CSCE633 HomeWork 2

Tang Yunzhi UIN:629008731

October 2019

1 Question 1

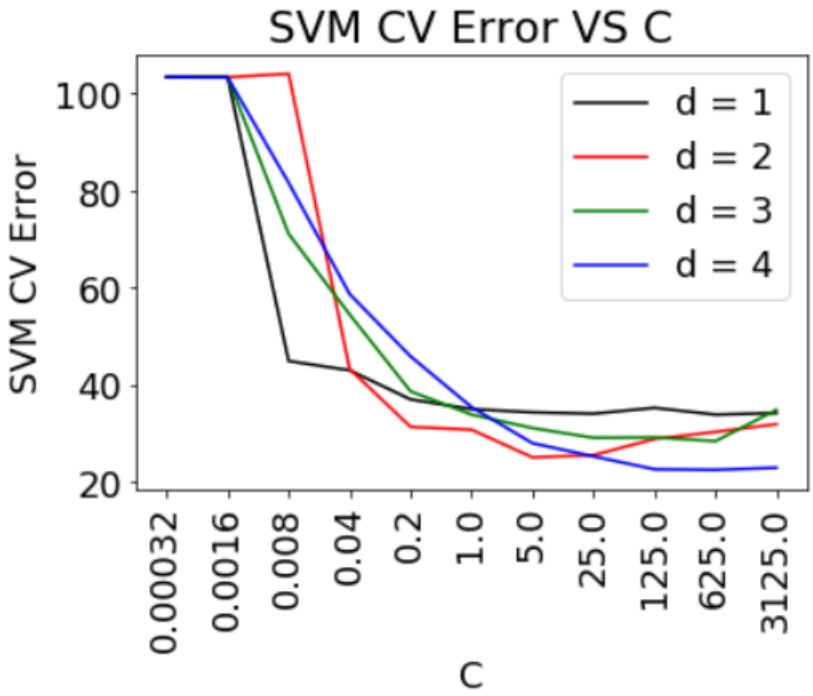
This entire code of this question is in hw2.ipynb
(a)

	Class	feature1	feature2	feature3	feature4	feature5	feature6	feature7	feature8	feature9	...	feature27	feature28	feature29	feature30	feat
0	0.516753	NaN	0.046771	-0.063565	-0.054178	0.046607	0.013642	-0.096574	-0.050629	-0.076969	...	-0.291088	-0.202175	-0.031279	-0.152815	-0.2
1	0.516753	NaN	-0.062320	-0.301661	-0.211203	-0.017909	-0.059086	-0.264995	-0.199066	-0.076969	...	-0.291088	-0.202175	0.030259	-0.083508	-0.2
2	0.516753	0.321252	0.228589	0.150720	0.053260	0.207897	0.150005	0.008689	-0.011566	0.141781	...	0.019256	-0.018842	0.137952	0.045205	-0.0
3	0.516753	NaN	-0.025957	-0.206422	-0.145087	-0.082425	-0.095449	-0.222890	-0.136566	-0.139469	...	-0.038215	-0.093842	-0.031279	-0.083508	-0.1
4	0.516753	NaN	-0.025957	-0.158803	-0.145087	0.046607	-0.059086	-0.180785	-0.167816	-0.014469	...	0.111210	0.006158	0.153336	0.074908	-0.0
5	0.516753	NaN	-0.025957	-0.206422	-0.120294	-0.017909	-0.095449	-0.222890	-0.136566	-0.076969	...	-0.256606	-0.127175	0.030259	-0.083508	-0.1
6	0.516753	NaN	-0.025957	-0.111184	-0.145087	0.046607	-0.022722	-0.054469	-0.136566	-0.076969	...	-0.118675	-0.093842	0.091798	0.035304	-0.1
7	0.516753	NaN	-0.025957	-0.158803	-0.120294	0.111123	0.050005	-0.054469	-0.050629	0.173031	...	-0.118675	-0.093842	0.091798	-0.004300	-0.1
8	0.516753	NaN	-0.025957	-0.158803	-0.145087	-0.017909	-0.059086	-0.180785	-0.136566	-0.076969	...	-0.210629	-0.127175	0.030259	-0.004300	-0.1
9	0.516753	0.352502	0.255862	0.222149	0.078053	0.369187	0.259096	0.198163	0.105621	0.298031	...	-0.084192	-0.010509	0.076413	0.074908	-0.0

