Question 1- Part a, pg1

A1 ; Q1

a) for $E[Z] = \underset{x,y\sim[0,1]}{E}[(X-Y)^2] = E[((X-\frac{1}{2})+\frac{1}{2}-Y)^2]$

$= E[((X-\frac{1}{2}) - (Y-\frac{1}{2}))^2]$

$= E[(X-\frac{1}{2})^2] + E[(Y-\frac{1}{2})^2] + 2E[(X-\frac{1}{2})(Y-\frac{1}{2})]$

Since Y, X are Independent ; $E[(X-\frac{1}{2})(Y-\frac{1}{2})]$
$= E(X-\frac{1}{2})\,E(Y-\frac{1}{2})$

Since X, Y are uniformly distributed on $[0,1]$ $\Rightarrow E(X) = E(Y$

$\Rightarrow E(X-\frac{1}{2}) = E(Y-\frac{1}{2}) = 0$

$\Rightarrow E[Z] = E[(X-\frac{1}{2})^2] + E[(Y-\frac{1}{2})^2]$ $\quad E[(X-\frac{1}{2})^2] = E[(Y-\frac{1}{2}$

$= 2E[(X-\frac{1}{2})^2]$

$= 2\int_0^1 (x-\frac{1}{2})^2\,dx = 2\left[\frac{x^3}{3} - \frac{x^2}{2} + \frac{x}{4}\right]_0^1 = 2\times\frac{1}{12} = \frac{1}{6} = E[Z$

Question 1, part a, pg2.

Q1) a)

for the Variance

$$Var(Z) = Var[(X-Y)^2] = E[((X-Y)^2)^2] - E[(X-Y)^2]^2$$

we compute for $E[(X-Y)^4] = E[X^4 - 4X^3Y + 6X^2Y^2 - 4XY^3 + Y^4]$

$$= E[X^4] - E[4X^3Y] + E[6X^2Y^2] - E[4XY^3] + E[Y^4]$$

$$= \frac{x^5}{5}\Big|_0^1 - 4\frac{x^4}{4}\Big|_0^1 \frac{y^2}{2}\Big|_0^1 + 6\frac{x^3}{3}\Big|_0^1 \frac{y^3}{3}\Big|_0^1 - 4\frac{x^2}{2}\Big|_0^1 \frac{y^4}{4}\Big|_0^1 + \frac{y^5}{5}\Big|$$

$$= \frac{1}{5} - 4\frac{1}{4}\frac{1}{2} + 6\frac{1}{3}\frac{1}{3} - 4\frac{1}{2}\frac{1}{4} + \frac{1}{5} = \frac{2}{5} + \frac{6}{9} - 1 = \frac{48-45}{45} =$$

$$\Rightarrow Var(Z) = \frac{1}{15} - \left[\frac{1}{6}\right]^2 = \frac{7}{180}$$

Question 1, part b

Q 1) b) for expected value

let us expand $E[R] = E\|\| \vec{x} - \vec{y} \|_2^2\|$

$= E\|(X_1 - Y_1)^2 + \cdots + (X_d - Y_d)^2] = E(X_1 - Y_1)^2 + \cdots + E[(X_d - Y_d)^2]$

note that $(X_i - Y_i)^2 = (X_j - Y_j)^2 \quad \forall i, j \in [1, d]$

$\Rightarrow E[R] = d\, E[(X_1 - Y_1)^2]$ we know the eucidian distance squared between two uniformly distributed variables is $\frac{1}{6}$ from part A

$\Rightarrow E[R] = d \times \frac{1}{6} = \frac{d}{6}$

for Variance(R) in the same manner we have

$Var(R) = Var[(X_1 - X_2)^2 + \cdots + (X_d - Y_d)^2]$ since all variables are independent,

$Var(R) = d\, Var[(X_1 - X_2)^2] = \frac{d}{15}$ from part a

Question 1, part c

Q1) c) two points in a d-dim cube are farthest from each other if one is on corner $\langle 0, \ldots, 0 \rangle^d$ and the other $\langle 1, \ldots, 1 \rangle^d$ where the squared euclidean distance will be $\underbrace{1^2 + \ldots + 1^2}_{d} = d$

Now, from part b, we know the expected value and standard deviation of the S.E.D in d-dim is of $O(d)$, since the maximum distance is also of $O(d)$ we can conclude that that ~~distance~~ points in d dimension space become farther, with the same distance as d grows

