Introduction

The program assignment 2 is a binary classification problem, it asks to correctly classify samples into different letters. To accomplish this goal, 5 different ML models should be applied, their hyperparameter should be tuned using 5-fold cross-validation, and some performance metrics (e.g. accuracy) should be used to help find the best model for this classification problem, then, dimension reduction should be applied, and the performance of different models before dimension reduction and after dimension reduction should be compared. Ultimately, some conclusions can be drawn from this.

I chose A and B for the third problem, before working on this problem, I tend to think M and Y will be the easiest to classify since the number of samples of these 2 letters is the largest among all 3 pairs of the 3 classification problems, which may could bring more information.

Dimension reduction should be regarded as useful for this problem, because before reducing the dimension, this dataset contains a sum of 16 features and is relatively too many, also, some of the features may be similar to each other hence should be eliminated.

A good dimension reduction method should be method that is able to eliminate the expected number of features while avoiding a significant accuracy loss/

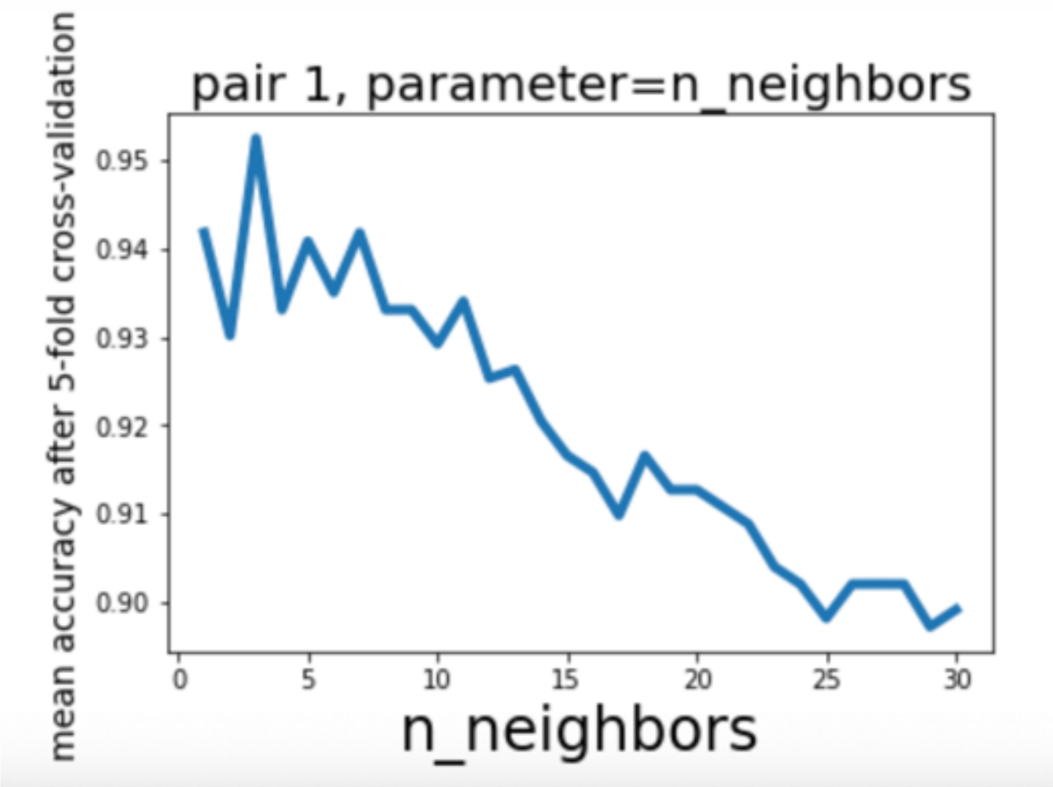
Results

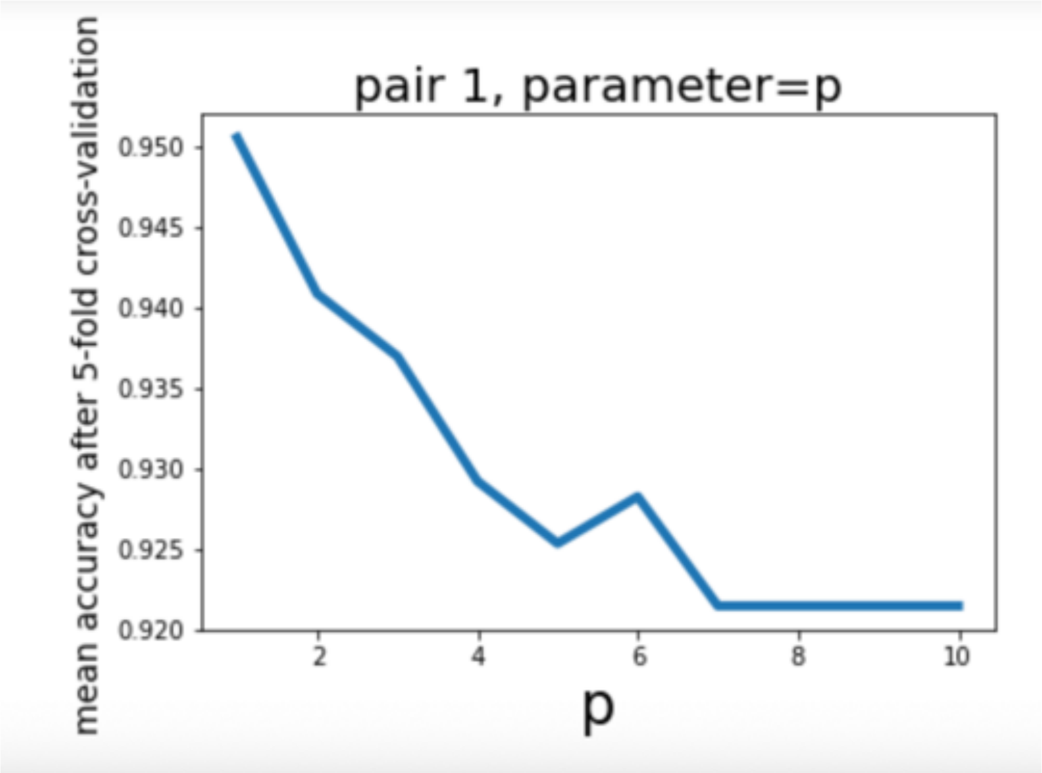
The pros and cons of each classifier are list in the following table.

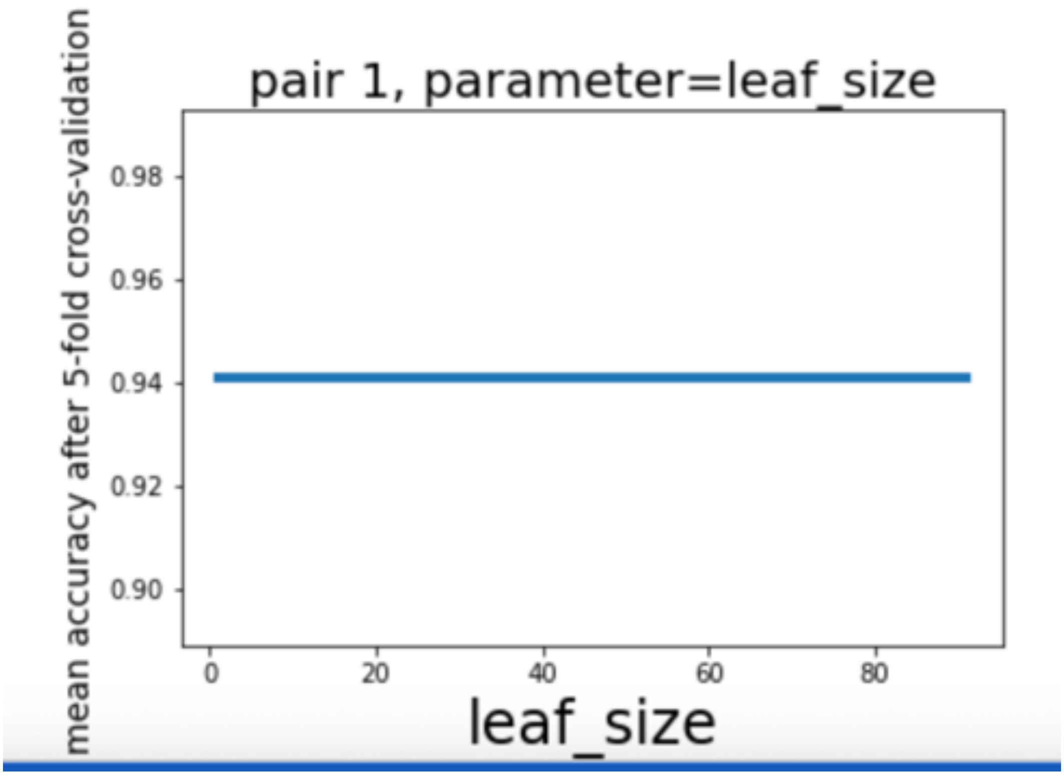
	pros	cons
k-nearest neighbors	Requires no training time	Requires the storage of all the training data, and the testing time of calculating the distance between test samples and every training sample is relatively inefficient
Decision tree	Easy to apply	Easy to overfit
Random forest	Can have high accuracy while avoiding overfitting like a single decision tree	Takes a long-time training
SVM	Good robustness，able to avoid dimensional disasters	Difficult to implement for large training samples, Difficult in solving the problem of multiple classifications
Artificial neural network	Can fit just about any dataset with high accuracy	Difficult to interpret to people with no data science or machine learning background; long training time

