Jordan Shartar
Collaborated with: Graeme Sharpe, Nicholas Loprinzo, Aishwarya Chitnis

Homework 1

1.

Person 1	Yy	yn	Ny
Person 2	yy	yn	Ny
Person 3	yy	yn	ny

Person 4	Yy	yn	Ny
Person 5	yy	yn	Ny
Person 6	yy	yn	ny

P(0) = 1/8
P(1) = 3/8
P(2) = 3/8
P(3) = 1/8

a)

$$P(L) = P(L|0)P(0) + P(L|1)P(1) + P(L|2)P(2) + P(L|3)P(3)$$

= (58/103)(1/8) + (59/103)(3/8) + (60/103)(3/8) + (61/103)(1/8)
= 0.578

b)

$$P(I) = P(I|0)P(0) + P(I|1)P(1) + P(I|2)P(2) + P(I|3)P(3)$$

= (56/103)(1/8) + (57/103)(3/8) + (58/103)(3/8) + (59/103)(1/8)
= 0.558

C)

$$P(S) = P(S|0)P(0) + P(S|1)P(1) + P(S|2)P(2) + P(S|3)P(3)$$

= (58/103)(1/8) + (59/103)(3/8) + (60/103)(3/8) + (61/103)(1/8)
= 0.578

D)

L {58, 59, 60, 61}

W {57, 58, 59, 60}

P(London and Wales) = P(1L, 2W) + P(0L, 1W) + P(2L, 3W)

P(London and Wales) = (59/103)(3/8)(59/103)(3/8) + (58/103)(1/8)(58/103)(3/8) +

(60/103)(3/8) (60/103)(1/8)

P(London and Wales) = 0.7691

Jordan Shartar
Collaborated with: Graeme Sharpe, Nicholas Loprinzo, Aishwarya Chitnis

L {58, 59, 60, 61}

I {56, 57, 58, 59}

P(London and Ireland) = P(0L, 2I) + P(1L, 3I)

P(London and Ireland) = (58/103)(1/8)(58/103)(3/8) + (59/103)(1/8)(59/103)(1/8)

P(London and Ireland) = 0.01999

S {58, 59, 60, 61}

I {56, 57, 58, 59}

P(Scotland and Ireland) = P(0S, 2I) + P(1S, 3I)

P(Scotland and Ireland) = (58/103)(1/8)(58/103)(3/8) + (59/103)(3/8)(59/103)(1/8)

P(Scotland and Ireland) = 0.03024

S {58, 59, 60, 61}

W {57, 58, 59, 60}

P(Scotland and Wales) = P(0S, 1W) + P(1S, 2W) + P(2S, 3W)

P(Scotland and Wales) = (58/103)(1/8)(58/103)(3/8) + (59/103)(3/8)(59/103)(3/8) +
(60/103)(3/8)(60/103)(1/8)

P(Scotland and Wales) = 0.07691

2. P(x=0,y=0) = ¼
   P(x=1,y=0) = 0
   P(x=0,y=1) = ½
   P(x=1,y=1) = ¼

   P(x=0) = ¾
   P(x=1) = ¼

   P(y=0) = ¼
   P(y=1) = 3/4

Jordan Shartar
Collaborated with: Graeme Sharpe, Nicholas Loprinzo, Aishwarya Chitnis

A) $H(x) = -\sum_x P(x) log P(x)$

H(x) = - ((3/4)log(3/4) + (1/4)log(1/4))

H(x) = 0.81128

$H(y) = -\sum_y P(y) log P(y)$

H(y) = - ((1/4)log(1/4) + (3/4)log(3/4))

H(y) = 0.81128

B)

$$H(x|y) = -\sum_x \sum_y P(x,y) log P(x|y)$$

H(x|y) = -((1/4)log(1) + (1/2)log(2/3) + (1/4)log(1/3))

H(x|y) = 0.68872

$$H(y|x) = -\sum_x \sum_y P(x,y) log P(y|x)$$

H(x|y) = -((1/4)log(1/3) + (1/2)log(2/3) + (1/4)log(1))

H(x|y) = 0.68872

C)

I(x;y) = H(y) – H(y|x)

I(x;y) = 0.81128 – 0.68872

I(x;y) = 0.12256

D)

$$H(x,y) = -\sum_x \sum_y P(x,y) log P(x,y)$$

H(x,y) = -((1/4)log(1/4) + (1/2)log(1/2) + (1/4)log(1/4))

H(x,y) = 1.5

E) Show H(y|x) = H(y), x and y independent

$$H(y|x) = -\sum_x \sum_y P(x,y) log P(y|x)$$

$$H(y|x) = -\sum_x \sum_y P(x)P(y) log P(y|x)$$

$$H(y|x) = -\sum_x P(x) \sum_y P(y) log P(y)$$

$$H(y|x) = -\sum_y P(y) log P(y) * 1 = H(y)$$

$$H(y|x) = H(y)$$

Jordan Shartar
Collaborated with: Graeme Sharpe, Nicholas Loprinzo, Aishwarya Chitnis

F) Show H(x,y) = H(x) + H(y), x and y independent

$$H(x|y) = -\sum_x \sum_y P(x,y) log P(x,y)$$

$$H(x|y) = -\sum_x \sum_y P(x,y) log P(y|x) P(x)$$

$$H(x|y) = -\sum_x \sum_y P(x,y) log P(x) - \sum_x \sum_y P(x,y) log P(y|x)$$

$$H(x|y) = -\sum_x \sum_y P(x) log P(x) - \sum_x \sum_y P(x,y) log P(y|x)$$

$$H(x|y) = H(x) + H(y|x) \qquad H(y|x) = H(y)$$

$$H(x|y) = H(x) + H(y)$$

G) Show I(x;x) = H(x)

$$I(x;x) = \sum_x P(x,x) \log\left(\frac{P(x,x)}{P(x)P(x)}\right)$$

$$I(x;x) = \sum_x P(x,x) \log(P(x)) + \sum_x P(x,x) log P(x|x)$$

$$I(x;x) = H(x) - H(x|x) \qquad H(x|x) = 0$$

$$I(x;x) = H(x)$$

H) Show H(x,y) – H(x) = H(y|x)

$$\equiv H(x,y) = H(y|x) + H(x)$$

$$H(x,y) = -\sum_x \sum_y P(x,y) log P(x,y)$$

$$H(x,y) = -\sum_x \sum_y P(x,y) log P(x) P(y|x)$$

$$H(x,y) = -\sum_x \sum_y P(x,y) log P(x) - \sum_x \sum_y P(x,y) log P(y|x)$$

$$H(x,y) = -\sum_x P(x) log P(x) - \sum_x \sum_y P(x,y) log P(y|x)$$

$$H(x,y) = H(x) + H(y|x)$$

$$H(x,y) - H(x) = H(y|x)$$

Jordan Shartar
Collaborated with: Graeme Sharpe, Nicholas Loprinzo, Aishwarya Chitnis

3. C)

   1) False, binary decision trees can represent any linearly seperable equations of Boolean expressions not every continuous function

   2) True, a neural network with 1 hidden layer can model all continuous functions, and one with > 1 hidden layers can model any arbitrary functions (more hidden layers leads to faster better solutions) so this neural network with 4 hidden layers can model any continuous function.