10 rows × 37 columns

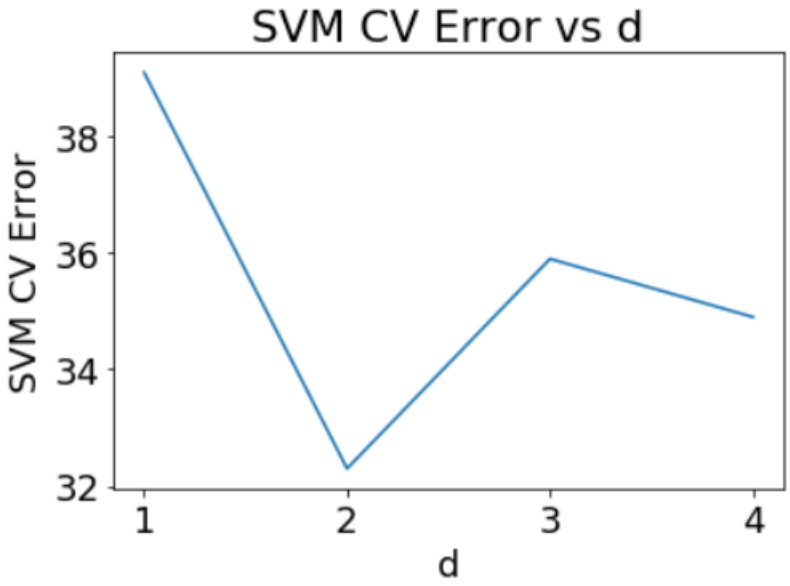
Firstly, I parsed the input data into two pandas library readable file: $parsed_training.csv$
(code in parse.py) and then use $df = (df - df.mean())/(df.max() - df.min())$
to normalize the data

(b)



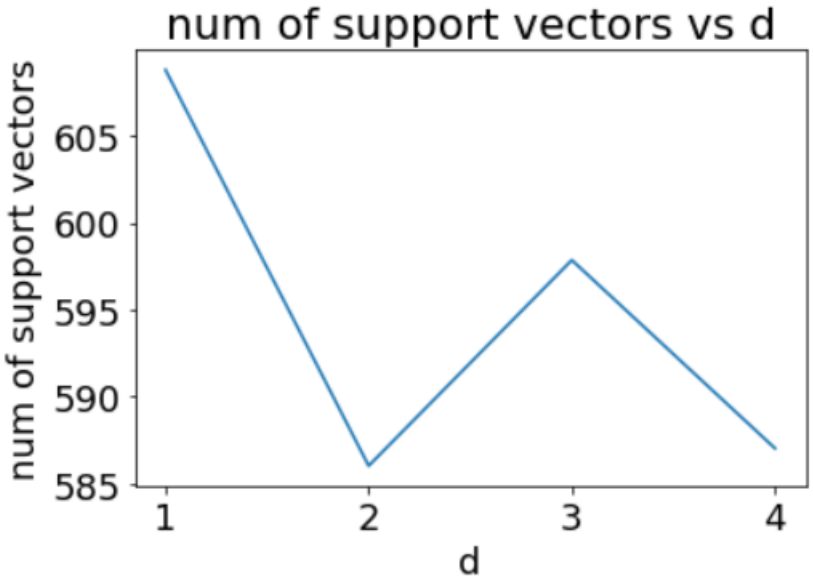
For each value of the polynomial degree, d = 1, 2, 3, 4, The plot shows the average 10-fold cross-validation error vs C, we derive C=5.0, d=2 would be the best value of the trade-off constant C measured on the training internal cross-validation. (C=125 d=4 SVM CV error is the lowest but I am afraid the model is overfitting at this point)

(c)



plot of the test errors for each model, as a function of d is shown above.

