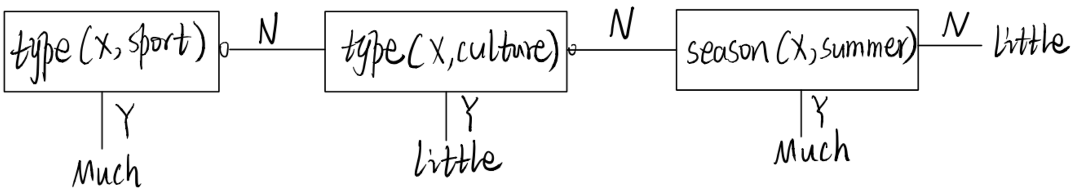
The attribute season have the highest score. So we could choose season as our second attribute. The follow plot is the final result of decision tree.



b)

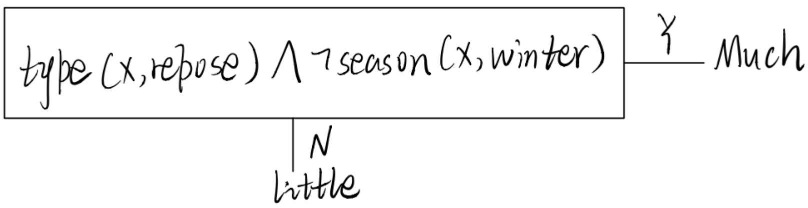
i)

The tests of the DL contain as few literals as possible (e.g. only one, if possible).



ii)

The DL consists of as few tests as possible (e.g. only one, if possible).



Question 2

a) $S(x) = \frac{1}{1+e^{-x}}$

Example	I_D	I_C	I_S	$I_\#$	I_M	I_h	I_m	I_1	T
1	0	0	1	2	1	1	0	0	1
2	0	1	0	3	0	0	0	1	0
3	0	1	0	1	0	0	1	0	0
4	1	0	0	2	1	0	0	1	1
5	1	0	0	1	0	1	0	0	1
6	0	0	1	3	1	0	0	1	1
7	0	1	0	1	0	1	0	0	1
8	0	0	1	2	0	0	0	1	0

The attributes "actors" "marketing" and "reception" we use the local encoding for them. And attributes "genre" and "cost" we use the distributed encoding.

b)

	example	O	E	W_D	W_C	W_S	$W_\#$	W_M	W_h	W_m	W_1
Initial.				+1	+1	+1	+1	+1	+1	+1	+1
Epoch 1	1	1	0								
	2	1	-1		-1		-5				-1
	3	0	0								
	4	0	+1	+3			-1	+3			+1
	5	1	0								
	6	1	0								
	7	0	+1		+1		+1		+3		
	8	1	-1			-1	-3				-1
Epoch 2	1	0	+1			+1	+1	+5	+5		
	2	1	-1		-1		-5				-3
	3	0	0								
	4	0	+1	+5			-1	+7			-1
	5	1	0								
	6	1	0								

	7	**1**	**0**								
	8	**0**	**0**								
Epoch 3	1	**1**	**0**								
	2	**0**	**0**								
	3	**0**	**0**								
	4	**1**	**0**								
	5	**1**	**0**								
	6	**1**	**0**								
	7	**1**	**0**								
	8	**0**	**0**								

Question 3

(a) Feed-forward neural network that has at most one hidden layer. We need to find how many hidden units in the one hidden layer. The follow plot shows the process of calculating.

$$f_1(x_1, x_2, x_3) = (x_1 \equiv (x_2 \wedge x_3))$$

$$\because \quad \alpha \equiv \beta = (\alpha \supset \beta) \wedge (\beta \supset \alpha)$$