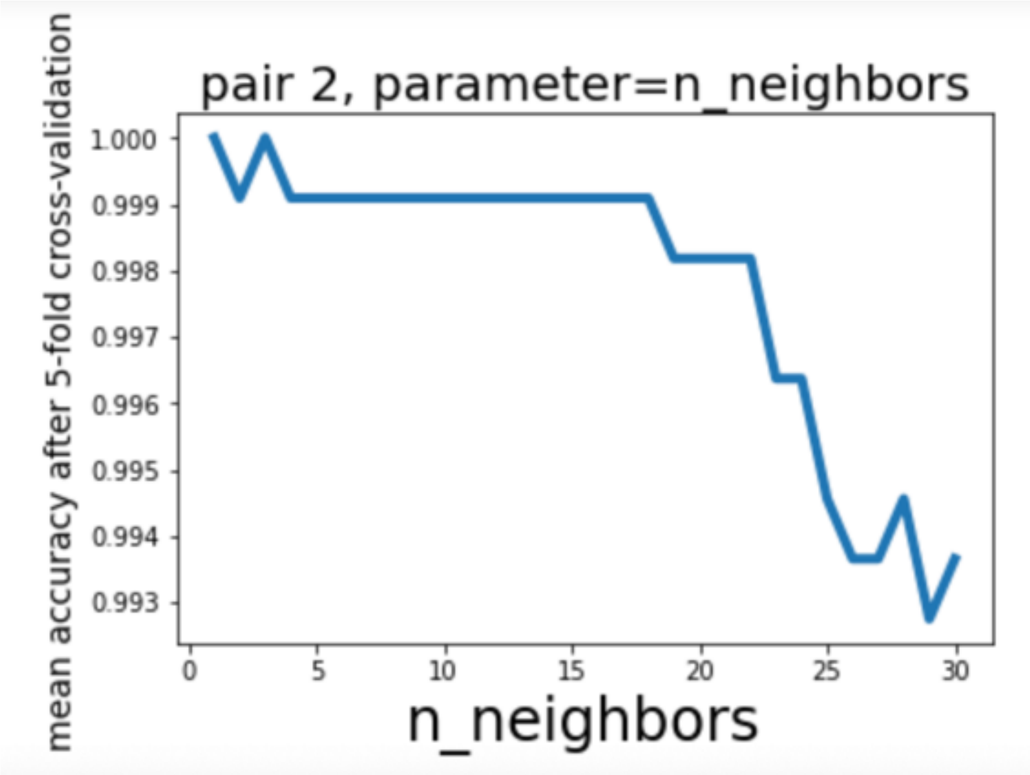
Cross validation results

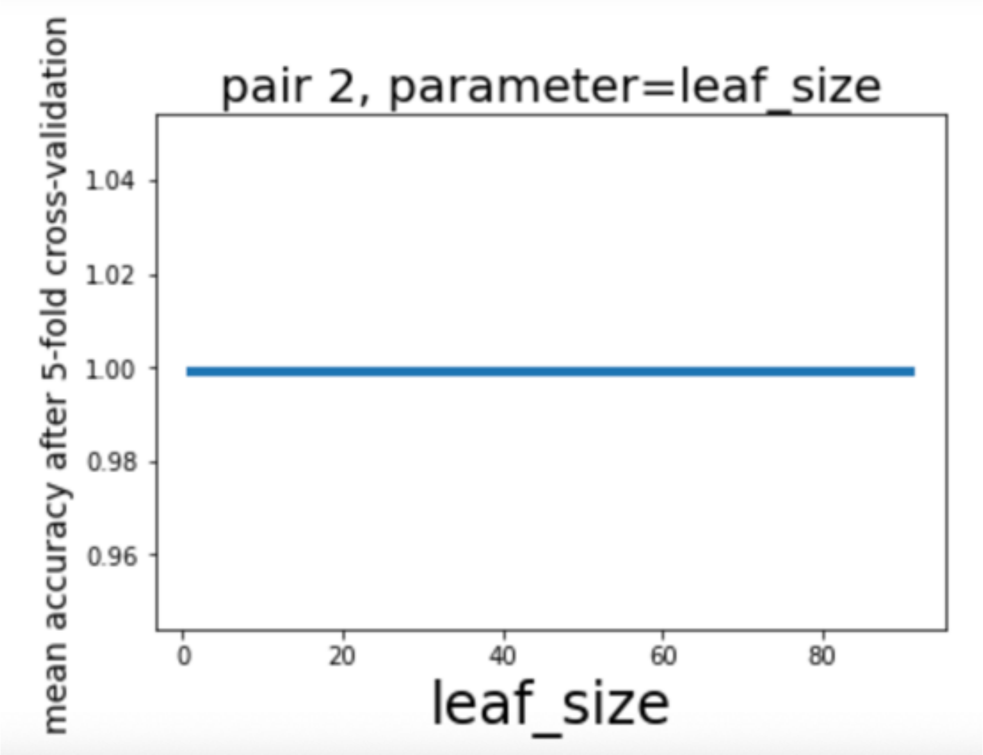
KNN

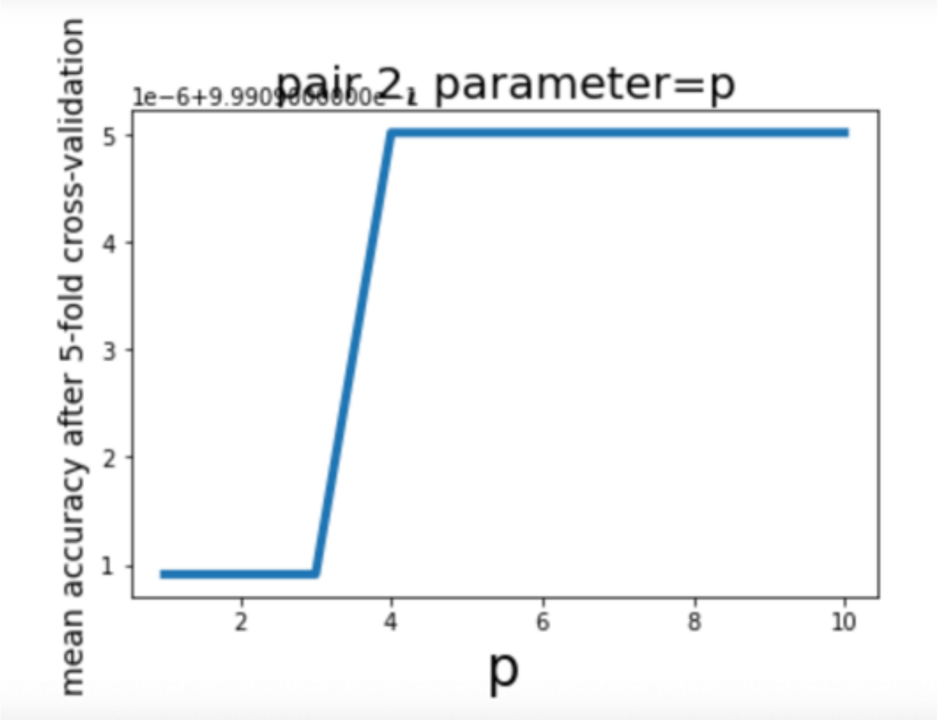


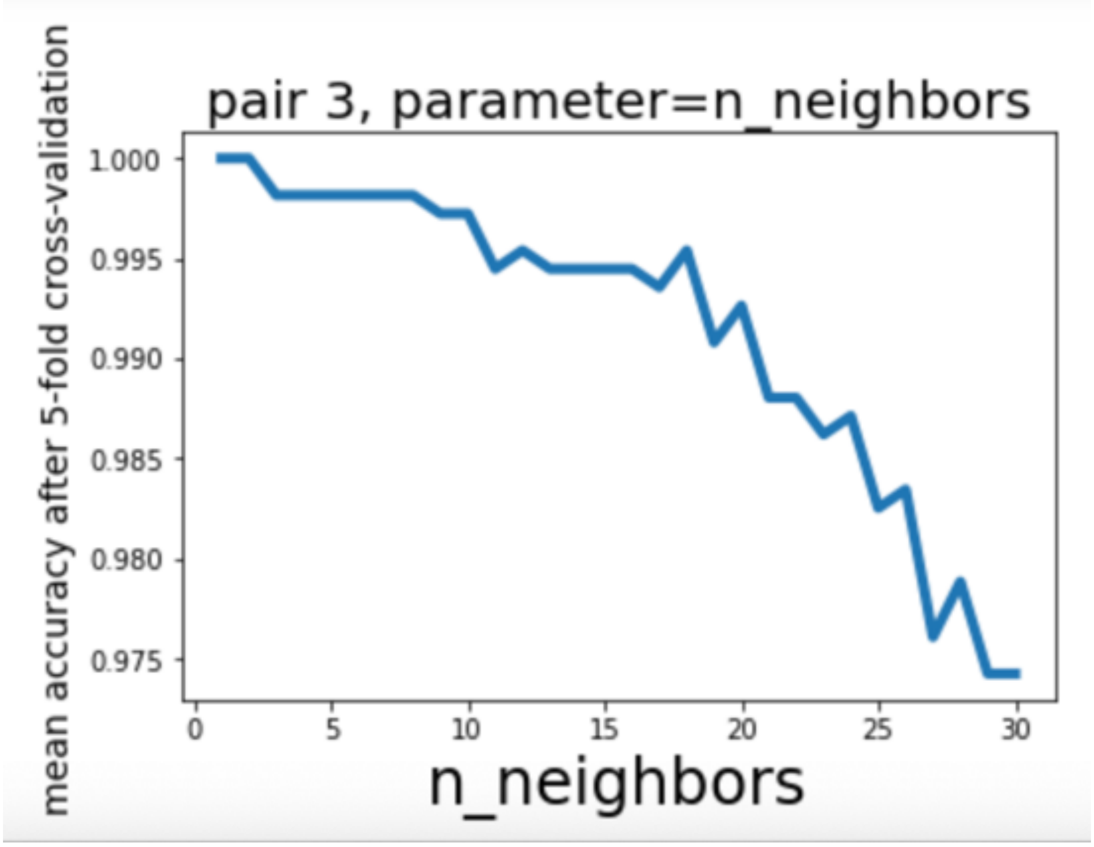


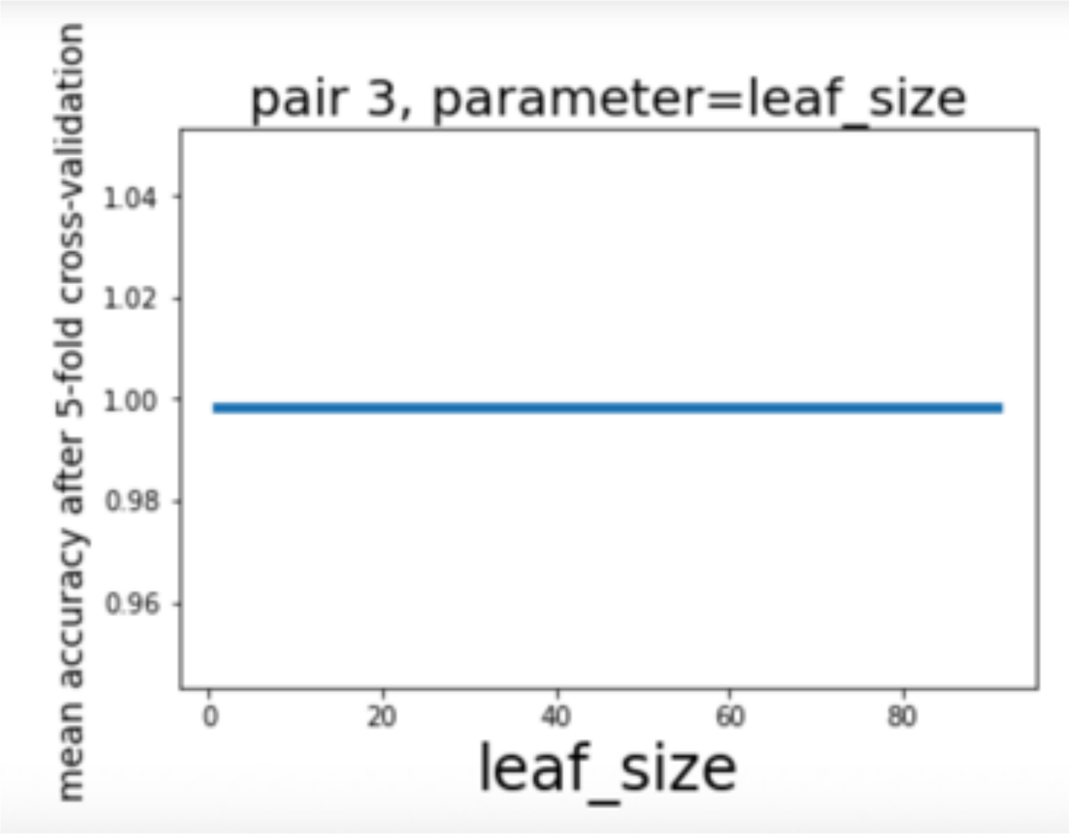

pair 1, parameter=leaf_size
mean accuracy after 5-fold cross-validation
0.98
0.96
0.94
0.92
0.90
0
20
40
60
80
leaf_size

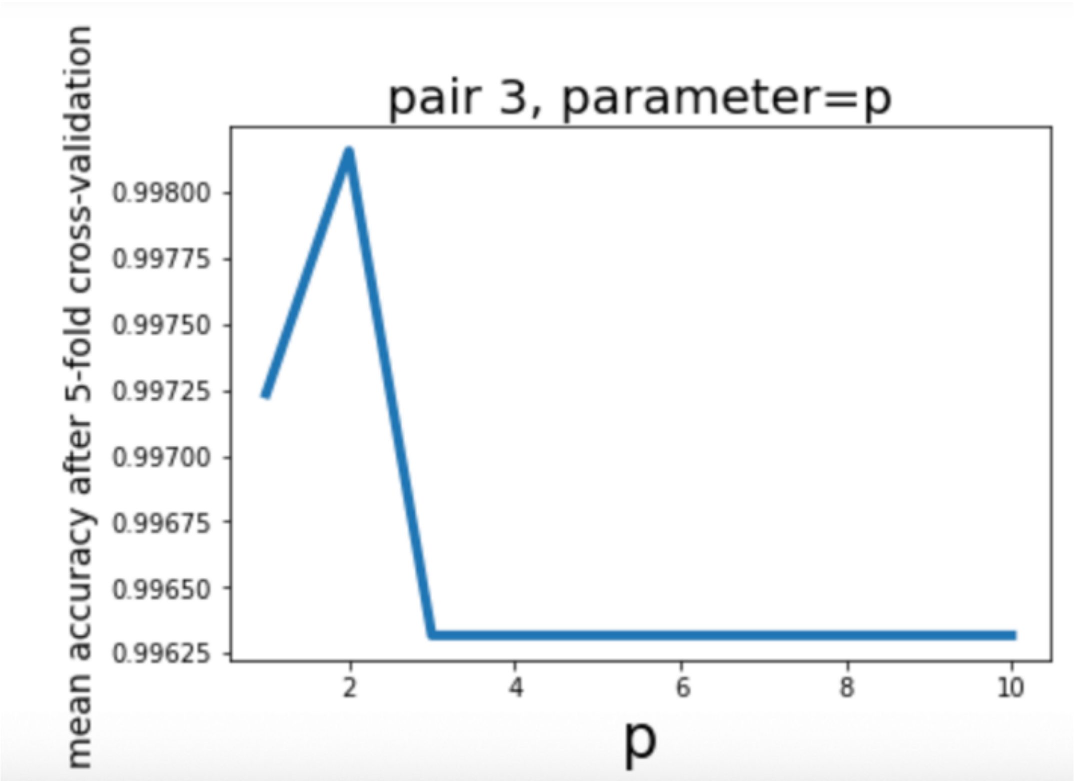

pair 2, parameter=n_neighbors
mean accuracy after 5-fold cross-validation
1.000
0.999
0.998
0.997
0.996
0.995
0.994
0.993
0
5
10
15
20
25
30
n_neighbors


pair 2, parameter=leaf_size
mean accuracy after 5-fold cross-validation
1.04
1.02
1.00
0.98
0.96
0
20
40
60
80
leaf_size

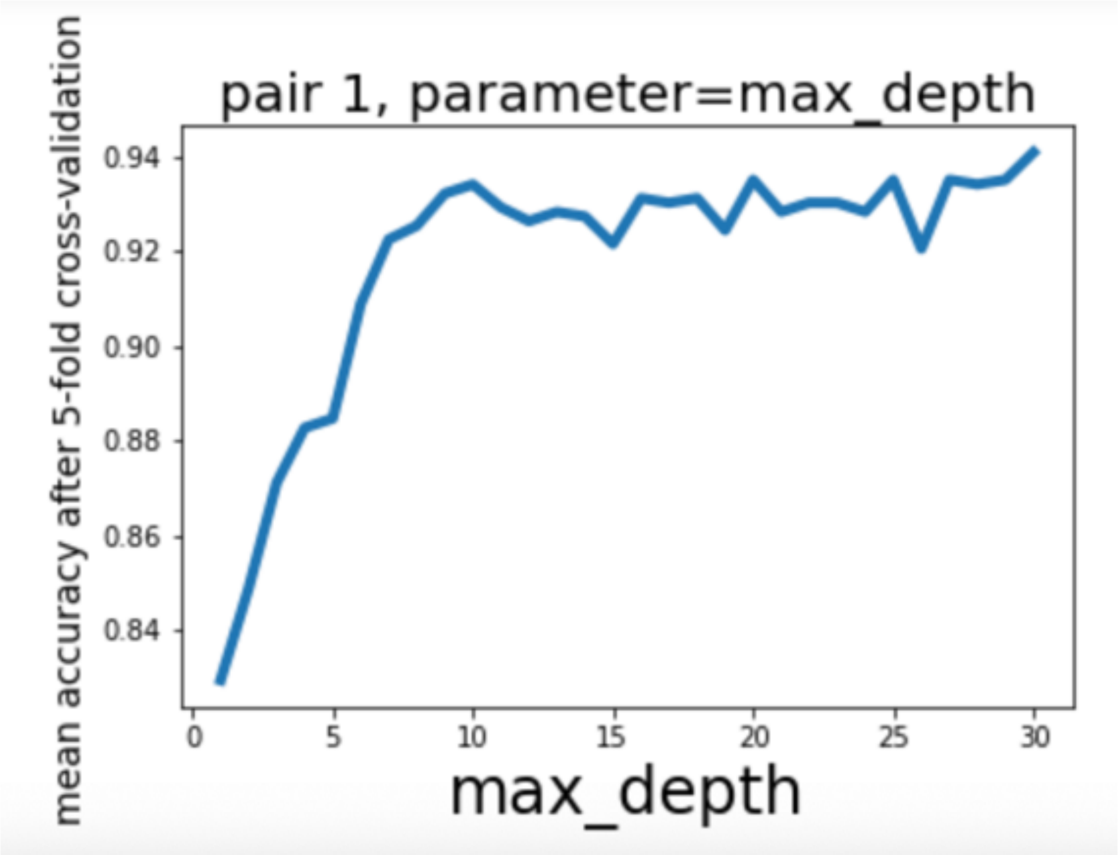

pair 2, parameter=p
1e−6+9.9909000000e−1
mean accuracy after 5-fold cross-validation
5
4
3
2
1
2
4
6
8
10
p

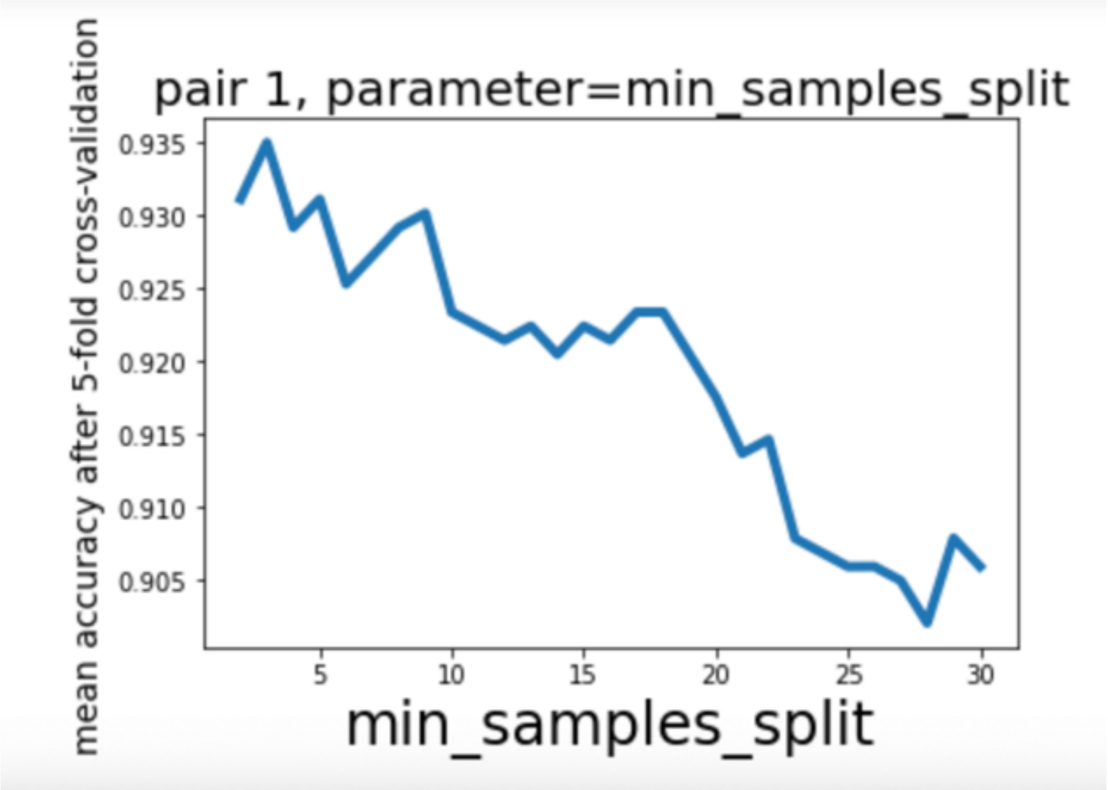

pair 3, parameter=n_neighbors
mean accuracy after 5-fold cross-validation
1.000
0.995
0.990
0.985
0.980
0.975
0
5
10
15
20
25
30
n_neighbors

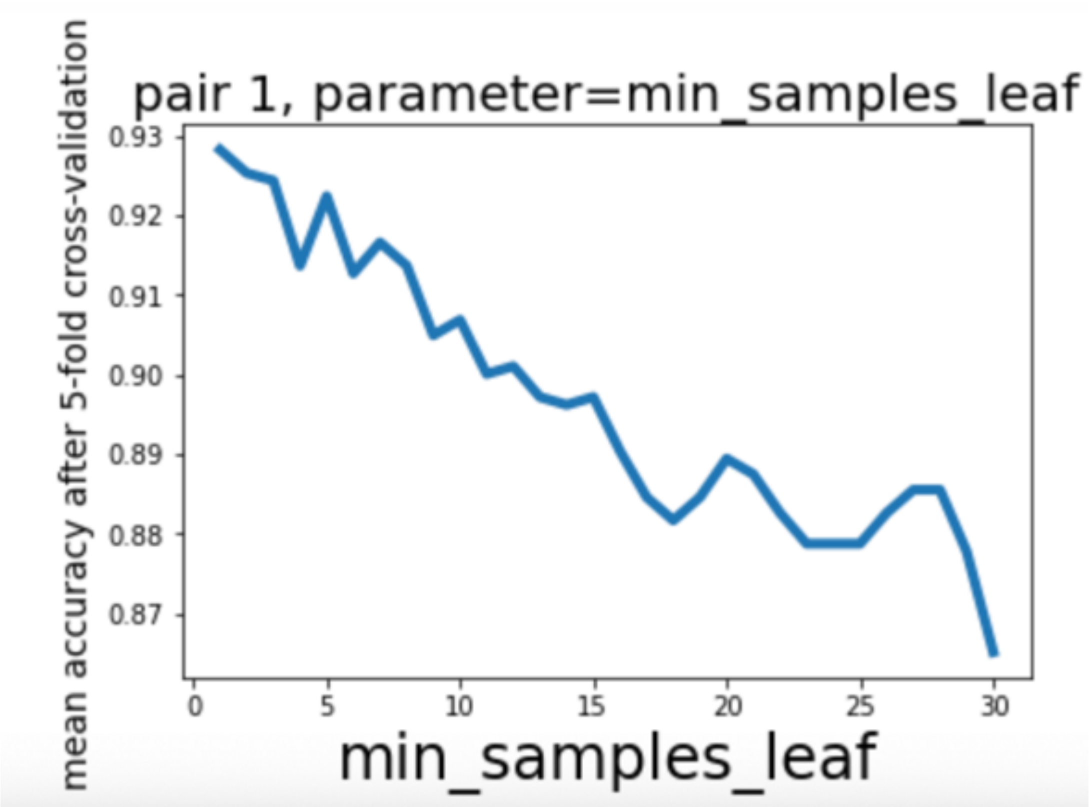

pair 3, parameter=leaf_size
mean accuracy after 5-fold cross-validation
1.04
1.02
1.00
0.98
0.96
0
20
40
60
80
leaf_size