(d)



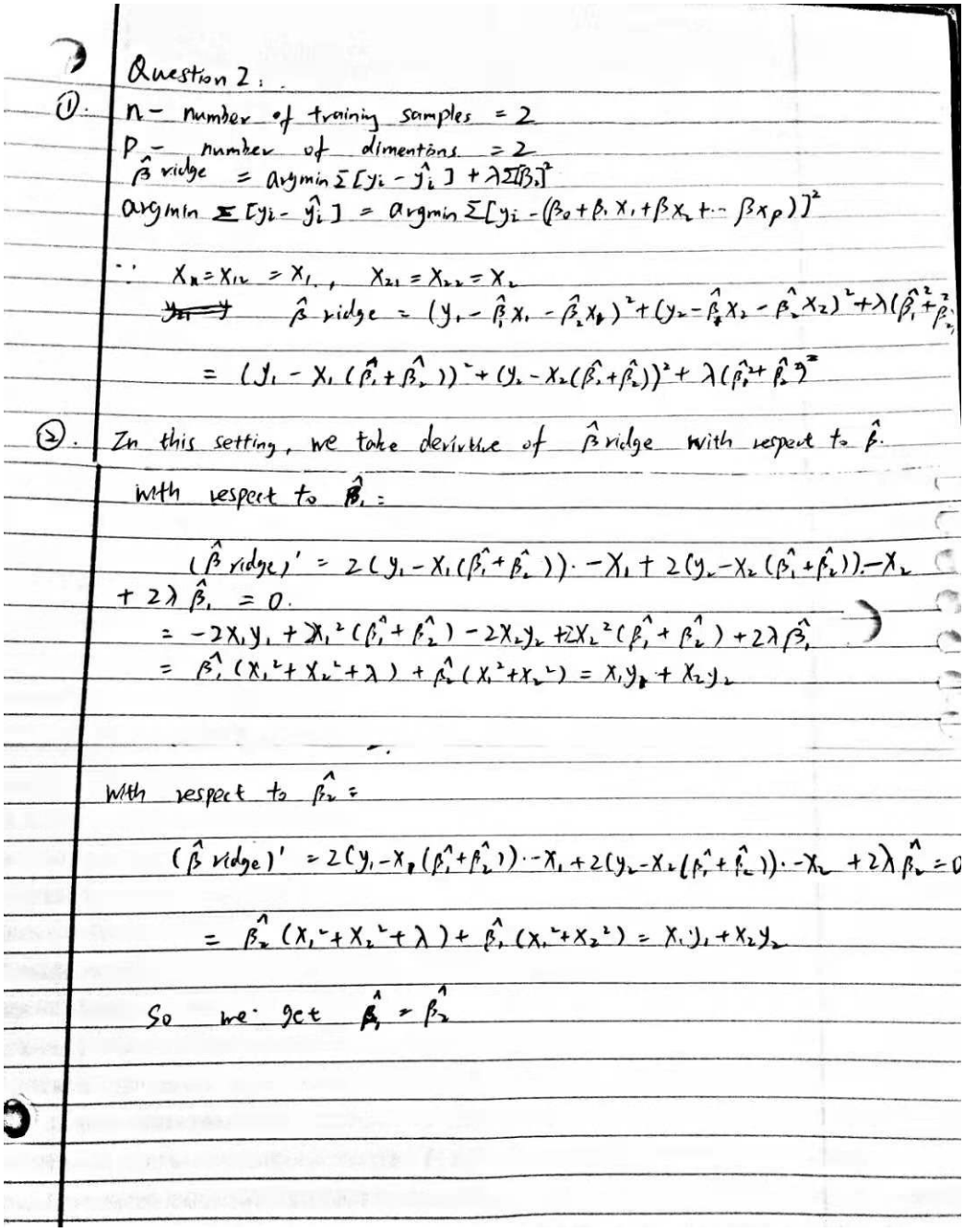
Plot of the average number of support vectors obtained as a function of d is shown above.

(e) There are 74 support vectors lie on the margin hyperplanes.

(f) The degree parameter controls the flexibility of the decision boundary. Higher degree kernels yield a more flexible decision boundary.

(g)$gamma$: γ controls the influence of new features on the decision boundary. The higher the gamma, the more influence of the features will have on the decision boundary, more wiggling the boundary will be.