Question 2, part a and b

Question 2)

a) we have $H(X) = \sum_{x} P(x) \log_2 \left(\frac{1}{P(x)}\right)$ where $P(x)$ is a PDF

it is clear that $0 \leq P(x) \leq 1 \quad \forall x \sim X$

$\Rightarrow \forall x \quad \log_2\left(\frac{1}{P(x)}\right) \geq 0 \quad \left(\text{since } \frac{1}{P(x)} \geq 1\right)$ $P(x) \neq 0$

$\Rightarrow P(x) \log_2 \left(\frac{1}{P(x)}\right) \geq 0 \quad \forall x$

$\Rightarrow \sum_{x} P(x) \log_2 \frac{1}{P(x)} \geq 0 \quad \square$

b) $H(X,Y)$ since X, Y are independent, $P(X,Y) = P(X) \times P$

$\therefore H(X,Y) = \sum_{x}\sum_{y} P(x,y) \log_2 \frac{1}{P(x,y)}$

$= \sum_{x}\sum_{y} P(x) P(y) \log_2 \frac{1}{P(x) P(y)} = \sum_{x,y} P(x) P(y) \log_2 \frac{1}{P(x)} + \sum_{x,y} P(y) P(x) \log_2 \frac{1}{P(y)}$

$= \sum_{x} P(x) \log_2 \frac{1}{P(x)} + \sum_{y} P(y) \log_2 \frac{1}{P(y)} = H(X) + H(Y) \quad \square$

Question 2, part c

Question 2)

Part c) we know that $P(X,Y) = P(X \cap Y) = P(Y \cap X) = P(X) \times P(Y|X)$

we have:

$$H(X,Y) = \sum_{x,y} P(X,Y) \log_2 \frac{1}{P(X,Y)} = \sum_{x,y} P(x) \cdot P(Y|x) \cdot \log_2 \left[\frac{1}{P(x) \cdot P(Y|x)} \right]$$

$$= \sum_{x,y} P(x) \cdot P(y|x) \cdot \log_2 \frac{1}{P(x)} + \sum_{x,y} P(x) \cdot P(y|x) \cdot \log_2 \frac{1}{P(y|x)}$$

$$= \sum_{x} P(x) \cdot \log_2 \frac{1}{P(x)} + \sum_{x,y} P(x,y) \cdot \log_2 \frac{1}{P(y|x)} = H(X) + H(Y|X) \quad \square$$

Question 2, part d

Question 2, Part d)

we know $\log_2(x)$ is concave ; ~~therefore $-\log_2(x)$ is convex~~

$KL(p\|q) = \sum_x p(x) \log_2 \frac{p(x)}{q(x)}$ if we show $-KL(p\|q) \leq 0$

$\Rightarrow KL(p\|q) \geq 0$

$-KL(p\|q) = -\sum_x p(x) \log_2 \frac{p(x)}{q(x)} =$

$\sum_x p(x) \log_2 \frac{q(x)}{p(x)}$

we know log is concave

$\Rightarrow \log[E(X)] \geq E(\log(X))$

$= \left[E \log_2\left(\frac{q(x)}{p(x)}\right)\right]$

$\leq \log_2 E\left[\left(\frac{q(x)}{p(x)}\right)\right]$

$= \log_2 \sum_x p(x) \frac{q(x)}{p(x)}$

$= \log_2 1 = 0 \Rightarrow -KL \leq 0$

$\Rightarrow KL \geq 0$ $\square$

Question 2, part e

Question 2: Part e).

$$I(Y;X) = H(Y) - H(Y|X)$$

$$= -\sum_{y} P(y)\log P(y) + \sum_{x,y} P(x)P(y|x)\log P(y|x) \qquad P(x)\cdot P(y|x) = P(x,y)$$

$$= \sum_{x,y} P(x,y)\log_2 P(y|x) - \sum_{y}\left(\sum_{x} P(x,y)\right)\log_2 P(y) \qquad P(y) = \sum_{x} P(x,y)$$

$$= \sum_{x,y} \overset{P(x,y)}{\cancel{P(x,y)}} \log_2 P(y|x) - \sum_{x,y} P(x,y)\log_2 P(y)$$

$$= \sum_{x,y} P(x,y)\log_2 \frac{P(y|x)}{P(y)}$$

$$= \sum_{x,y} P(x,y)\log_2 \frac{P(y|x)\,P(x)}{P(y)\,P(x)}$$

$$= \sum_{x,y} P(x,y)\log_2 \frac{P(x,y)}{P(y)\,P(x)}$$

$$= KL\left(P(x,y)\,\|\,P(x)\,P(y)\right) \qquad \square$$

Question 3:

Part a)

Please refer to the code and run function `load_data()`

Part b)

Please refer to the code and run function `select_tree_model()`

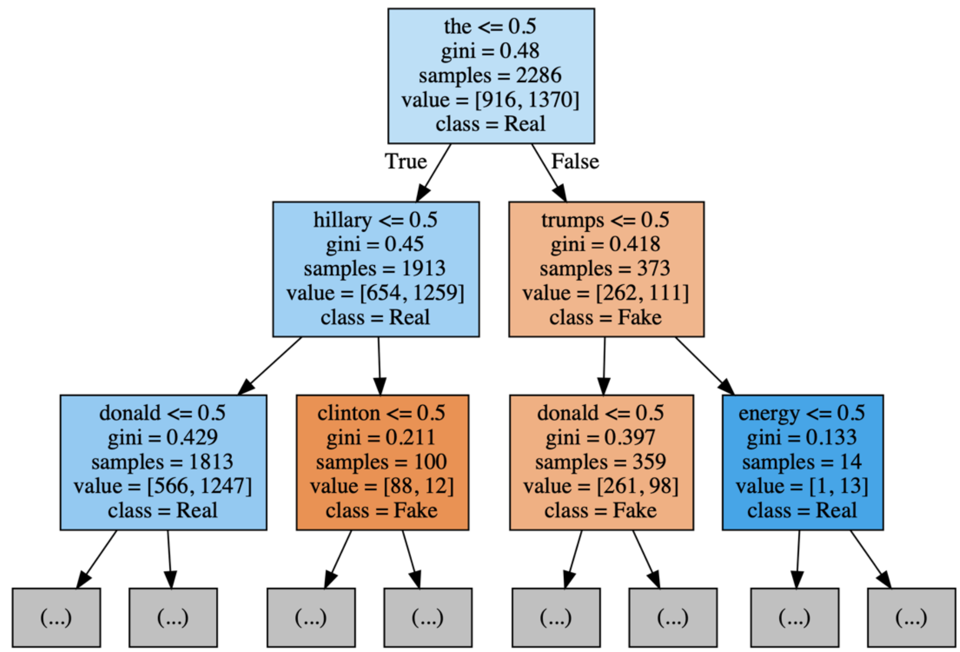
Please also check below for sample reports on the validation sets

Part c)

Reports:

```
(csc311) farazkhoshbakhtian@Farazs-MacBook-Pro 1 % python hw1_code.py
Results for  gini  index, max depth of  16  :
        Accuracy :  74.6938775510204
Results for  gini  index, max depth of  33  :
        Accuracy :  75.71428571428571
Results for  gini  index, max depth of  49  :
        Accuracy :  76.53061224489795
Results for  gini  index, max depth of  66  :
        Accuracy :  75.51020408163265
Results for  entropy  index, max depth of  16  :
        Accuracy :  72.85714285714285
Results for  entropy  index, max depth of  33  :
        Accuracy :  74.08163265306122
Results for  entropy  index, max depth of  49  :
        Accuracy :  74.89795918367346
Results for  entropy  index, max depth of  66  :
        Accuracy :  75.10204081632654
Max accuracy on validation sets: 76.53061224489795
-----
 Best tree is based on: gini and gets this accuracy on the training set:
        Accuracy :  73.67346938775509
```

Graph:



Part d)

Please run compute_information_gain() and check below for the reports:

```
l = ['noneexistenceword', 'the', 'hillary', 'trumps', 'trump', 'donald', 'blasts', 'war']
for word in l:
    ig = compute_information_gain(learning_data['X_train'], learning_data['y_train'], x=word)
    print("Information Gain for <{}> is: {}".format(word, ig))
```

```
(csc311) farazkhoshbakhtian@Farazs-MacBook-Pro 1 % python hw1_code.py
Information Gain for <noneexistenceword> is: 0.0
Information Gain for <the> is: 0.05263747727044343
Information Gain for <hillary> is: 0.0443445873158429
Information Gain for <trumps> is: 0.045006363601046706
Information Gain for <trump> is: 0.034388047535468425
Information Gain for <donald> is: 0.049398847926479306
Information Gain for <blasts> is: 0.0003827842554411376
Information Gain for <war> is: 0.0036013172443748465
(csc311) farazkhoshbakhtian@Farazs-MacBook-Pro 1 %
```

As you can see, “noneexistenceword” gets an information gain of 0 which is correct, and the word “the” gets the highest information gain of all, which justifies it being at the root of the decision tree.