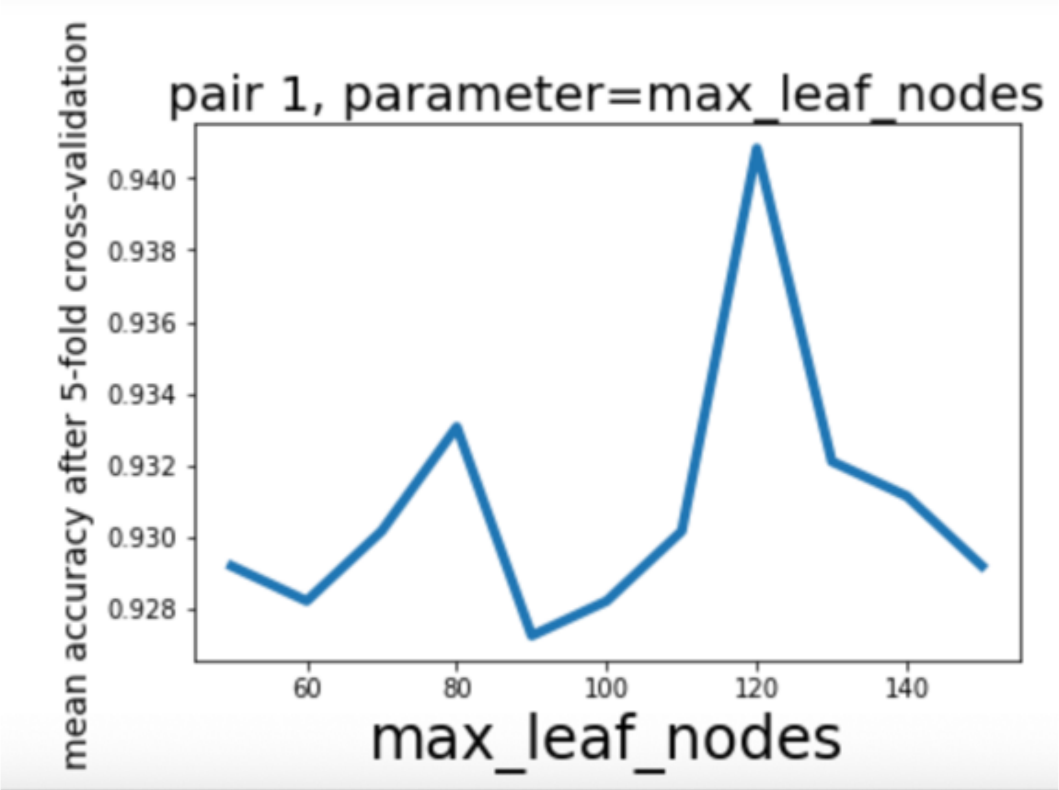

pair 3, parameter=p
mean accuracy after 5-fold cross-validation
0.99800
0.99775
0.99750
0.99725
0.99700
0.99675
0.99650
0.99625
2
4
6
8
10
p

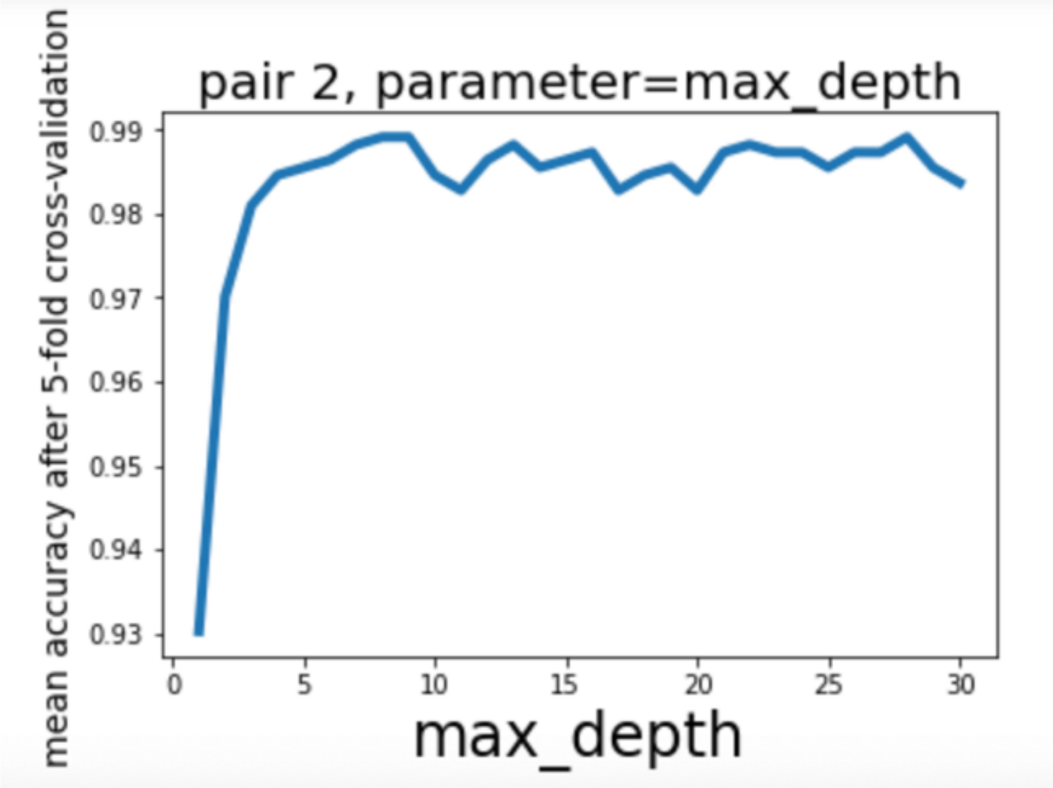
Decision tree

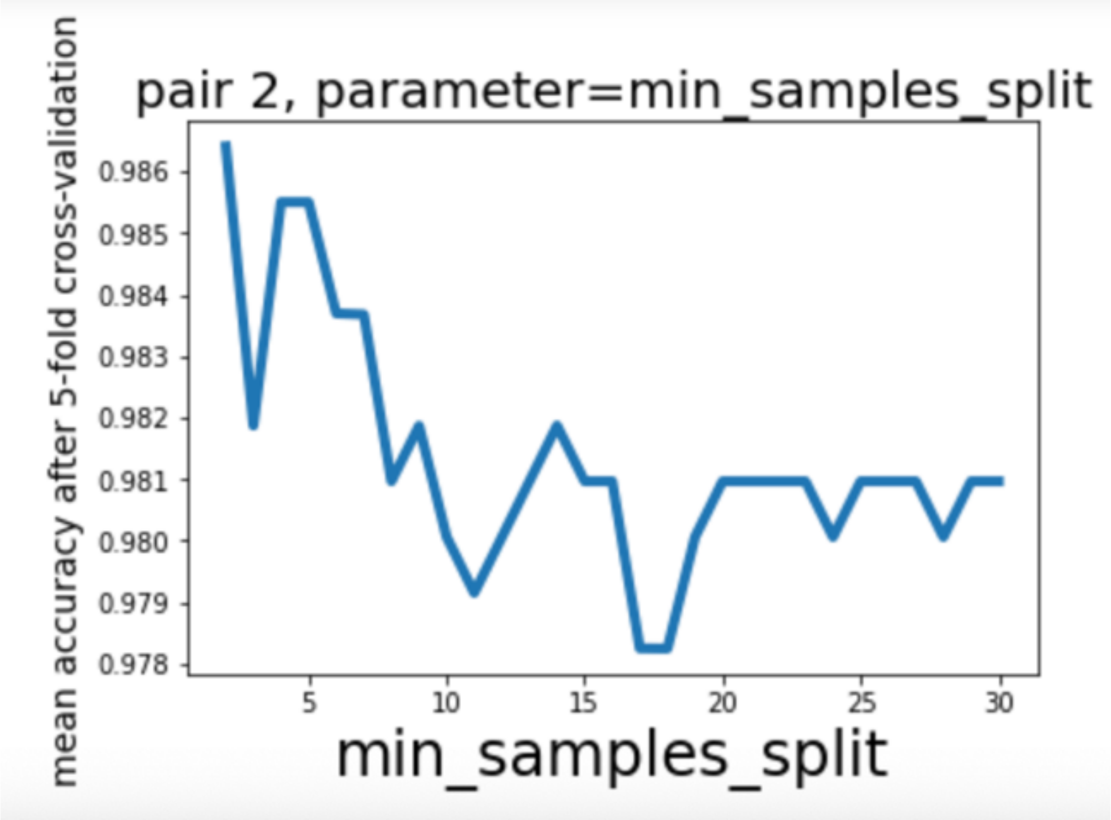


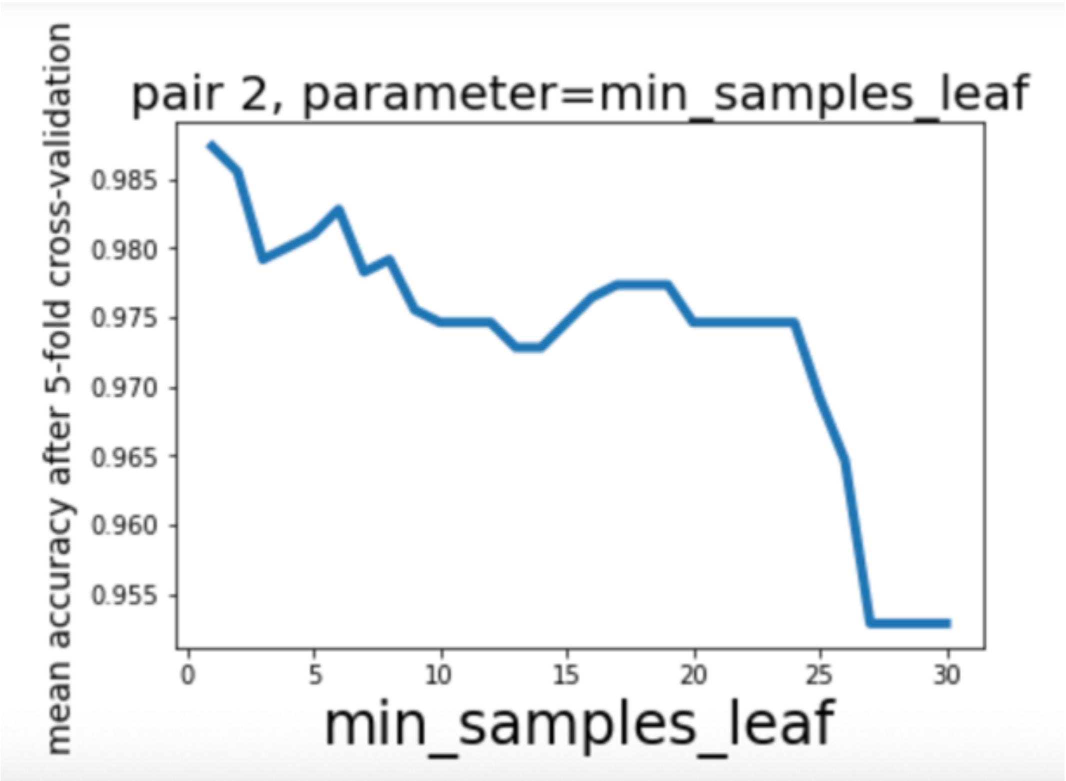


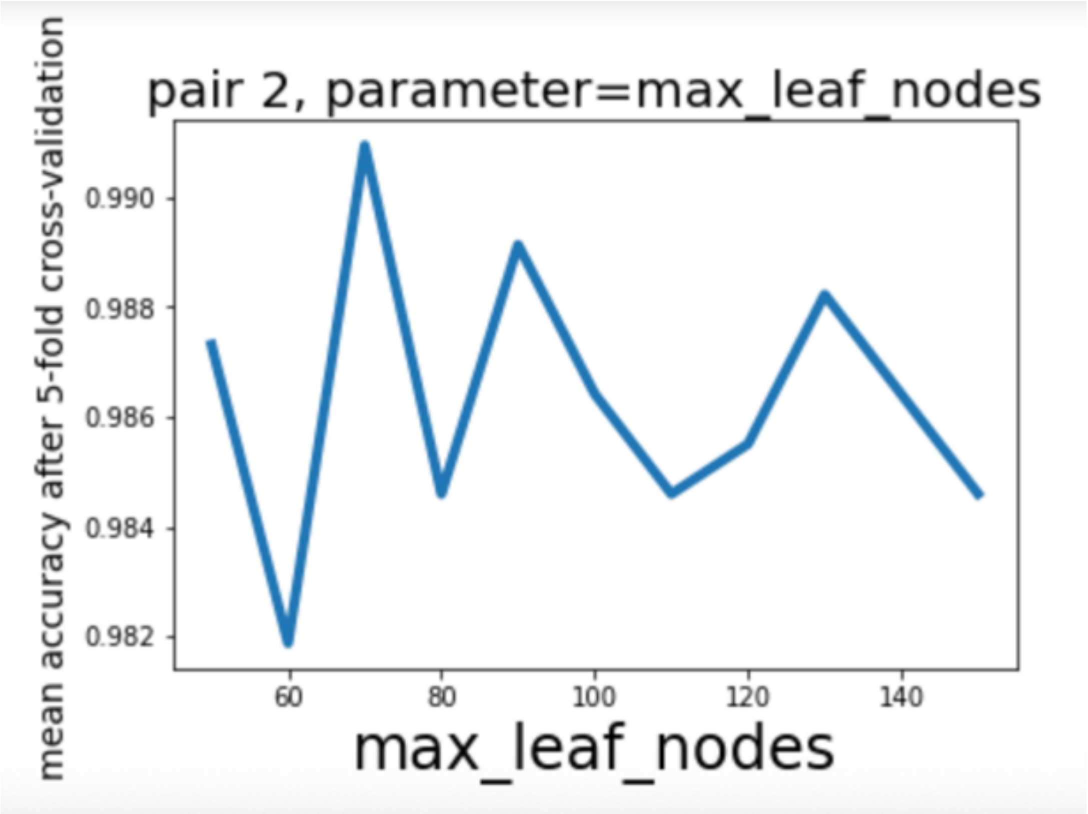

pair 1, parameter=min_samples_leaf
mean accuracy after 5-fold cross-validation
0.93
0.92
0.91
0.90
0.89
0.88
0.87
0
5
10
15
20
25
30
min_samples_leaf