2 Question 2



Question 2:

① n – number of training samples = 2

p – number of dimentions = 2

$\hat{\beta}^{ridge} = \arg\min \sum [y_i - \hat{y}_i] + \lambda \sum [\beta_j]^2$

$\arg\min \sum [y_i - \hat{y}_i] = \arg\min \sum [y_i - (\beta_0 + \beta_1 X_1 + \beta X_2 + \cdots \beta X_p)]^2$

$\because X_1 = X_{11} = X_1, \quad X_{21} = X_{22} = X_2$

$\hat{\beta}^{ridge} = (y_1 - \hat{\beta}_1 X_1 - \hat{\beta}_2 X_2)^2 + (y_2 - \hat{\beta}_1 X_2 - \hat{\beta}_2 X_2)^2 + \lambda(\hat{\beta}_1^2 + \hat{\beta}_2^2)$

$= (y_1 - X_1(\hat{\beta}_1 + \hat{\beta}_2))^2 + (y_2 - X_2(\hat{\beta}_1 + \hat{\beta}_2))^2 + \lambda(\hat{\beta}_1^2 + \hat{\beta}_2^2)$

② In this setting, we take derivtive of $\hat{\beta}^{ridge}$ with respect to $\hat{\beta}$.

with respect to $\hat{\beta}_1$:

$(\hat{\beta}^{ridge})' = 2(y_1 - X_1(\hat{\beta}_1 + \hat{\beta}_2)) \cdot -X_1 + 2(y_2 - X_2(\hat{\beta}_1 + \hat{\beta}_2)) \cdot -X_2 + 2\lambda\hat{\beta}_1 = 0$

$= -2X_1 y_1 + 2X_1^2(\hat{\beta}_1 + \hat{\beta}_2) - 2X_2 y_2 + 2X_2^2(\hat{\beta}_1 + \hat{\beta}_2) + 2\lambda\hat{\beta}_1$

$= \hat{\beta}_1(X_1^2 + X_2^2 + \lambda) + \hat{\beta}_2(X_1^2 + X_2^2) = X_1 y_1 + X_2 y_2$

With respect to $\hat{\beta}_2$:

$(\hat{\beta}^{ridge})' = 2(y_1 - X_1(\hat{\beta}_1 + \hat{\beta}_2)) \cdot -X_1 + 2(y_2 - X_2(\hat{\beta}_1 + \hat{\beta}_2)) \cdot -X_2 + 2\lambda\hat{\beta}_2 = 0$

$= \hat{\beta}_2(X_1^2 + X_2^2 + \lambda) + \hat{\beta}_1(X_1^2 + X_2^2) = X_1 y_1 + X_2 y_2$

So we get $\hat{\beta}_1 = \hat{\beta}_2$

③. $\hat{\beta}^{Lasso} = \text{Argmin} \Sigma [y_i - \hat{y}_i]^2 + \lambda \Sigma [\beta_j]$

$= \Sigma (y_i - \Sigma x_{ij} \beta_j)^2 + \lambda \Sigma |\beta_j|$

Similar to ridge:

$X_{11} = X_{12} = X_1, \quad X_{21} = X_{22} = X_2.$

To minimize $\hat{\beta}^{Lasso} = (y_1 - \hat{\beta_1} X_1 - \hat{\beta_2} X_1)^2 + (y_2 - \hat{\beta_1} X_2 - \hat{\beta_2} X_2)^2 + \lambda(|\hat{\beta_1}| + |\hat{\beta_2}|)$

$= [y_1 - X_1(\hat{\beta_1} + \hat{\beta_2})]^2 + (y_2 - X_2(\hat{\beta_1} + \hat{\beta_2}))^2 + \lambda(|\hat{\beta_1}| + |\hat{\beta_2}|)$

④. $\min \Sigma (y_i - \beta_0 - \Sigma \beta_i x_i)^2 = \min \Sigma (y_i - \Sigma \beta_i x_i)^2$
is subject to $\Sigma |\beta_i| \le s$ for Lasso.