$$\alpha \supset \beta \Rightarrow (\neg \alpha \vee \beta)$$

$$= [x_1 \supset (x_2 \wedge x_3)] \wedge [(x_2 \wedge x_3) \supset x_1]$$

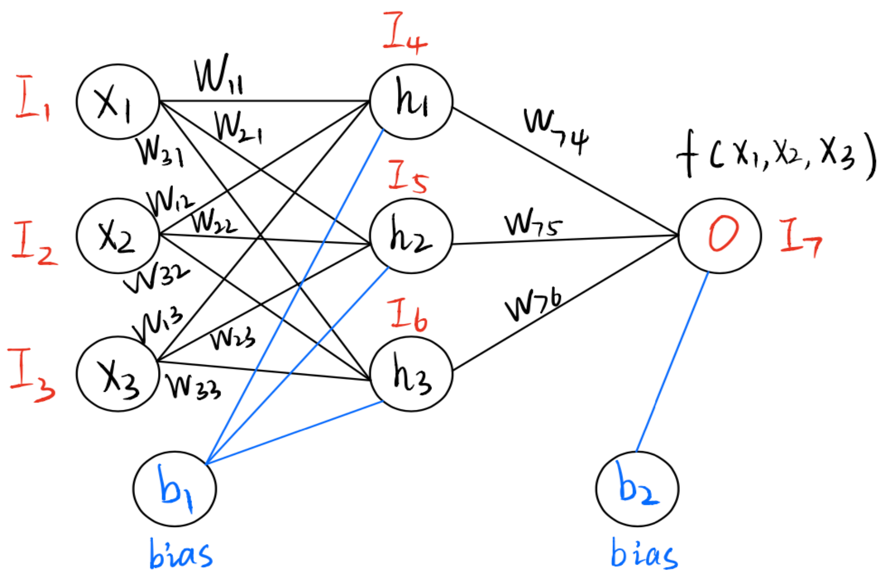
$$= [\neg x_1 \vee (x_2 \wedge x_3)] \wedge [\neg (x_2 \wedge x_3) \vee x_1]$$

$$\because \neg(\alpha \wedge \beta) \Rightarrow (\neg \alpha \vee \neg \beta)$$

$$((\alpha \wedge \beta) \vee \gamma) \Rightarrow ((\alpha \vee \gamma) \wedge (\beta \vee \gamma))$$

$$= (\neg x_1 \vee x_2 \wedge \neg x_1 \vee x_3) \wedge (\neg x_2 \vee \neg x_3 \vee x_1)$$

We could see there are three hidden units in layer and we can build neural network base on that.



Step function

$$\begin{cases} 0 & \text{if } x < t \\ 1 & \text{if } x \geq t \end{cases}$$

$$O_i = Step_o\left(\sum_{j=1}^{6} W_{j,i} * I_j\right)$$

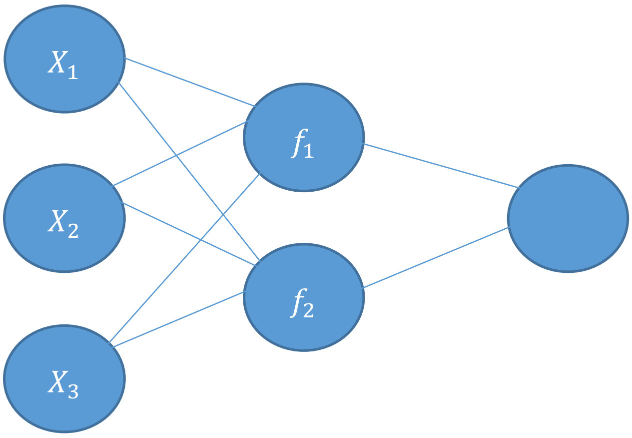
(b)

AND ∧		
0	0	0
0	1	0
1	0	0
1	1	1

OR ∨		
1	1	1
1	0	1
0	1	1
0	0	0

$f_2(x_1,x_2,x_3)=(x_1 \wedge x_2 \wedge x_3) \vee (\neg x_1 \wedge \neg x_2 \wedge \neg x_3))$

X_1	X_2	X_3	$X_1 \wedge X_2 \wedge X_3$	$\neg X_1 \wedge \neg X_2 \wedge \neg X_3$	$F_2(X_1, X_2, X_3)$
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	1	0	1
0	0	0	0	1	1
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0



Make the assumption of otherwise.

$$\begin{cases} 1 & X_1 * W_{11} + X_2 * W_{12} + X_3 * W_{13} \geq t \\ 0 & \text{else} \end{cases}$$

The perceptrons can represent exactly the class of linearly separable function. However, this plow show the function is not a linear separable function. The three input value can combine eight different combinations just shown the above table. However, only two inputs (1,1,1) and (0,0,0) can get the result “1”. I am trying to make a line that can separate this two points from other outcome. I notice that it is impossible that two points on one side and remaining point can stay in other same side. So we can conclude that this Boolean function is **not** linearly separable and we cannot let this Boolean function be represented by a perceptron, using only the step function.