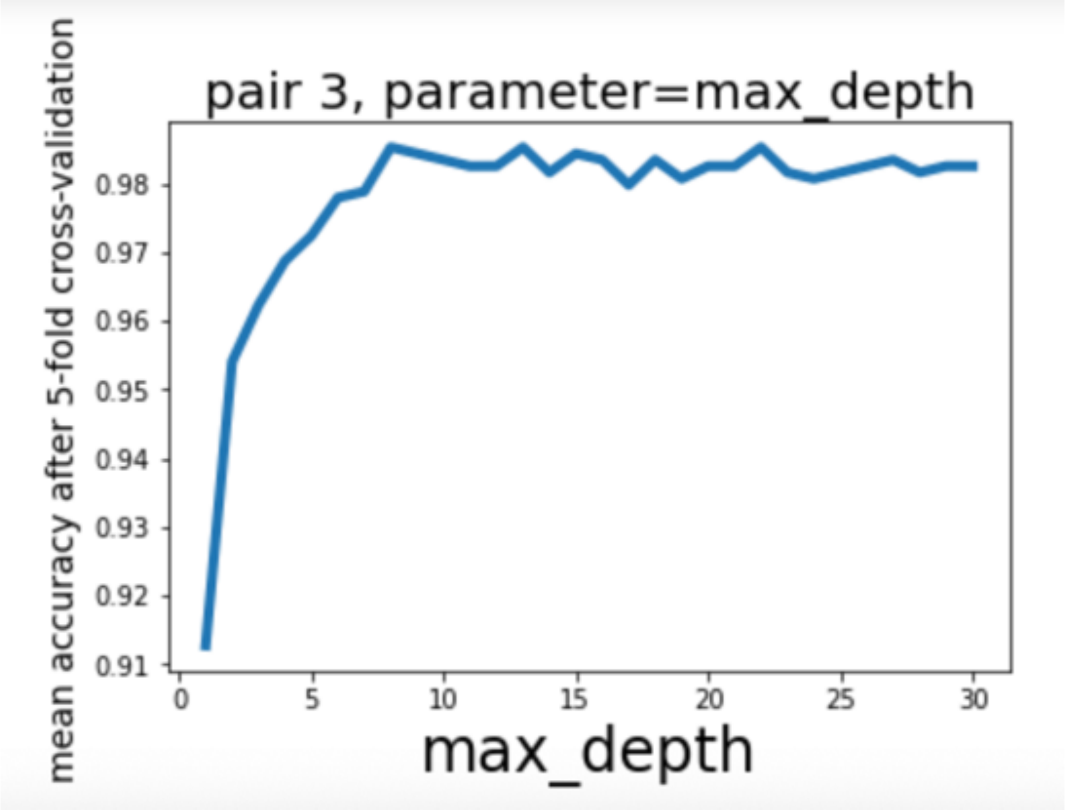

pair 1, parameter=max_leaf_nodes
mean accuracy after 5-fold cross-validation
0.940
0.938
0.936
0.934
0.932
0.930
0.928
60
80
100
120
140
max_leaf_nodes

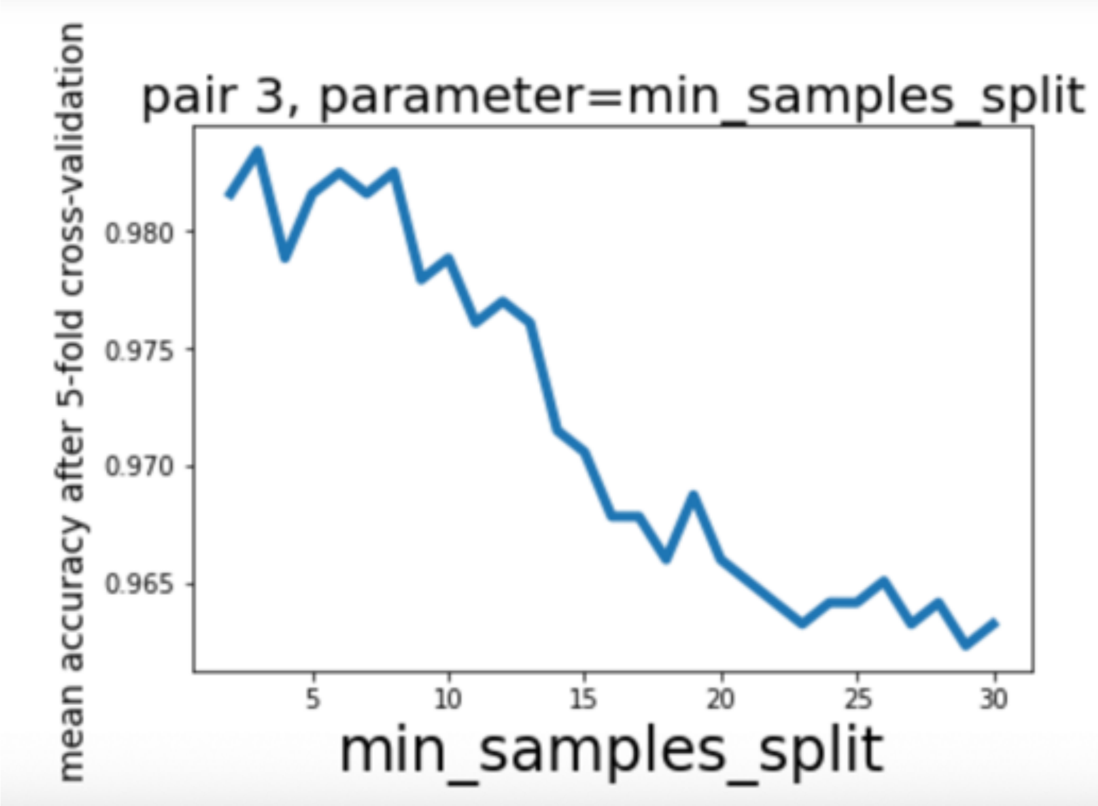

pair 2, parameter=max_depth
mean accuracy after 5-fold cross-validation
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0
5
10
15
20
25
30
max_depth

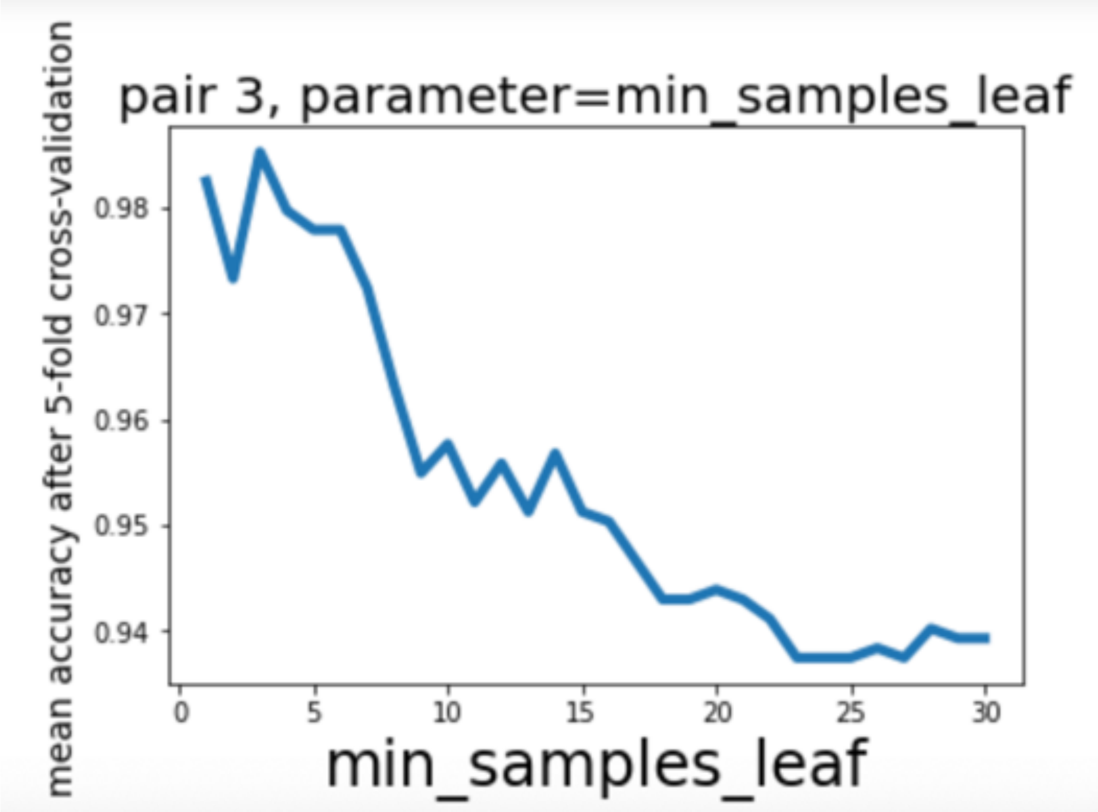

pair 2, parameter=min_samples_split
mean accuracy after 5-fold cross-validation
0.986
0.985
0.984
0.983
0.982
0.981
0.980
0.979
0.978
5
10
15
20
25
30
min_samples_split

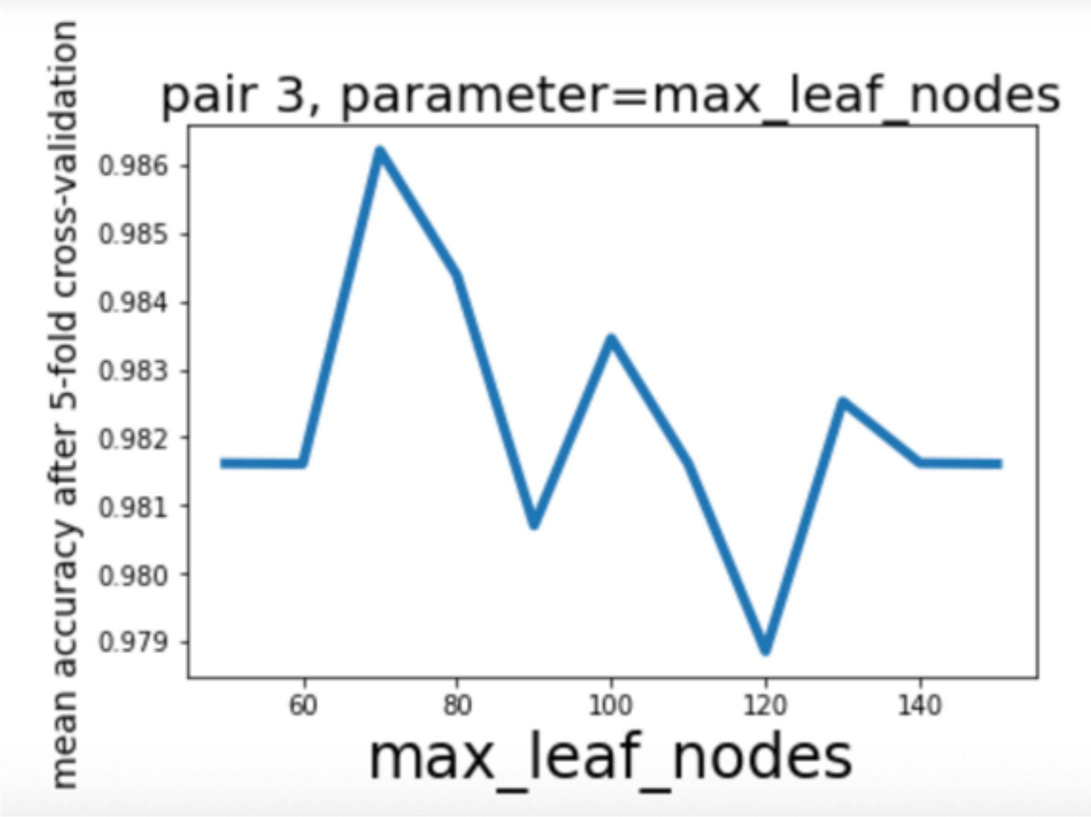

pair 2, parameter=min_samples_leaf
mean accuracy after 5-fold cross-validation
0.985
0.980
0.975
0.970
0.965
0.960
0.955
0
5
10
15
20
25
30
min_samples_leaf


pair 2, parameter=max_leaf_nodes
mean accuracy after 5-fold cross-validation
0.990
0.988
0.986
0.984
0.982
60
80
100
120
140
max_leaf_nodes

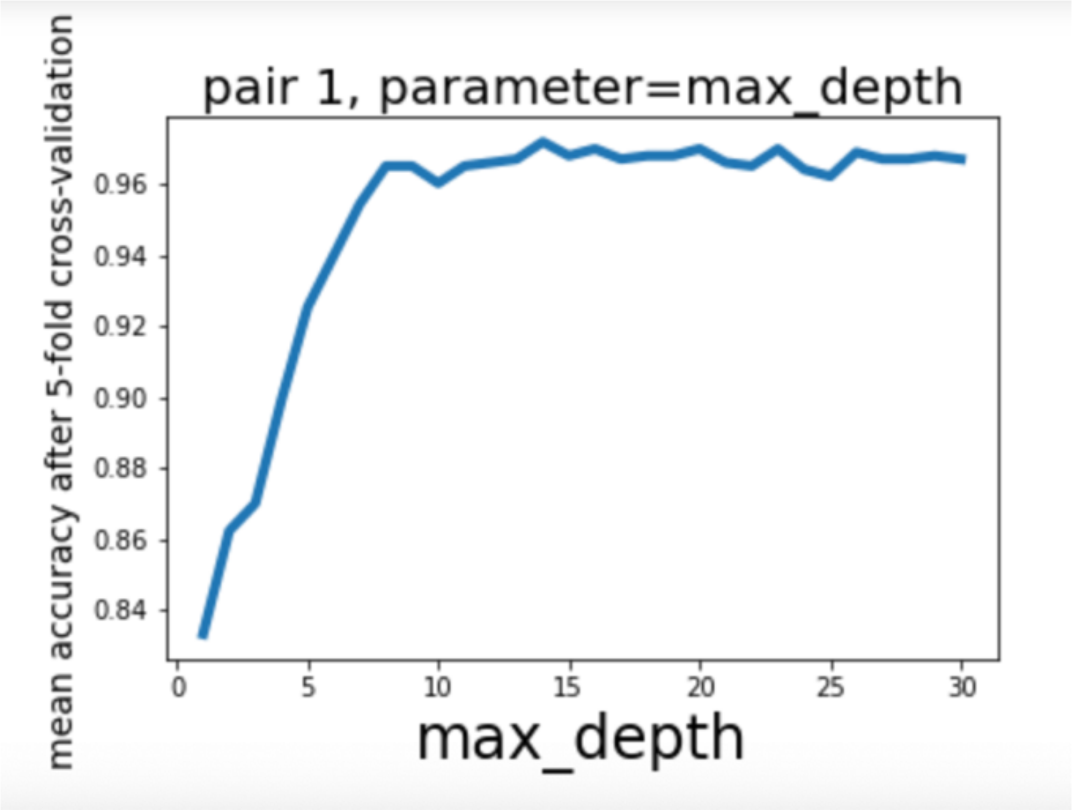

pair 3, parameter=max_depth
mean accuracy after 5-fold cross-validation
0.91
0.92
0.93
0.94
0.95
0.96
0.97
0.98
0
5
10
15
20
25
30
max_depth

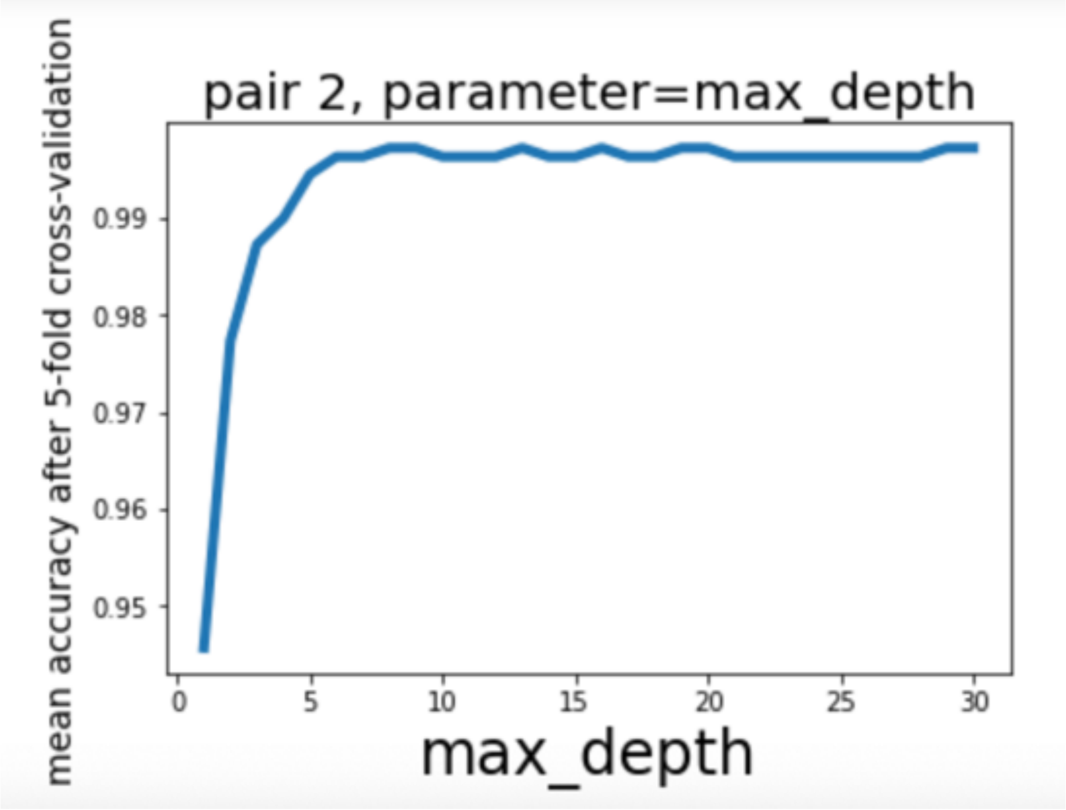

pair 3, parameter=min_samples_split
mean accuracy after 5-fold cross-validation
0.965
0.970
0.975
0.980
5
10
15
20
25
30
min_samples_split