if $p=2$ Lasso solution falls in the diamond $|\beta_1| + |\beta_2| \le s$

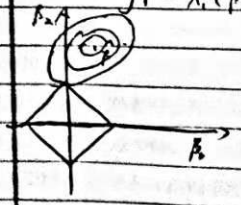
Consider $X_{11} = X_{12} = X_1, \quad X_{21} = X_{22} = X_2. \quad X_1 + X_2 = 0. \quad y_1 + y_2 = 0.$

$(y_1 - X_1(\hat{\beta_1} + \hat{\beta_2}))^2 + (y_2 - X_2(\hat{\beta_1} + \hat{\beta_2}))^2 \ge 0$

$\Rightarrow \quad 2(y_1 - X_1(\hat{\beta_1} + \hat{\beta_2}))^2 \ge 0.$

To Let $2(y_1 - X_1(\hat{\beta_1} + \hat{\beta_2}))^2 = 0. \quad y_1 - X_1(\hat{\beta_1} + \hat{\beta_2}) = 0.$

$y_1 = X_1(\hat{\beta_1} + \hat{\beta_2}) \quad \therefore \hat{\beta_1} + \hat{\beta_2} = \frac{y_1}{X_1}$ is one solution.



the function $(y_1 - X_1(\hat{\beta_1} + \hat{\beta_2}))^2$ ~~intersect~~ intersect with $|\hat{\beta_1}| + |\hat{\beta_2}| = s$. ($\hat{\beta_1} + \hat{\beta_2} = s$ and $\hat{\beta_1} + \hat{\beta_2} = -s$) are solutions to Lasso optimization problem.

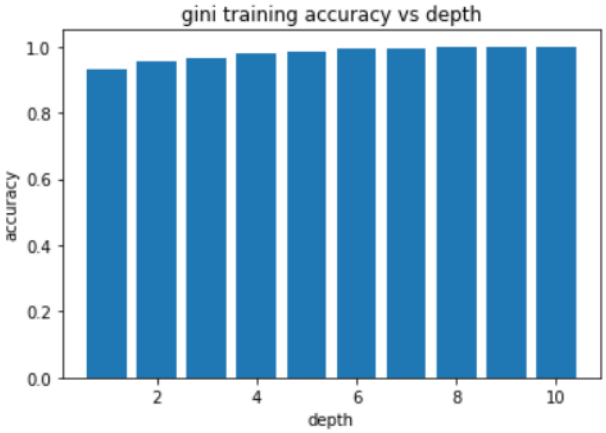
3 Question 3

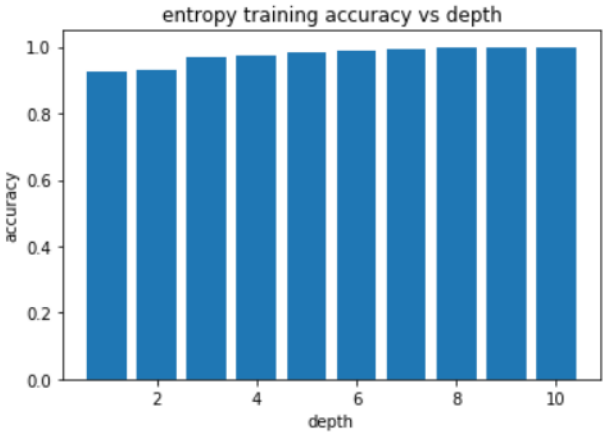
This entire code of this question is in hw2q3.ipynb
(a)Because column 10 is class and 2 for benign and 4 for malignant, By using function $df.feature10.value-counts()$: we get number counts are shown below:

$2(benign)444samples$
$4(malignant)239samples$

I observes that the two classes are not equally represented in the data. The benign samples account for 2/3 of the entire data while malignant samples account for 1/3 of the data.

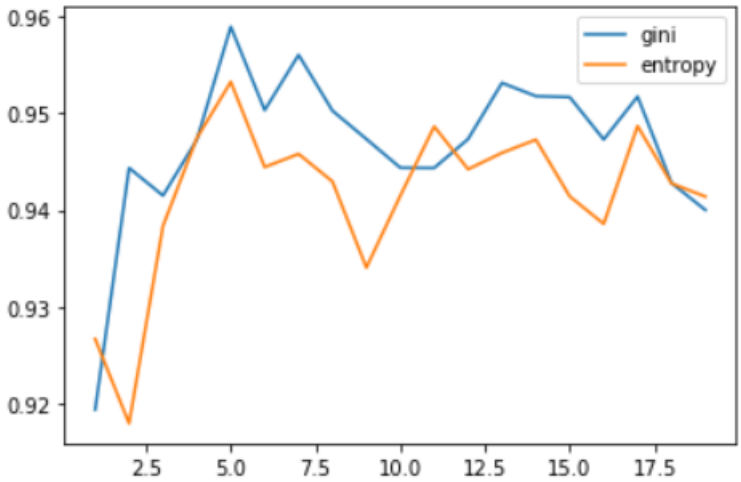
(b) Plot for decision tree model training data accuracy vs depth (depth from 1 to 10)



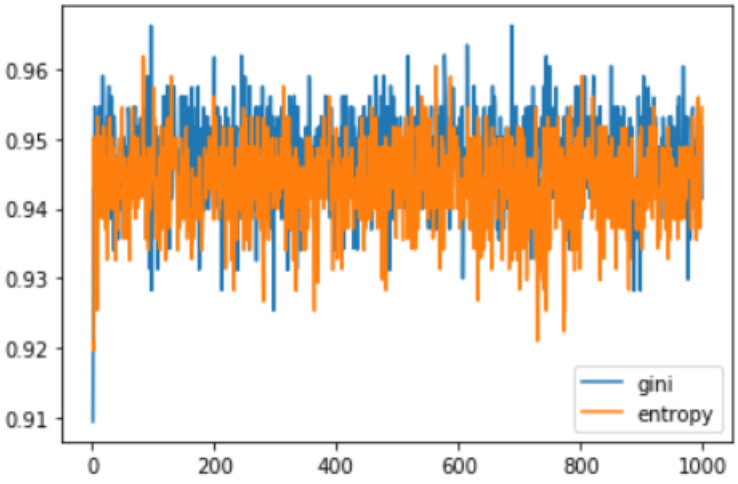


We can observe that gini after depth 5 the accuracy is pretty close to 1(0.99)

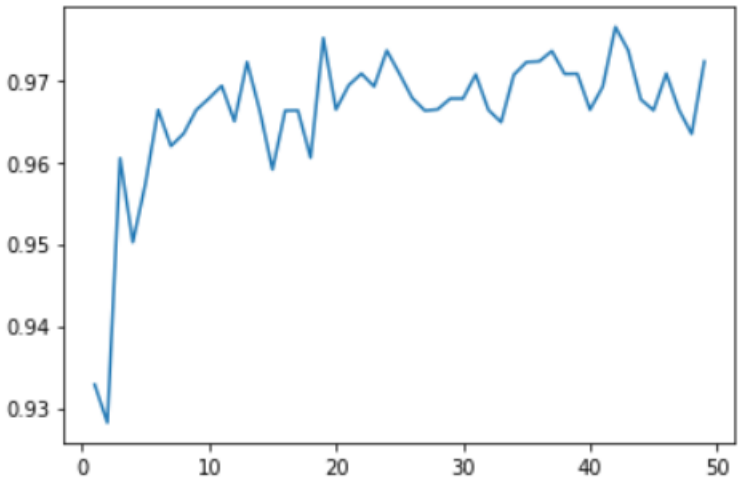
while entropy after depth 6 accuracy is close to 1
Plot for decision tree model test data accuracy vs depth (depth from 1 to 10)



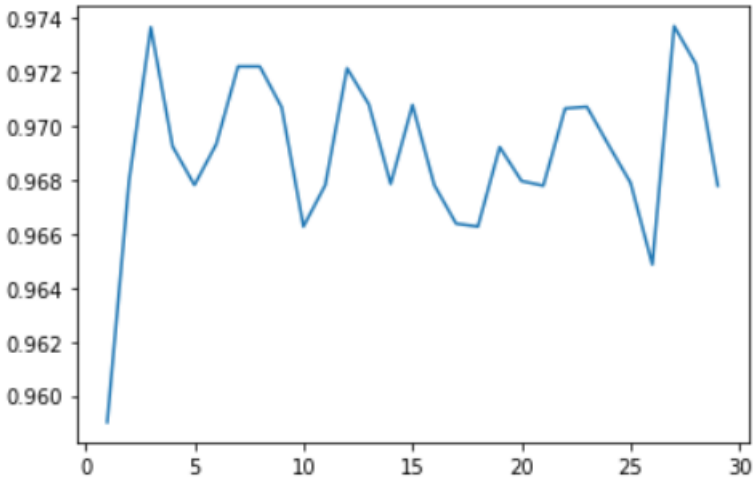
I observe that the test accuracy resides between 0.92-0.96
I also ran a deeper depth test and plot it out. After plotting the 10 fold validation accuracy of the test data while the max depth increases (depth from 1 to 1000), I observe that the general accuracy of gini will be a little higher than entropy and the range of the accuracy resides between [0.93,0.96]and the mean accuracy of the two split criterion is around 0.945.



(c) While doing the secondary 10-fold cross-validation on the training data, I find the optimal number of trees is 42

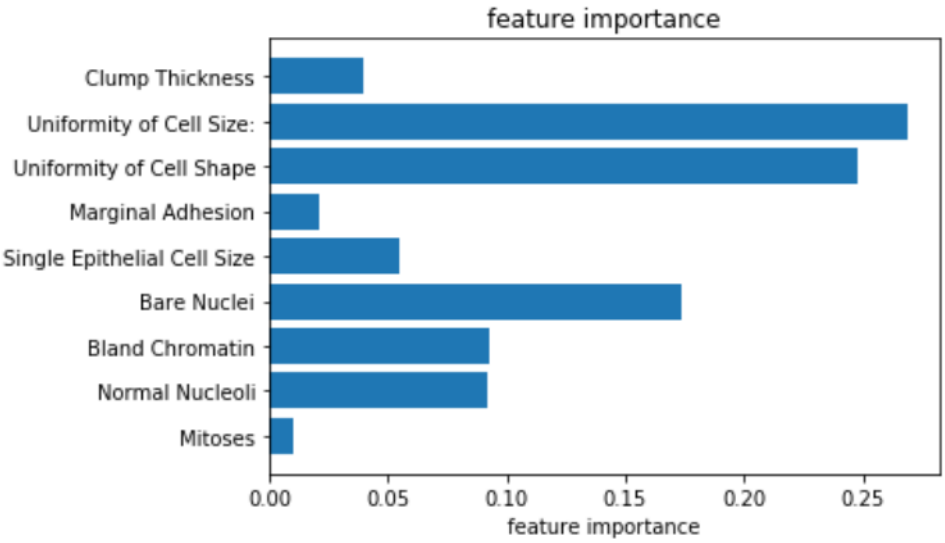


The optimal max depth when n-estimator=42 is 3



So we run 10 cross validation on the model:$clf = RandomForestClassifier(n-estimators = 42, max - depth = 3)$

Then we use the RandomForestClassifier in-built feature importance to derive feature importance of these data. The ranking are shown below:



Ranks:
Uniformity of cell size - 1
Uniformity of cell shape - 2
Bare Nuclei- 3
Bland Chromatin- 4
Normal Nucleoli- 5

The random forest model is more accurate than the decision trees since the mean accuracy of its 10 fold validation is 0.9693295033782494 where decision tree mean accuracy is at most 0.96 while the mean is around 0.945 In scikit-learn's RandomForestClassifier, the feature importance is the mean decrease in impurity (or gini importance) mechanism. The mean decrease in impurity importance of a feature is computed by measuring how effective the feature is at reducing uncertainty (classifiers) or variance (regressors) when creating decision trees within RFs. The problem is that this mechanism, while fast, does not always give an accurate picture of importance. To finalize the feature list, we should perform 10 fold validation and average the feature importance of each fold to get most important features. In this case, I suggest that only Mitoses and Marginal Adhesion should be eliminated from the list beacuse their feature importance is lower than 0.05.