Q1 a) $E[z] = E[(x-y)^2] = E[x^2 - 2xy + y^2]$

$= E[x^2] - E[2xy] + E[y^2]$

$= Var[x] + E[x]^2 - E[2xy] + (Var[y] + E[y]^2)$

$= Var[x] + (E[x]^2 - E[2xy] + E[y]^2) + Var[y]$

$= Var[x] + E(x)E(x) - 2E(x)E(y) + Var(y) + E(y)E(y)$

$= \frac{1}{12} + (\frac{1}{2})^2 - 2(\frac{1}{2})(\frac{1}{2}) + \frac{1}{12} + (\frac{1}{2})^2$

$= 1/6$

*$Var[x] = E[x^2] - E[x]^2$

$E[x^2] = Var[x] + E[x]^2$

*Uniform Distribution

$Var[z] = Var[(x-y)^2] = Var[x^2 - 2xy + y^2]$

$= Var(x^2) - Var(2xy) + Var(y^2)$

$= 7/180$

So $E(z) = 1/6$

$Var(z) = 7/180$

b) $E(R) = E(z_1 + z_2 + \cdots + z_d)$

$= E(z_1) + E(z_2) + \cdots E(z_d)$

$= \sum_{i=1}^{d} E(z_i)$

$= d \cdot 1/6 = d/6$

$Var(R) = Var(z_1 + z_2 + \cdots + z_d)$

$= Var(z_1) + Var(z_2) + \cdots Var(z_d)$

$= d \cdot 7/180$

$= 7d/180$

c)

as the results from part b, $E(R_d) = d/6$, $Var(R_d) = 7d/180$.
we can find that expected value and variance is proportional to dimensions.
So as we go into higher dimentions, the mean and variance of euclidean distance in high-dimention increased significantly, losing all the meanings.

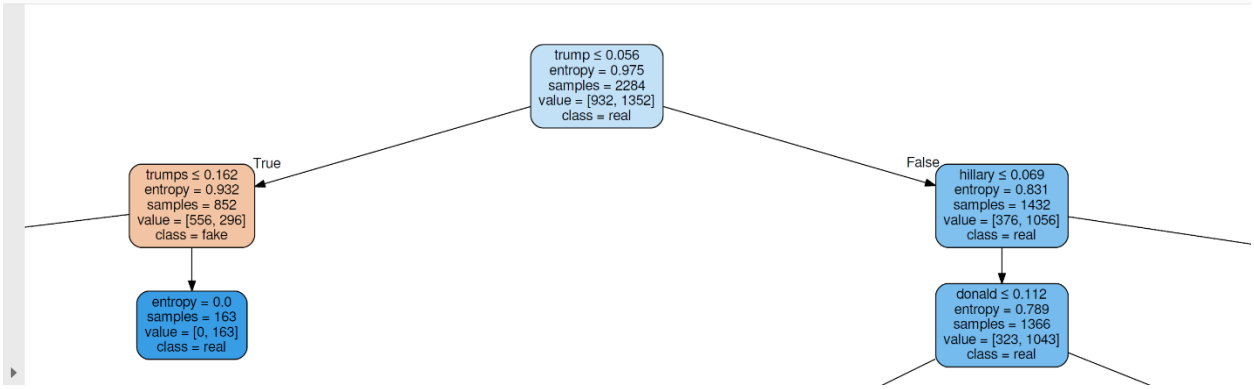
Q2.

a) See in python file

b)

```
root@DESKTOP-O03G45E:/mnt/c/users/lixun/desktop/csc411# python hw1_code.py
criterion=entropy, max depth = 4, score is 0.7934560327198364
criterion=entropy, max depth = 6, score is 0.8139059304703476
criterion=entropy, max depth = 8, score is 0.8118609406952966
criterion=entropy, max depth = 10, score is 0.8098159509202454
criterion=entropy, max depth = 12, score is 0.803680981595092
criterion=entropy, max depth = 14, score is 0.8139059304703476
criterion=gini, max depth = 4, score is 0.7914110429447853
criterion=gini, max depth = 6, score is 0.8077709611451943
criterion=gini, max depth = 8, score is 0.8139059304703476
criterion=gini, max depth = 10, score is 0.8016359918200409
criterion=gini, max depth = 12, score is 0.7995910020449898
criterion=gini, max depth = 14, score is 0.7995910020449898
```

c) As the result shown above, we can see that some hyperparameters reached same validation score. So I will pick criterion= entropy with max depth 14 as example:



d)

```
('keyword: ', 'trumplon', ';information gain: ', nan)
root@DESKTOP-O03G45E:/mnt/c/users/lixun/desktop/csc411# python hw1_code.py
('keyword: ', 'trump', ';information gain: ', 0.0125)
root@DESKTOP-O03G45E:/mnt/c/users/lixun/desktop/csc411# python hw1_code.py
('keyword: ', 'american', ';information gain: ', 0.0035)
root@DESKTOP-O03G45E:/mnt/c/users/lixun/desktop/csc411#
```