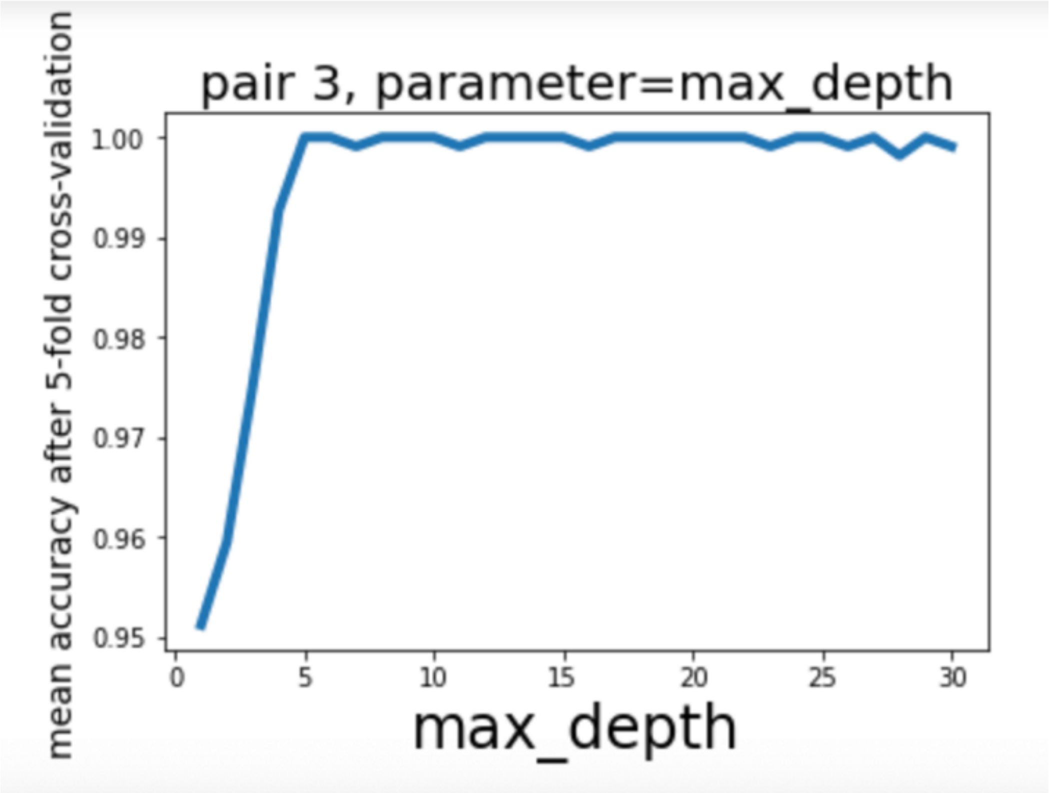

pair 3, parameter=min_samples_leaf
mean accuracy after 5-fold cross-validation
0.98
0.97
0.96
0.95
0.94
0
5
10
15
20
25
30
min_samples_leaf


pair 3, parameter=max_leaf_nodes
mean accuracy after 5-fold cross-validation
0.986
0.985
0.984
0.983
0.982
0.981
0.980
0.979
60
80
100
120
140
max_leaf_nodes

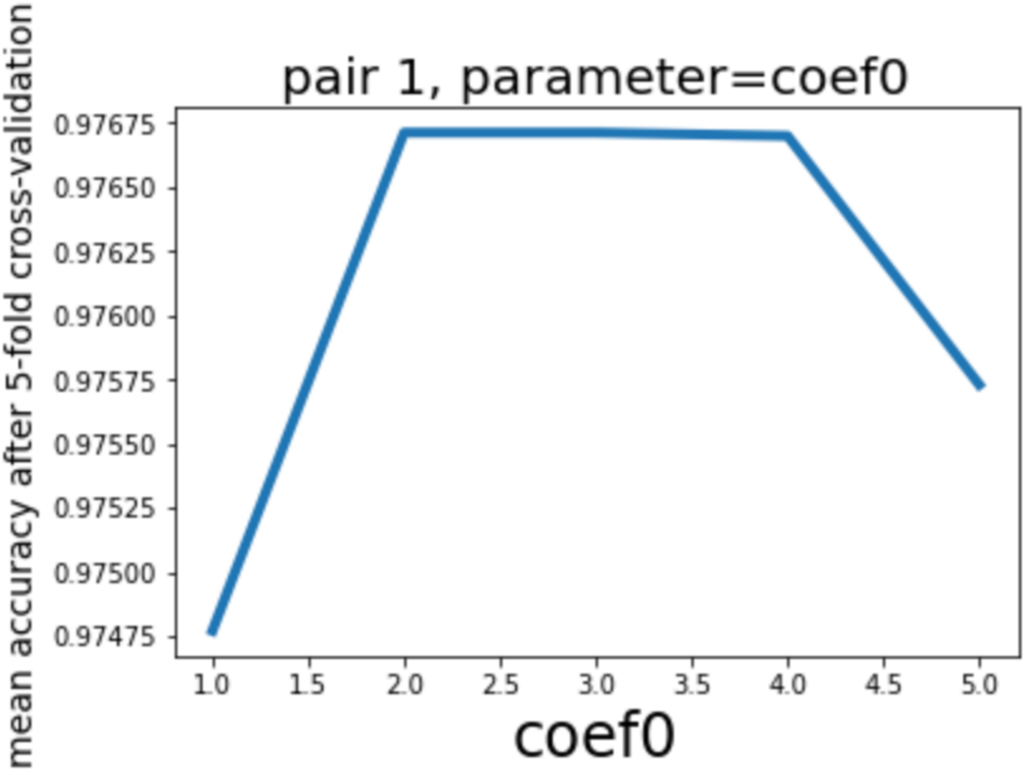
Random forest

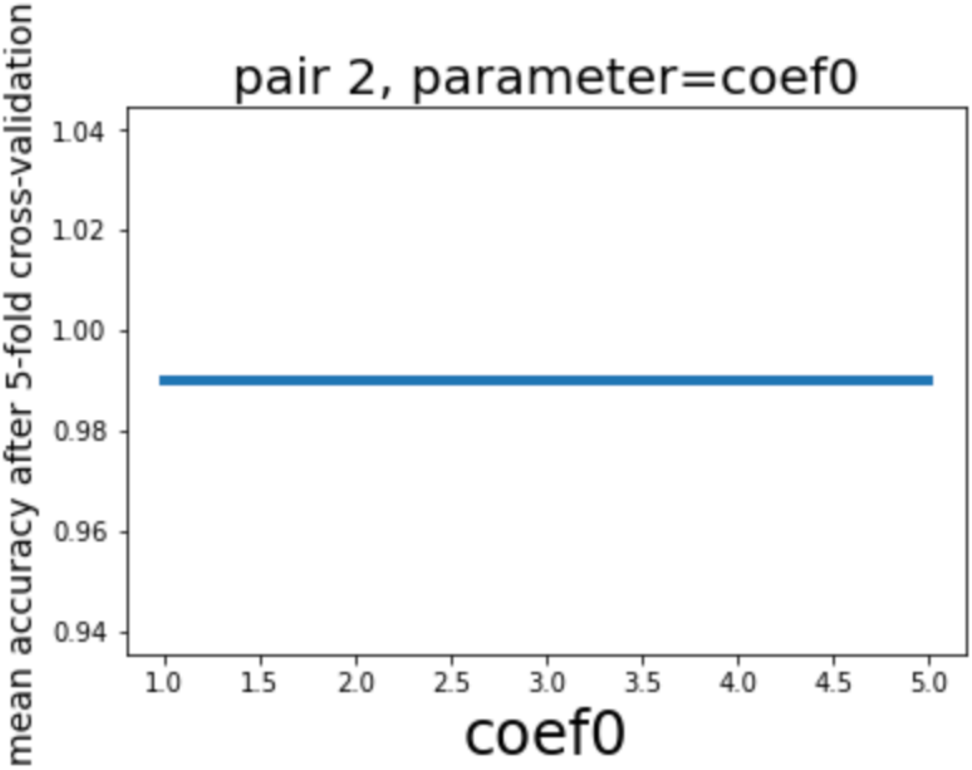


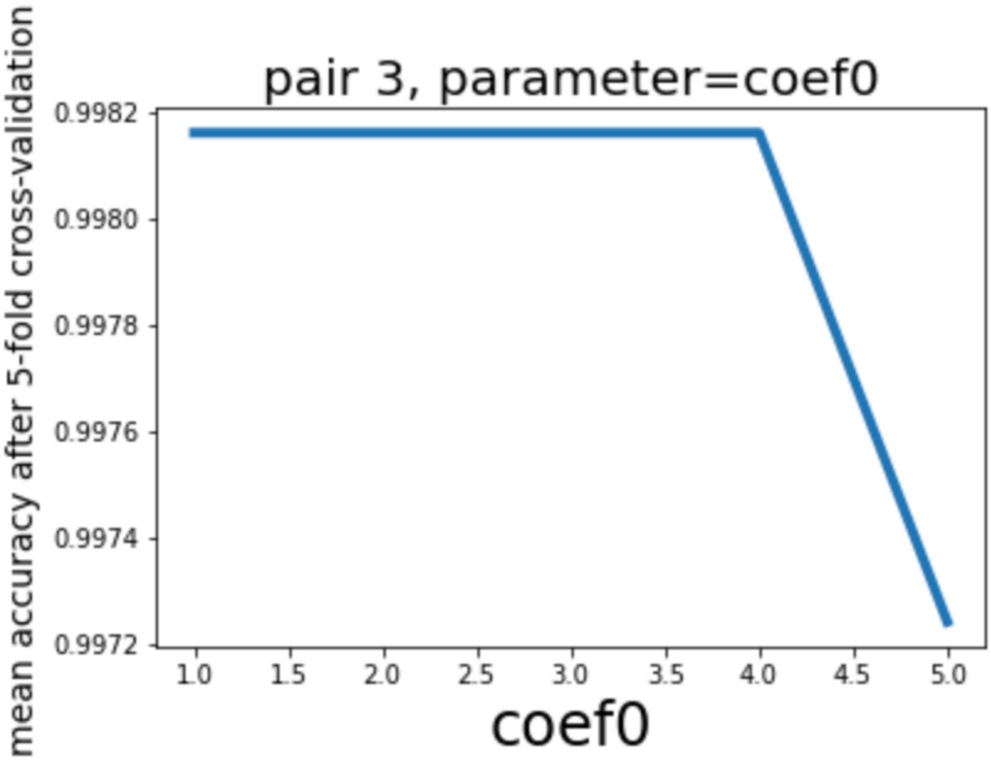



pair 3, parameter=max_depth
mean accuracy after 5-fold cross-validation
1.00
0.99
0.98
0.97
0.96
0.95
0
5
10
15
20
25
30
max_depth

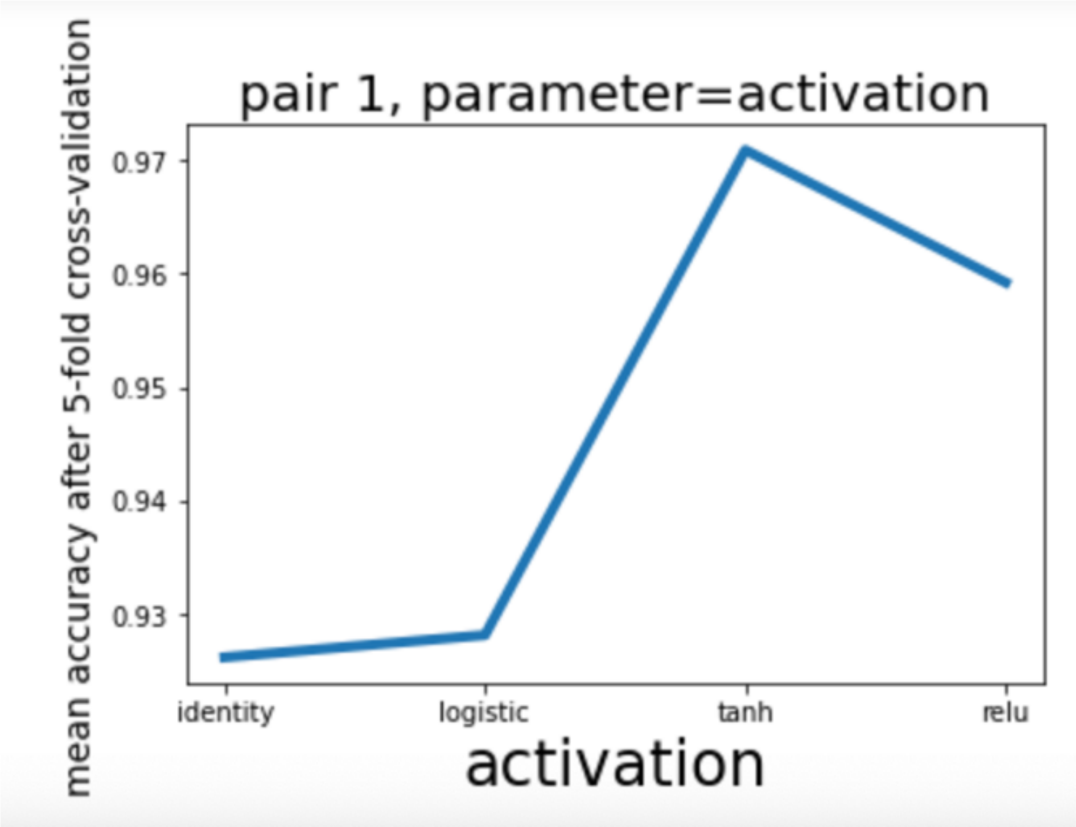
SVM


pair 1, parameter=coef0
mean accuracy after 5-fold cross-validation
0.97675
0.97650
0.97625
0.97600
0.97575
0.97550
0.97525
0.97500
0.97475
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
coef0

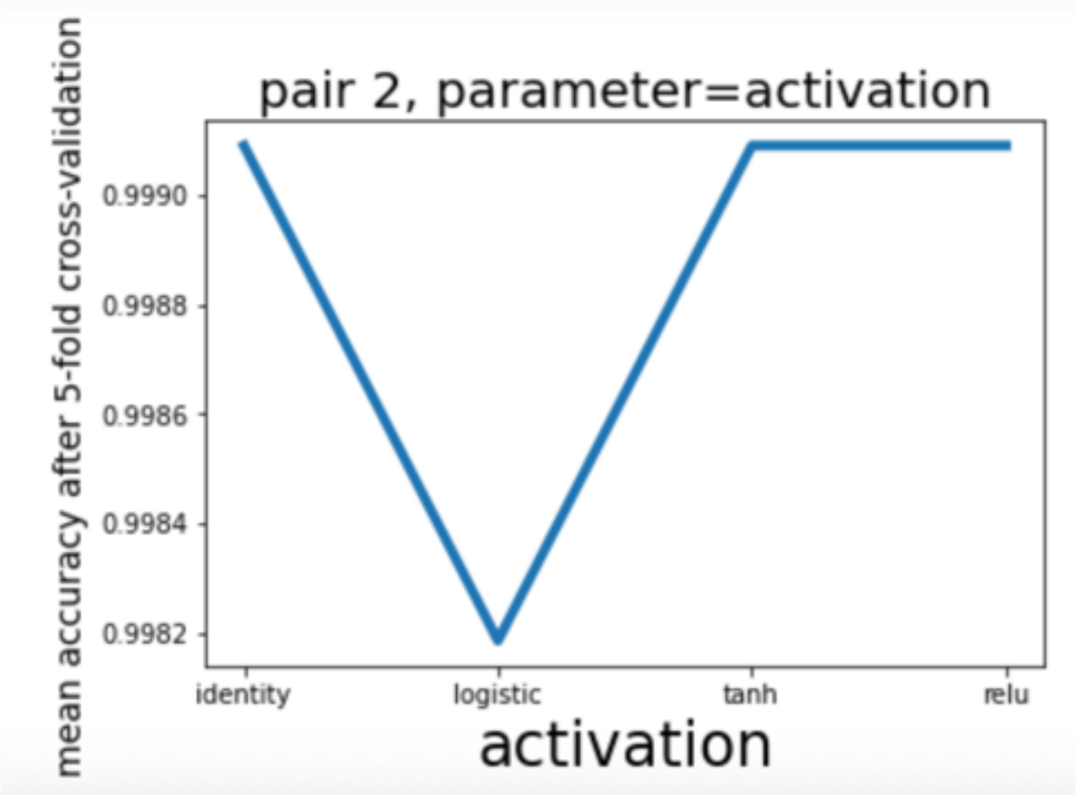

pair 2, parameter=coef0
mean accuracy after 5-fold cross-validation
1.04
1.02
1.00
0.98
0.96
0.94
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
coef0


pair 3, parameter=coef0
mean accuracy after 5-fold cross-validation
0.9982
0.9980
0.9978
0.9976
0.9974
0.9972
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
coef0

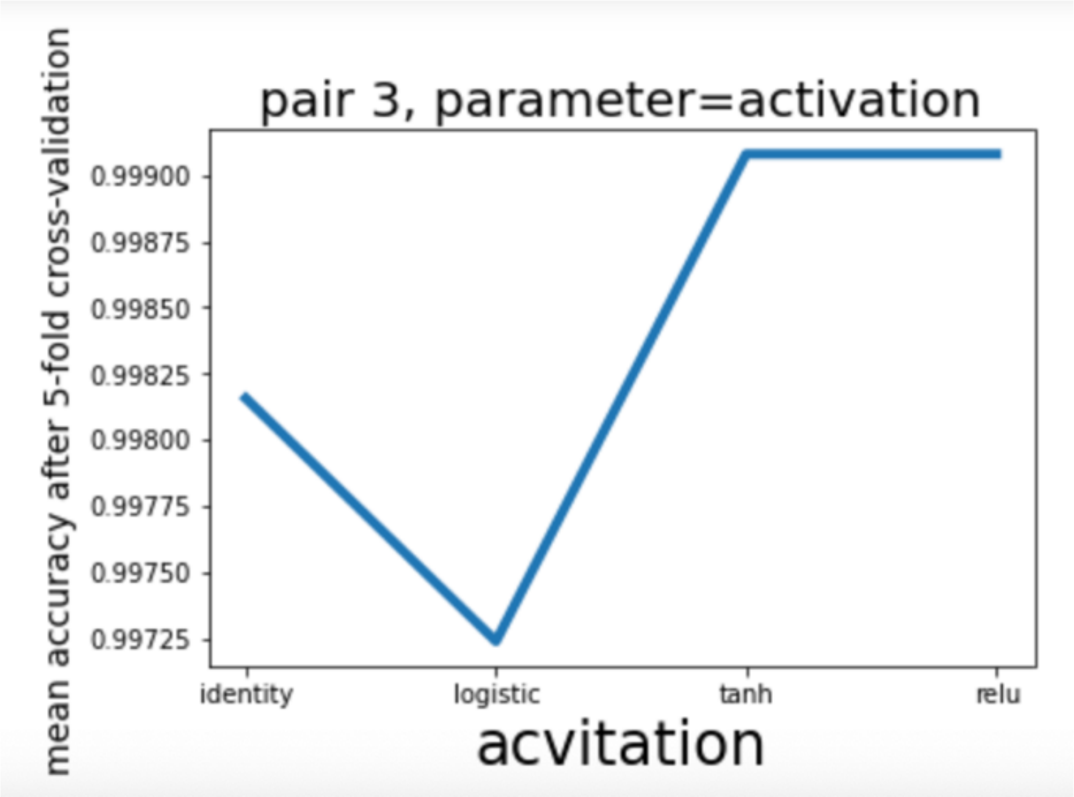
ANN



pair 1, parameter=activation
mean accuracy after 5-fold cross-validation
0.97
0.96
0.95
0.94
0.93
identity
logistic
tanh
relu
activation



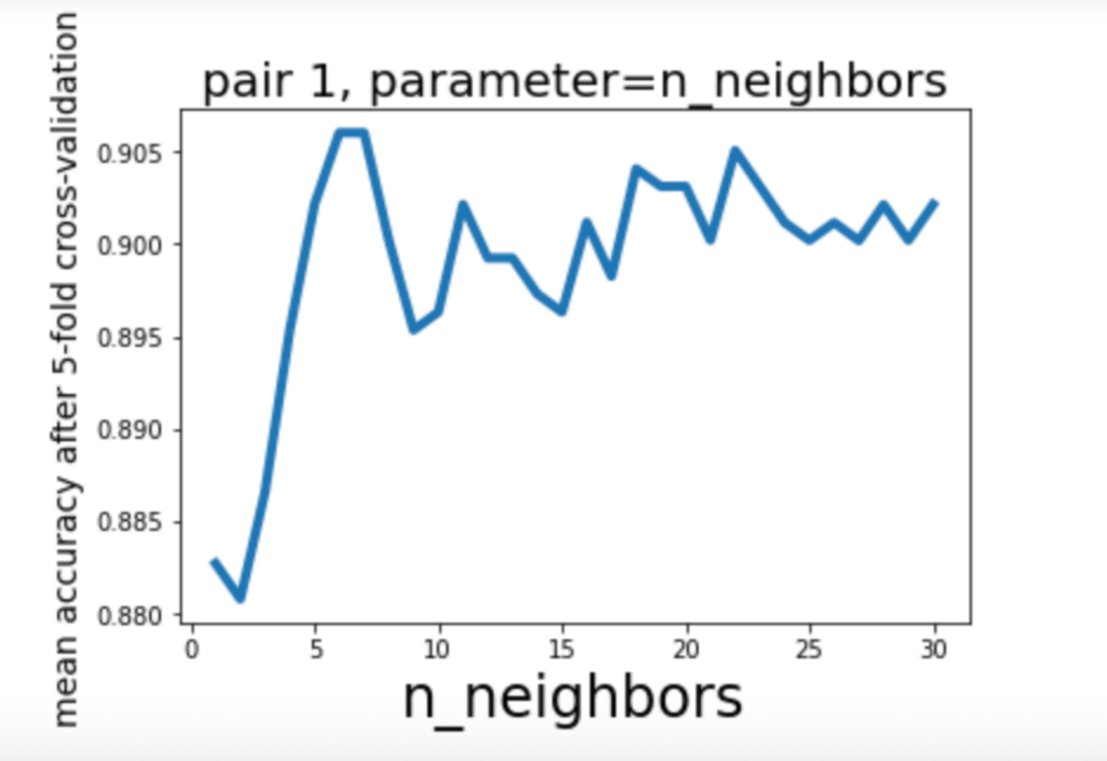
pair 2, parameter=activation
mean accuracy after 5-fold cross-validation
0.9990
0.9988
0.9986
0.9984
0.9982
identity
logistic
tanh
relu
activation

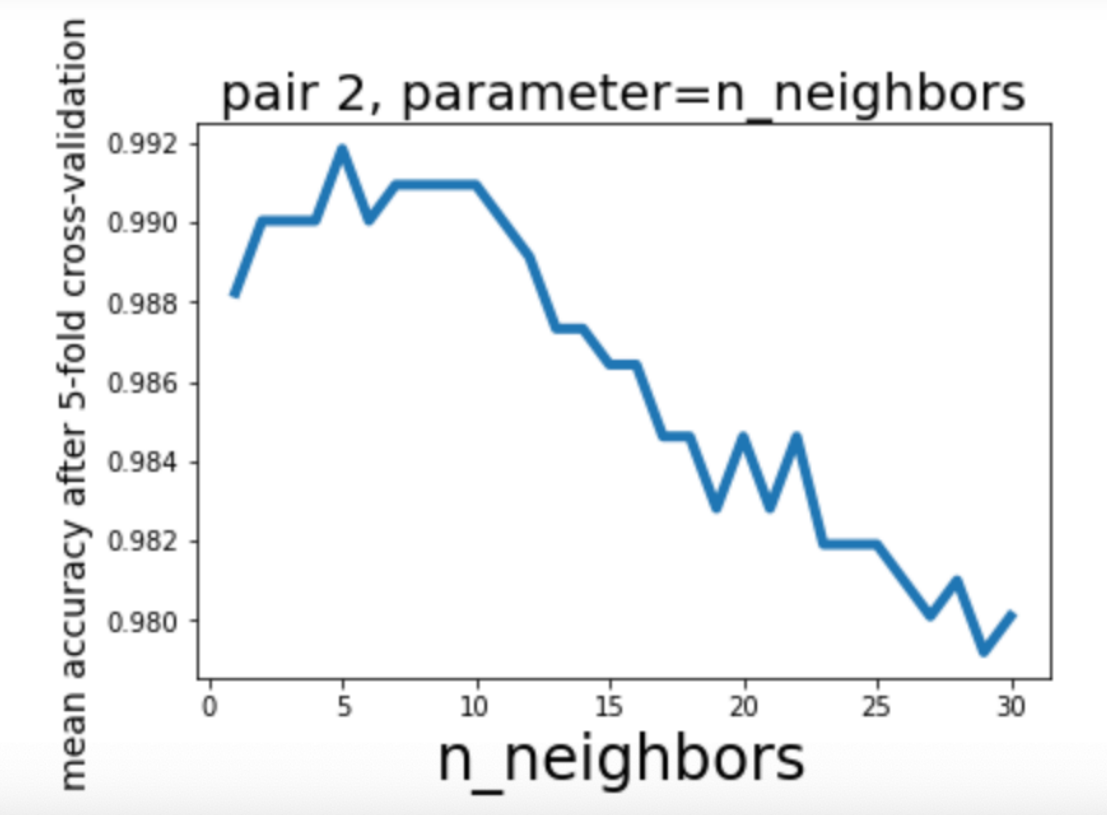

pair 3, parameter=activation
mean accuracy after 5-fold cross-validation
0.99900
0.99875
0.99850
0.99825
0.99800
0.99775
0.99750
0.99725
identity
logistic
tanh
relu
acvitation

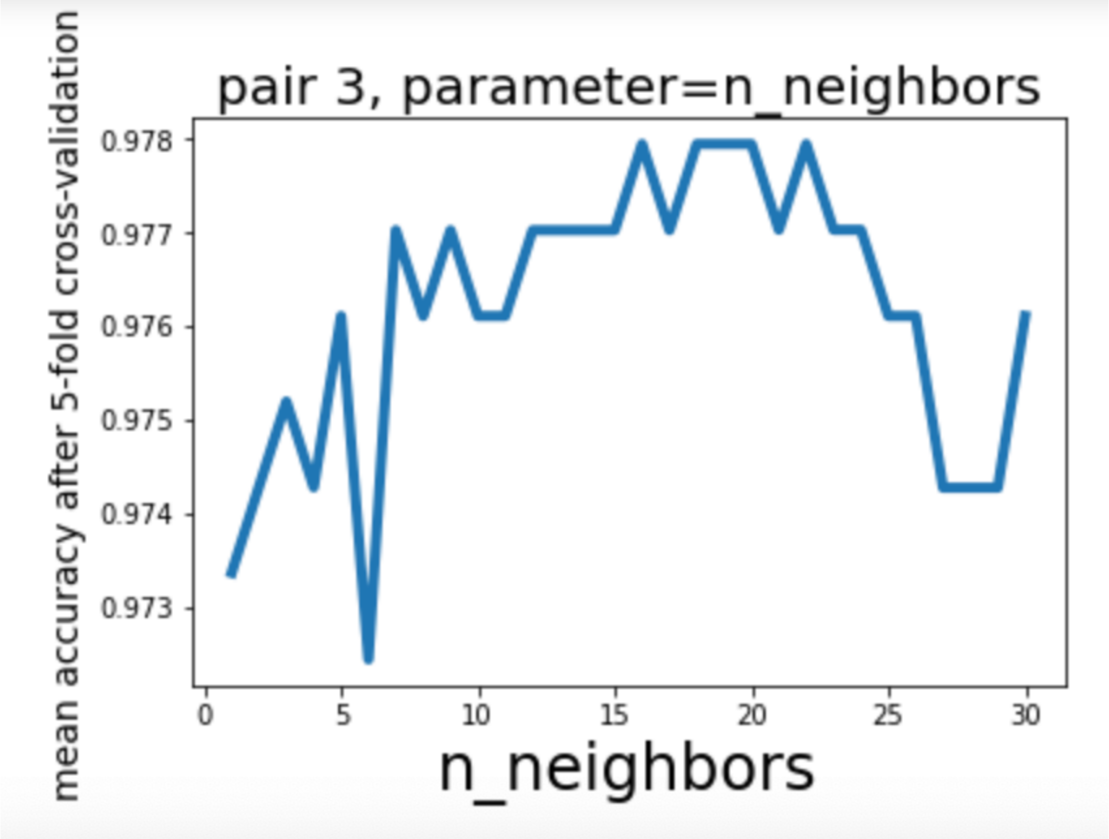
Dimension reduction methods used

The three methods I used are filter method--f_score, Wrapper method--forward feature selection, and Embedded Method—random forest.

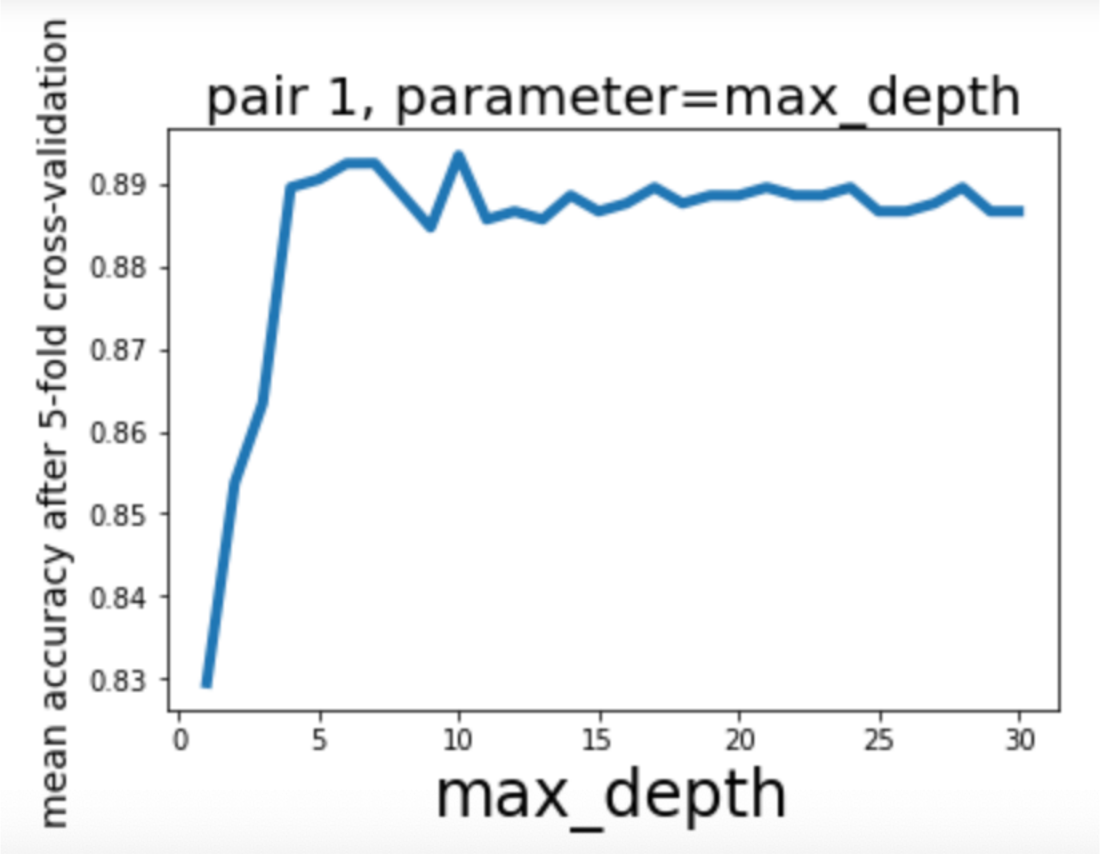
KNN

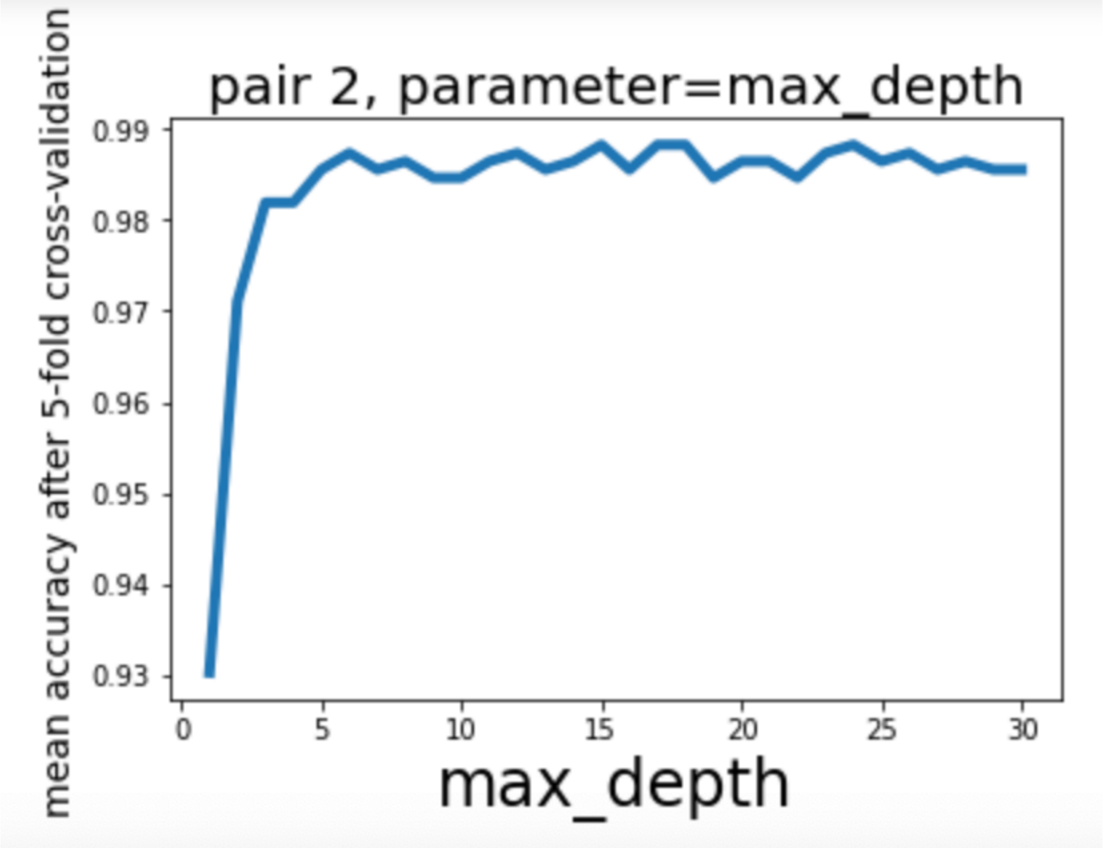


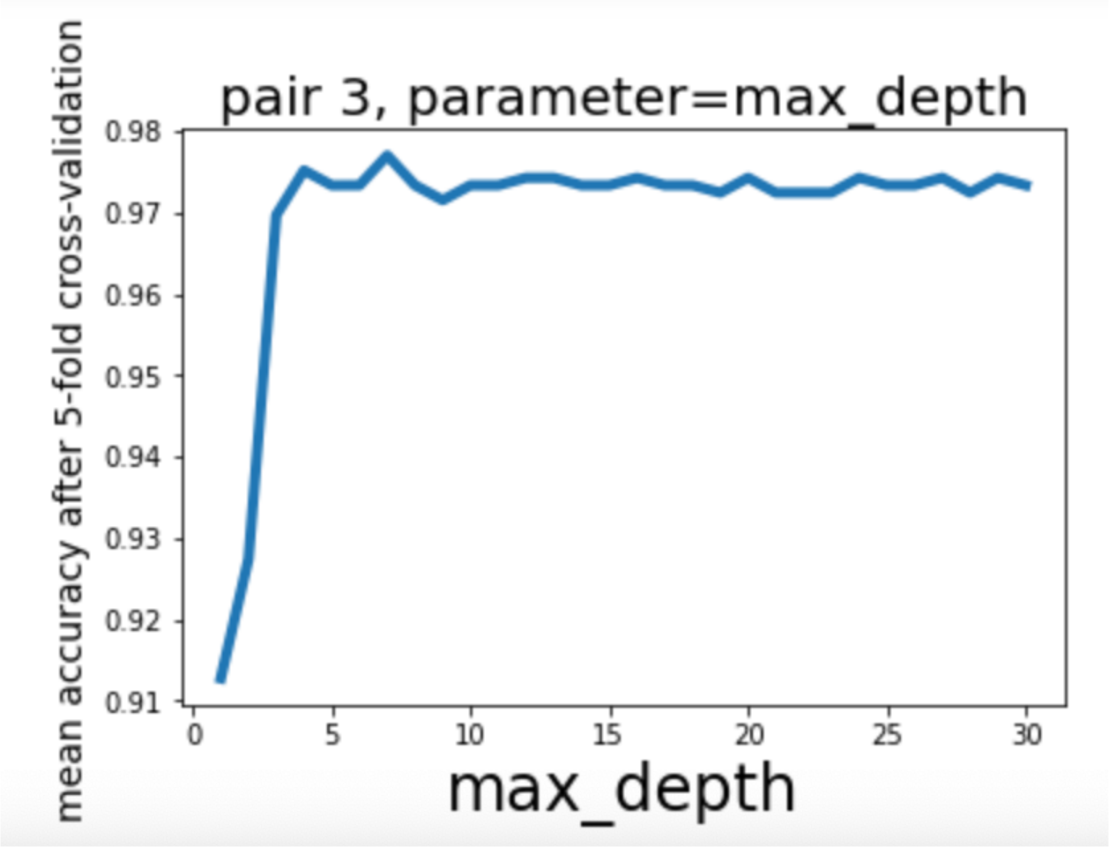



pair 3, parameter=n_neighbors
mean accuracy after 5-fold cross-validation
0.978
0.977
0.976
0.975
0.974
0.973
0
5
10
15
20
25
30
n_neighbors

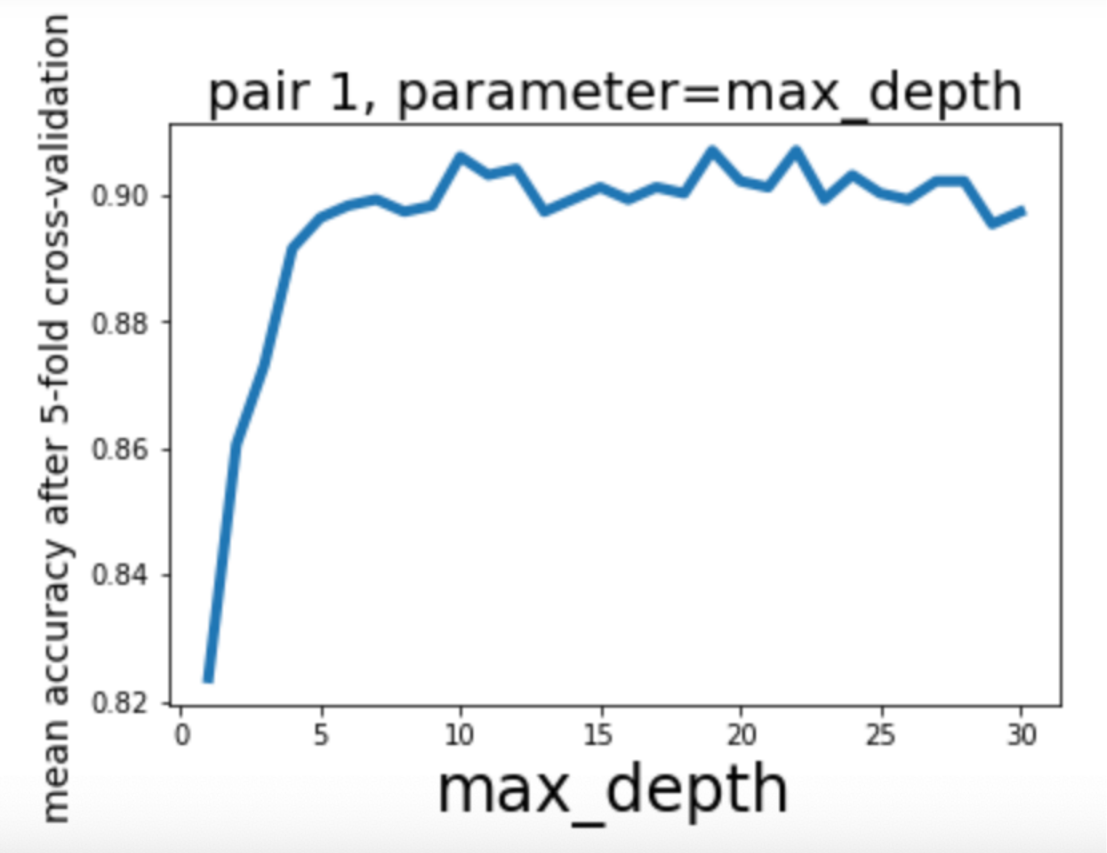
Decision tree

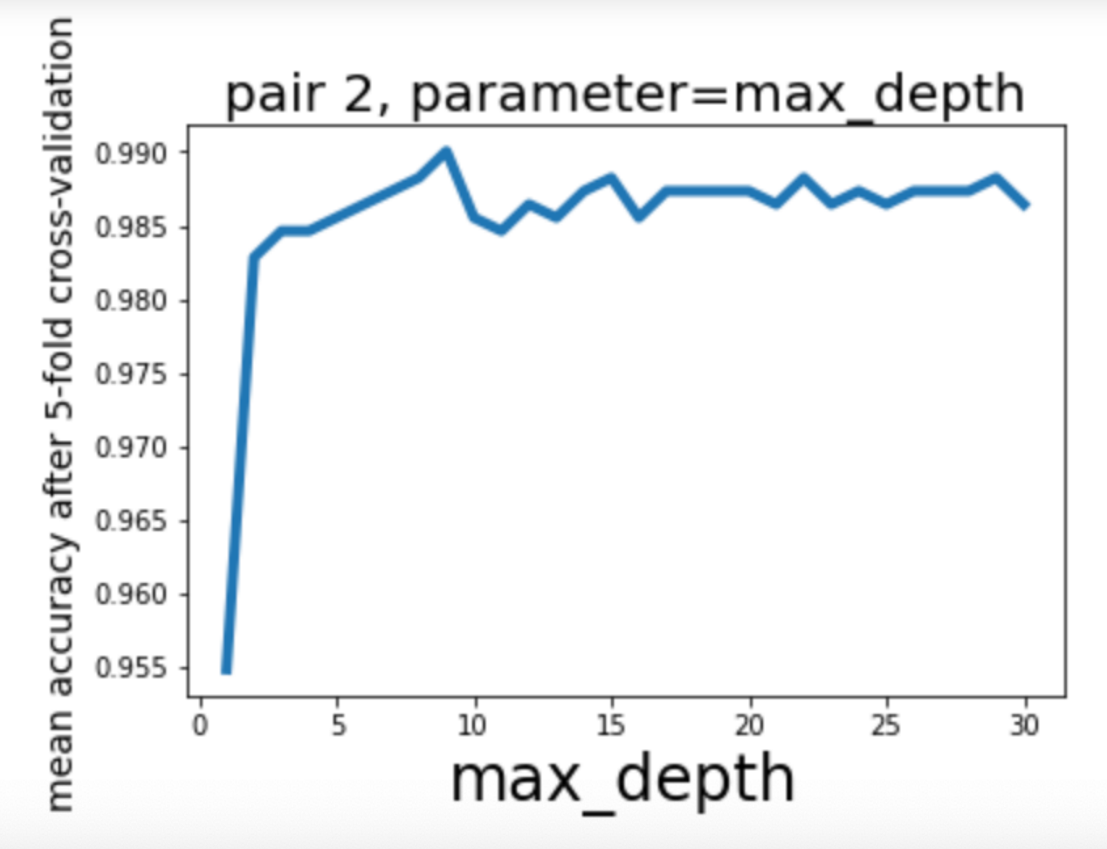


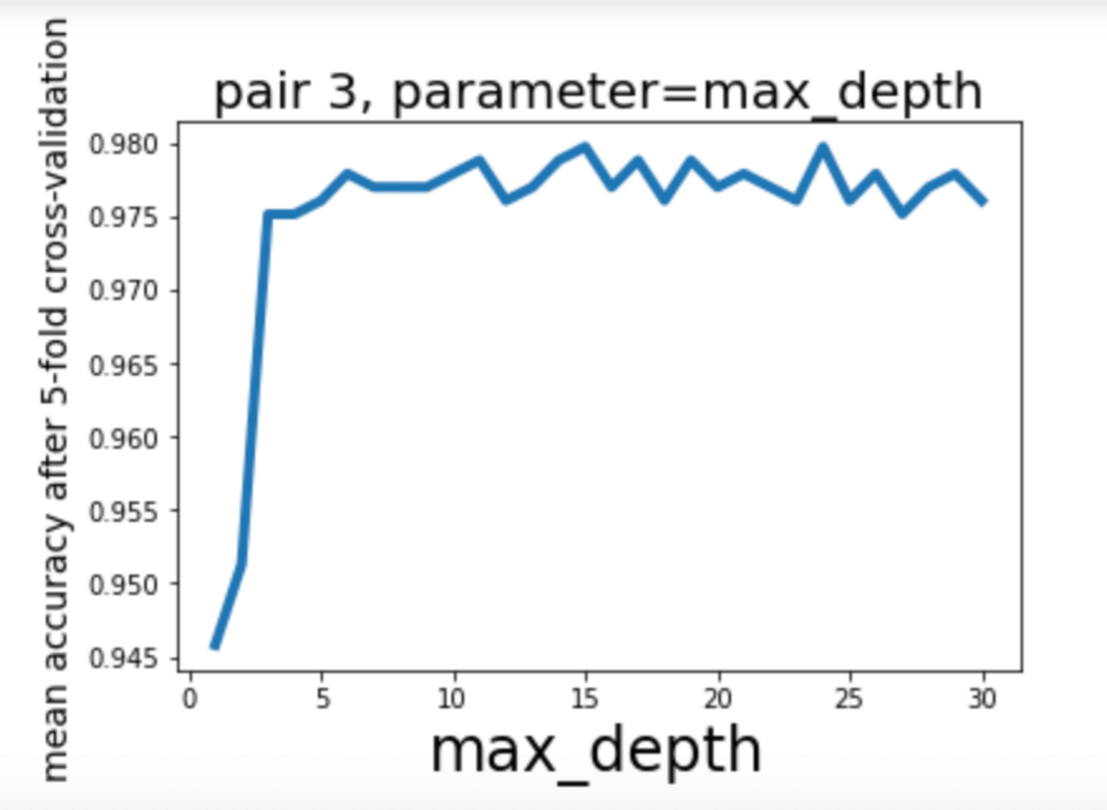



pair 3, parameter=max_depth
mean accuracy after 5-fold cross-validation
0.98
0.97
0.96
0.95
0.94
0.93
0.92
0.91
0
5
10
15
20
25
30
max_depth

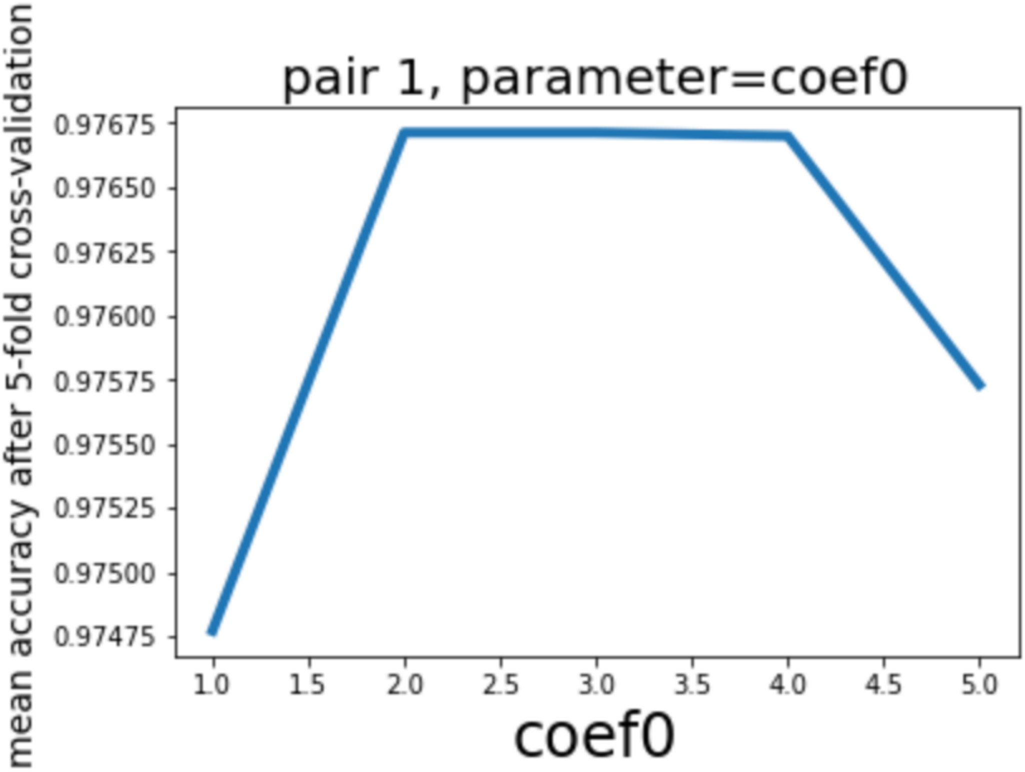
Random forest

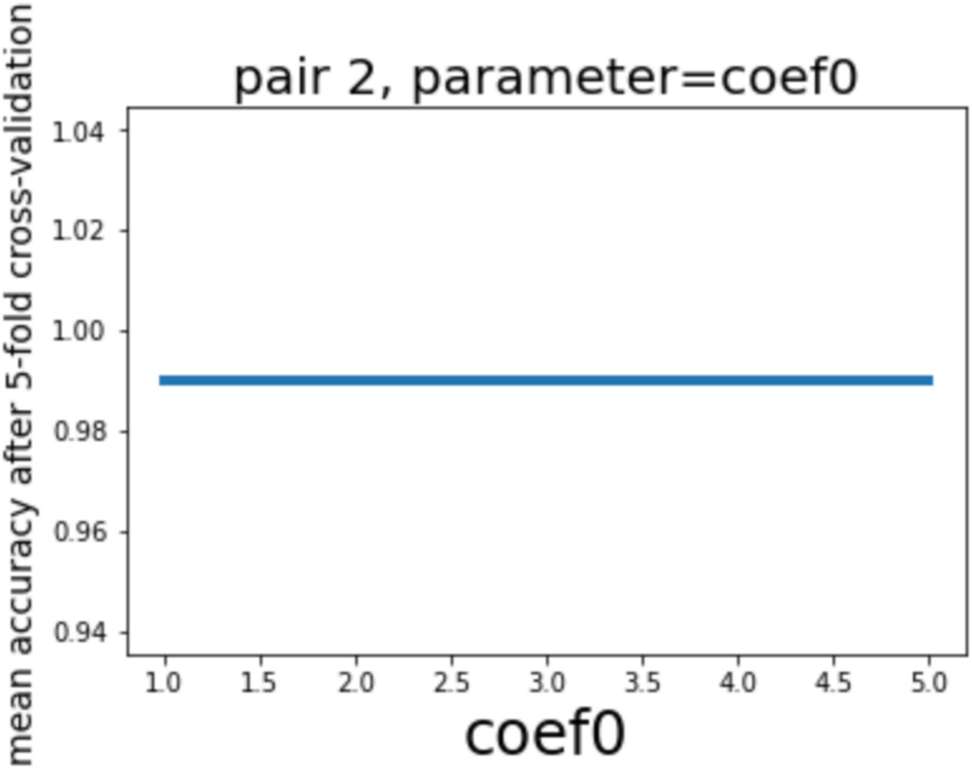


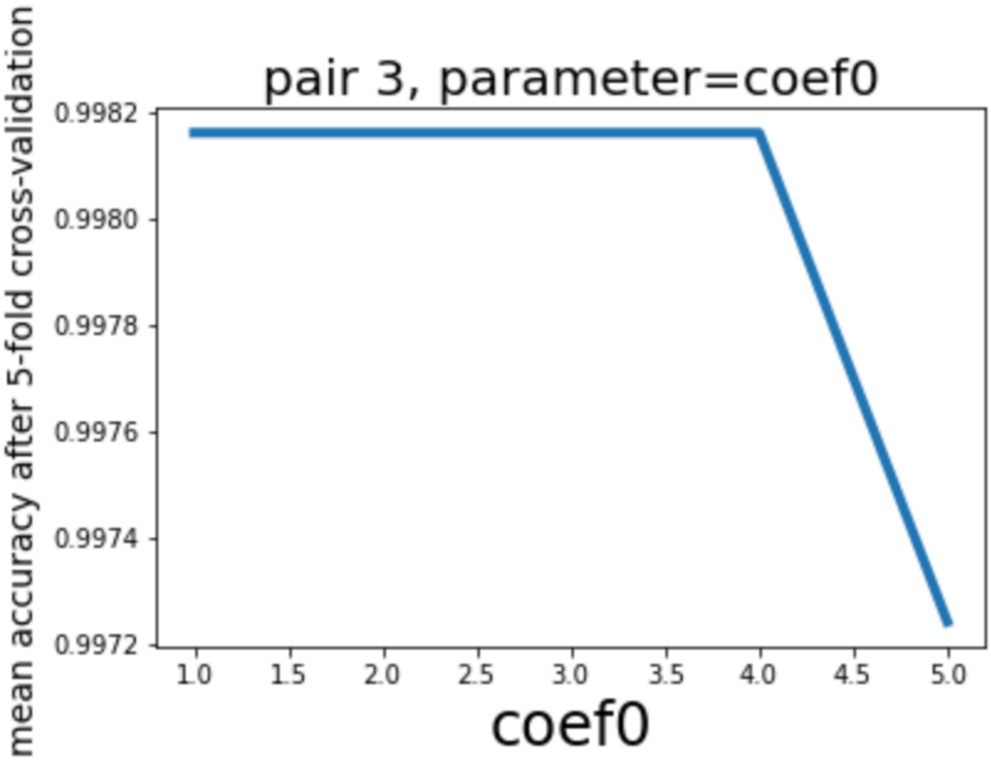



pair 3, parameter=max_depth
mean accuracy after 5-fold cross-validation
0.980
0.975
0.970
0.965
0.960
0.955
0.950
0.945
0
5
10
15
20
25
30
max_depth

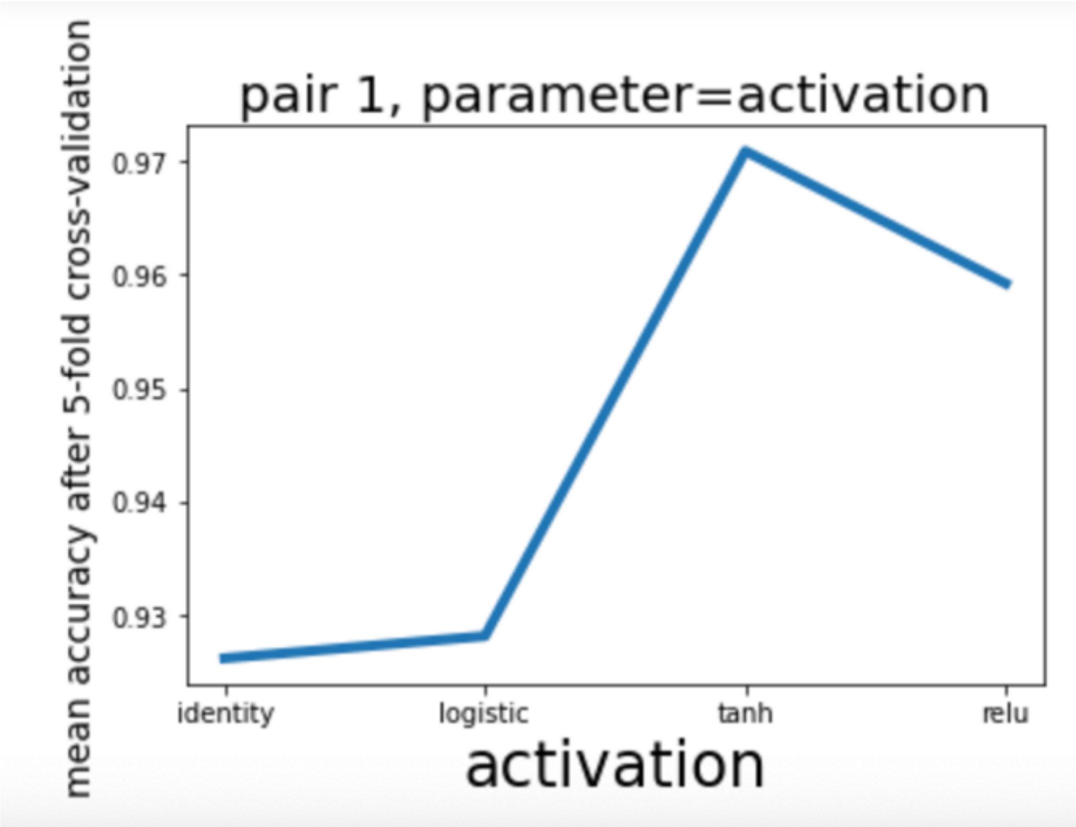
SVM

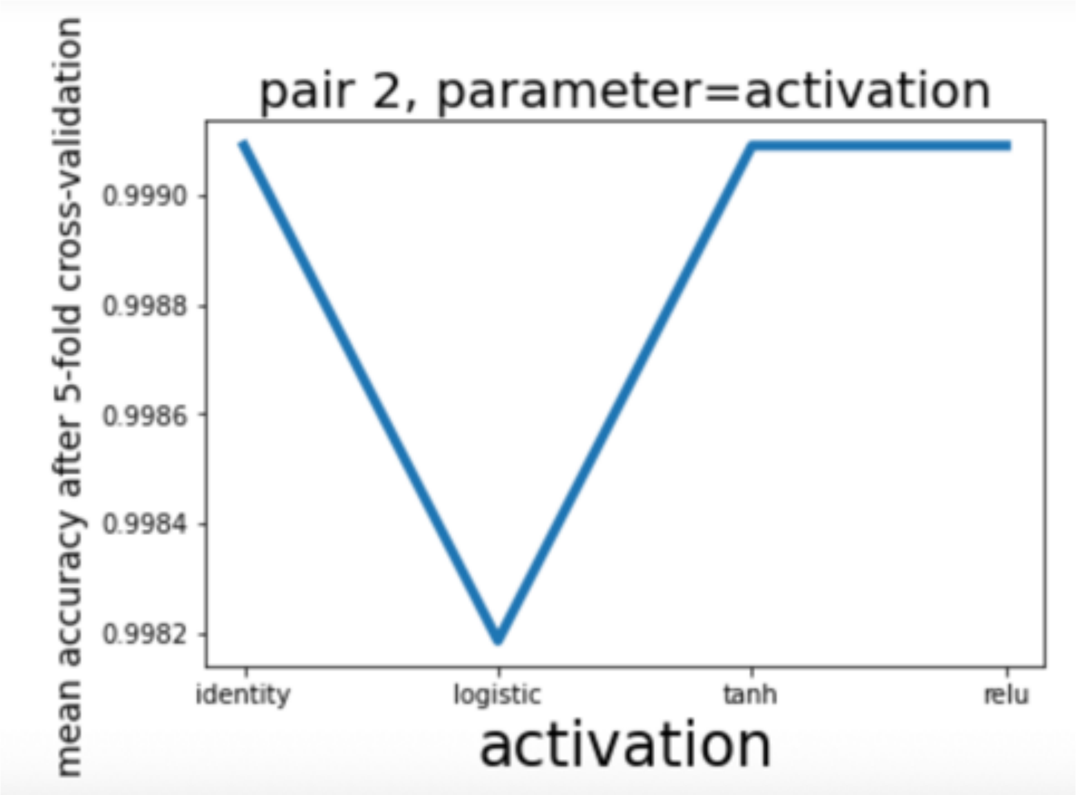

pair 1, parameter=coef0
mean accuracy after 5-fold cross-validation
0.97675
0.97650
0.97625
0.97600
0.97575
0.97550
0.97525
0.97500
0.97475
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
coef0

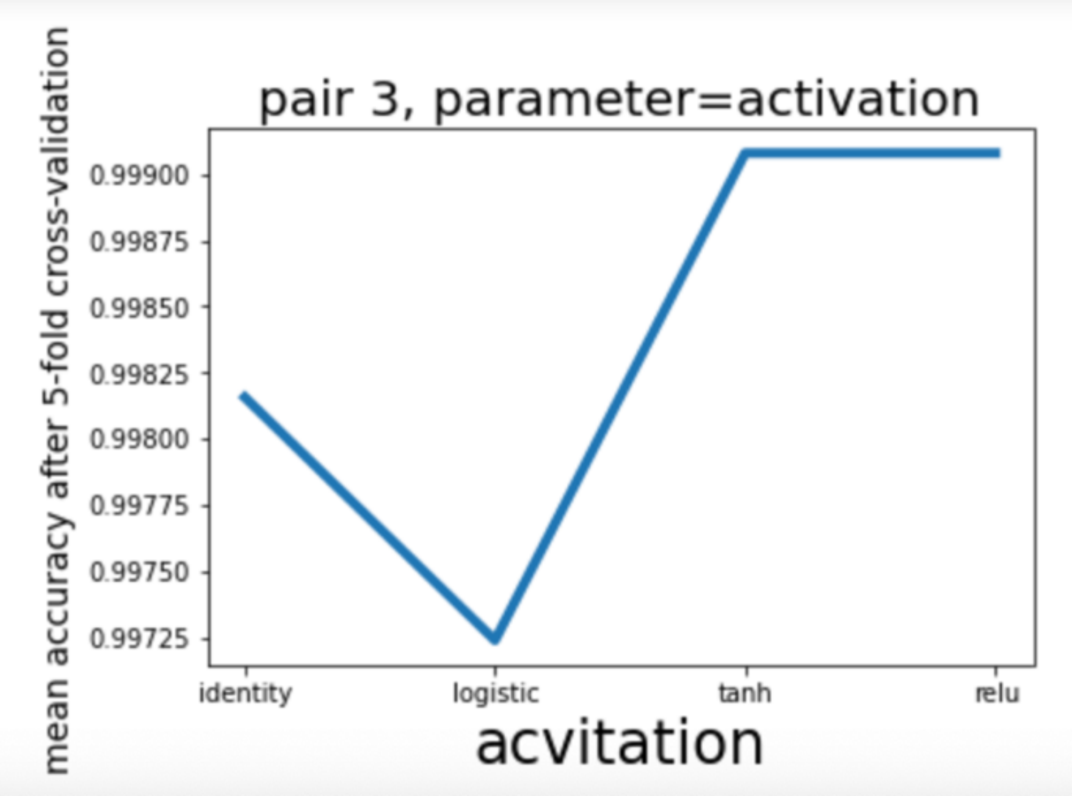

pair 2, parameter=coef0
mean accuracy after 5-fold cross-validation
1.04
1.02
1.00
0.98
0.96
0.94
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
coef0


pair 3, parameter=coef0
mean accuracy after 5-fold cross-validation
0.9982
0.9980
0.9978
0.9976
0.9974
0.9972
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
coef0

ANN


pair 1, parameter=activation
mean accuracy after 5-fold cross-validation
0.97
0.96
0.95
0.94
0.93
identity
logistic
tanh
relu
activation


pair 2, parameter=activation
mean accuracy after 5-fold cross-validation
0.9990
0.9988
0.9986
0.9984
0.9982
identity
logistic
tanh
relu
activation


pair 3, parameter=activation
mean accuracy after 5-fold cross-validation
0.99900
0.99875
0.99850
0.99825
0.99800
0.99775
0.99750
0.99725
identity
logistic
tanh
relu
acvitation

Pair 1

	Accuracy on validation set before dimension reduction
KNN	0.8245977011494252
Decision tree	0.8372413793103449
Random forest	0.8510344827586207
SVM	0.8917241379310346
ANN	0.8917241379310343

Pair 2

	Accuracy on validation set before dimension reduction
KNN	0.9875
Decision tree	0.98125
Random forest	0.98125
SVM	0.98125
ANN	0.9875

Pair 3

	Accuracy on validation set before dimension reduction
KNN	0.955241935483871
Decision tree	0.9358870967741936
Random forest	1.0
SVM	0.955241935483871
ANN	0.9679435483870968

Pair 1

	Accuracy on validation set after dimension reduction
KNN	0.8581609195402299
Decision tree	0.8301149425287356
Random forest	0.871264367816092
SVM	0.864367816091954
ANN	0.83816091954023

Pair 2

	Accuracy on validation set after dimension reduction
KNN	0.99375
Decision tree	0.975
Random forest	0.98125
SVM	0.98125
ANN	0.9872983870967742

Pair 3

	Accuracy on validation set after dimension reduction
KNN	0.9423387096774194
Decision tree	0.955241935483871
Random forest	0.9487903225806452
SVM	0.9233870967741936
ANN	0.9233870967741936

According to accuracy, I will choose random forest as my final model for this problem, because its accuracy is relatively robust among 3 pairs and is also higher than the average level of 5 ML models.

On the one hand, dimension reduction lowers the accuracy to a certain extent, but it is acceptable since those are not a significant loss of accuracy; on the other hand, dimension reduction decrease the running time of all the 5 models which increases efficiency.

If I was given this same task for a new dataset, I will first normalize the data, because in most cases, after normalization, the performance of the models will be better.