Part e)

For the reports:

```
(csc311) farazkhoshbakhtian@Farazs-MacBook-Pro 1 % python hw1_code.py
Training Accuracy for 1 Nearest Neighbours -->  Accuracy :  100.0
Validation Accuracy for 1 Nearest Neighbours -->        Accuracy :  68.16326530612244
Training Accuracy for 2 Nearest Neighbours -->  Accuracy :  91.86351706036746
Validation Accuracy for 2 Nearest Neighbours -->        Accuracy :  62.44897959183674
Training Accuracy for 3 Nearest Neighbours -->  Accuracy :  85.7830271216098
Validation Accuracy for 3 Nearest Neighbours -->        Accuracy :  69.59183673469389
Training Accuracy for 4 Nearest Neighbours -->  Accuracy :  83.90201224846894
Validation Accuracy for 4 Nearest Neighbours -->        Accuracy :  66.3265306122449
Training Accuracy for 5 Nearest Neighbours -->  Accuracy :  79.39632545931758
Validation Accuracy for 5 Nearest Neighbours -->        Accuracy :  67.75510204081633
Training Accuracy for 6 Nearest Neighbours -->  Accuracy :  80.31496062992126
Validation Accuracy for 6 Nearest Neighbours -->        Accuracy :  65.91836734693878
Training Accuracy for 7 Nearest Neighbours -->  Accuracy :  77.90901137357831
Validation Accuracy for 7 Nearest Neighbours -->        Accuracy :  70.40816326530613
Training Accuracy for 8 Nearest Neighbours -->  Accuracy :  79.04636920384952
Validation Accuracy for 8 Nearest Neighbours -->        Accuracy :  68.36734693877551
Training Accuracy for 9 Nearest Neighbours -->  Accuracy :  75.54680664916886
Validation Accuracy for 9 Nearest Neighbours -->        Accuracy :  67.9591836734694
Training Accuracy for 10 Nearest Neighbours -->         Accuracy :  77.95275590551181
Validation Accuracy for 10 Nearest Neighbours -->       Accuracy :  69.79591836734694
Training Accuracy for 11 Nearest Neighbours -->         Accuracy :  74.93438320209974
Validation Accuracy for 11 Nearest Neighbours -->       Accuracy :  68.77551020408164
Training Accuracy for 12 Nearest Neighbours -->         Accuracy :  77.07786526684166
Validation Accuracy for 12 Nearest Neighbours -->       Accuracy :  70.20408163265306
Training Accuracy for 13 Nearest Neighbours -->         Accuracy :  74.10323709536308
Validation Accuracy for 13 Nearest Neighbours -->       Accuracy :  68.16326530612244
Training Accuracy for 14 Nearest Neighbours -->         Accuracy :  76.42169728783902
Validation Accuracy for 14 Nearest Neighbours -->       Accuracy :  69.79591836734694
Training Accuracy for 15 Nearest Neighbours -->         Accuracy :  72.96587926509186
Validation Accuracy for 15 Nearest Neighbours -->       Accuracy :  67.75510204081633
Training Accuracy for 16 Nearest Neighbours -->         Accuracy :  74.71566054243219
Validation Accuracy for 16 Nearest Neighbours -->       Accuracy :  70.0
Training Accuracy for 17 Nearest Neighbours -->         Accuracy :  72.52843394575677
Validation Accuracy for 17 Nearest Neighbours -->       Accuracy :  67.9591836734694
Training Accuracy for 18 Nearest Neighbours -->         Accuracy :  73.53455818022748
Validation Accuracy for 18 Nearest Neighbours -->       Accuracy :  67.75510204081633
Training Accuracy for 19 Nearest Neighbours -->         Accuracy :  71.21609798775154
Validation Accuracy for 19 Nearest Neighbours -->       Accuracy :  66.73469387755102
Training Accuracy for 20 Nearest Neighbours -->         Accuracy :  72.52843394575677
Validation Accuracy for 20 Nearest Neighbours -->       Accuracy :  68.16326530612244
------
Best model on validation set is with 7 neighbours, with accuracy 70.40816326530613
Test Accuracy for 7 Nearest Neighbours -->      Accuracy :  68.57142857142857
```

For the graph: