Programming Assignment 2 Report

- **Student Name: Hongyi Xu**
- **Student Number: 499173**

1. Introduction

Description/formulation of the problem

For this report, my goal is to implement different classification models and use different dimension reduction functions to solve a binary classification problems. I determine the "best" classifier towards these problems by comparing and analyzing the accuracy and the running time of these different models and functions. From this report. I will have a deeper understanding of classification models and dimension reductions. I can implement them into many similar classification problems in the future. They will be very useful for companies and our daily life things' predictions not only recognizing this binary English alphabet classification problem.

Introduction to the binary classification problems

The objective is to identify each of a large number of black-and-white rectangular pixel displays as one of the 26 capital letters in the English alphabet. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli. Each stimulus was converted into 16 primitive numerical attributes (statistical moments and edge counts) which were then scaled to fit into a range of integer values from 0 through 15. The attribute information is listed below:

1. lettr capital letter (26 values from A to Z)
2. x-box horizontal position of box (integer)
3. y-box vertical position of box (integer)
4. width width of box (integer)
5. high height of box (integer)
6. onpix total # on pixels (integer)
7. x-bar mean x of on pixels in box (integer)
8. y-bar mean y of on pixels in box (integer)
9. x2bar mean x variance (integer)
10. y2bar mean y variance (integer)
11. xybar mean x y correlation (integer)
12. x2ybr mean of x * x * y (integer)
13. xy2br mean of x * y * y (integer)
14. x-ege mean edge count left to right (integer)
15. xegvy correlation of x-ege with y (integer)
16. y-ege mean edge count bottom to top (integer)
17. yegvx correlation of y-ege with x (integer)

In this programming report, we consider three classification problems:

Pair 1: H and K   Pair 2: M and Y   Pair3: B and O

Discussion about dimension reduction

I used 8 different methods to reduce the number of features from 16 to 4. I both tried use them in one model(KNN) and in different models to compare the results. Then I found that the running time reduced after using dimension reduction. But the accuracy scores are always worse than 16 features' work (some methods excepted). One of the reason I guess is we have reduced many features(from 16 to 4) and this way may loss accuracy.

2.Results

For this part, I will give brief description of each classifier(7 in total). I will graph the cross validation results over the range of hyperparameter values (7 classifiers and each has tested 2 hyperparameters, 14 in total). Then I will use different dimension reduction methods and tested the hyperparameter values with cross validation.

Note: Some introduction of Classifiers and dimension reduction methods are referred by wiki or other internet resources.

Classifier

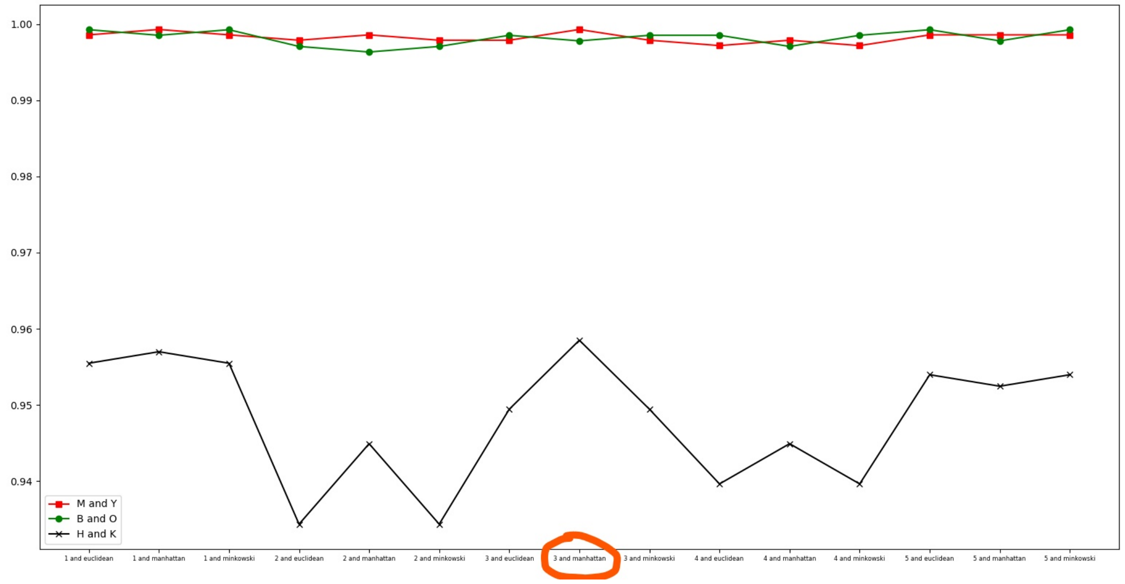
1. k-nearest neighbors

KNN works by finding the distances between a query and all the examples in the data, selecting the specified number examples (K) closest to the query, then votes for the most frequent label.

The advantage of KNN: Easy to explain, requires no training time.

The disadvantage of KNN: 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.

I tuned two hyperparameters, n_neighbors and metric. For the n_neighbors, I tried 1-5. For the metric, I tried "Euclidean", "Manhattan" and "Minkowski". The accuracy of those three binary problems are shown below:



We can see that when the hyperparameters are 3 and "Manhattan", the accuracy is relatively better.

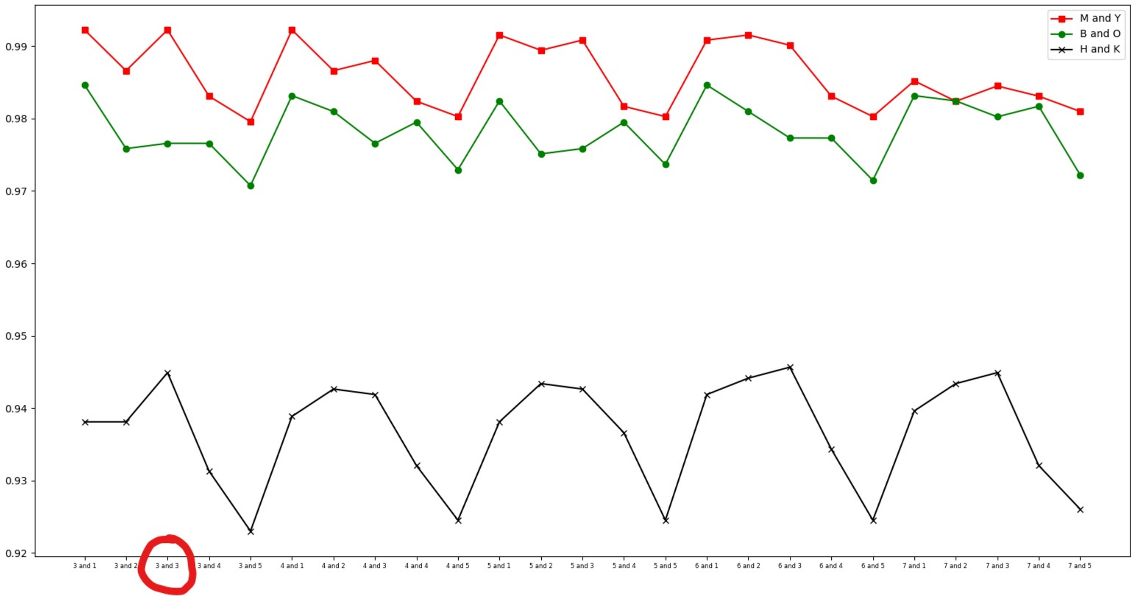
2. Decision tree

A decision tree is a tree-like model that acts as a decision support tool, visually displaying decisions and their potential outcomes, consequences, and costs. From there, the “branches” can easily be evaluated and compared in order to select the best courses of action.

The advantage of decision tree: Easy to apply - each chosen split can be understood and checked by a human user during the prediction process.

The disadvantage of decision tree: Easy to overfit by splitting over and over again

I tuned two hyperparameters, min_samples_split and min_samples_leaf. I tried 1-5 for both of them. The accuracy of those three binary problems are shown below:



We can see that when the hyperparameters are both 3, the accuracy is relatively better.

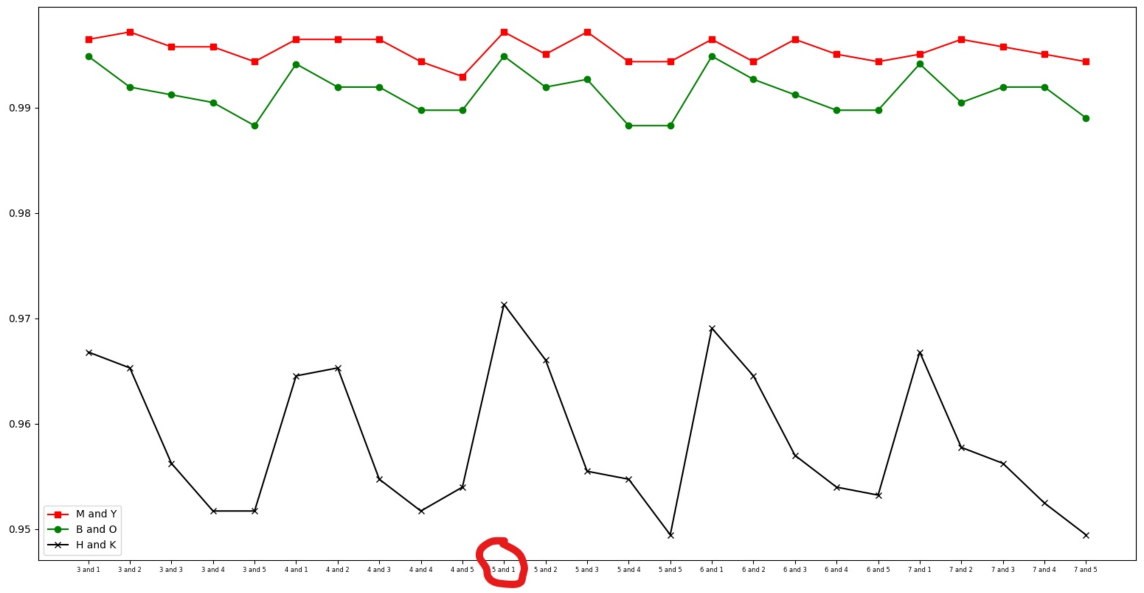
3. Random Forest

The random forest is a classification algorithm consisting of many decisions trees. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.

The advantage of random forest: Can have high accuracy while avoiding overfitting like a single decision tree.

The disadvantage of random forest: Takes longer to train than a single decision tree

I tuned two hyperparameters, min_samples_split and min_samples_leaf. I tried 1-5 for both of them. The accuracy of those three binary problems are shown below:



We can see that when the hyperparameters are 5 and 1, the accuracy is relatively better.

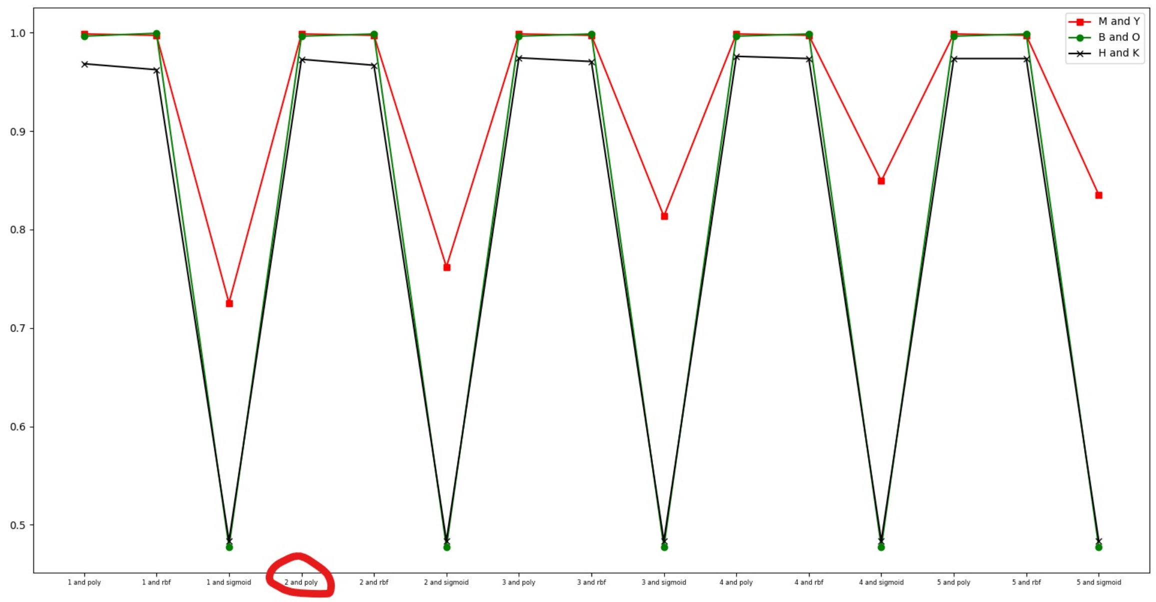
4. SVM

SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. A separator between the categories is found, then the data are transformed in such a way that the separator could be drawn as a hyperplane.

The advantage of SVM: SVM works relatively well when there is a clear margin of separation between classes. SVM is more effective in high dimensional spaces. SVM is effective in cases where the number of dimensions is greater than the number of samples. SVM is relatively memory efficient

The disadvantage of SVM: SVM algorithm is not suitable for large data sets. SVM does not perform very well when the data set has more noise, for example, target classes are overlapping. In cases where the number of features for each data point exceeds the number of training data samples, the SVM will underperform.

I tuned two hyperparameters, C value and kernel. For the C value, I tried 1-5. For the kernel, I tried "Polynomial", "RBF" and "Sigmoid". The accuracy of those three binary problems are shown below:



We can see that when the hyperparameters are 2 and "Polynomial", the accuracy is relatively better.

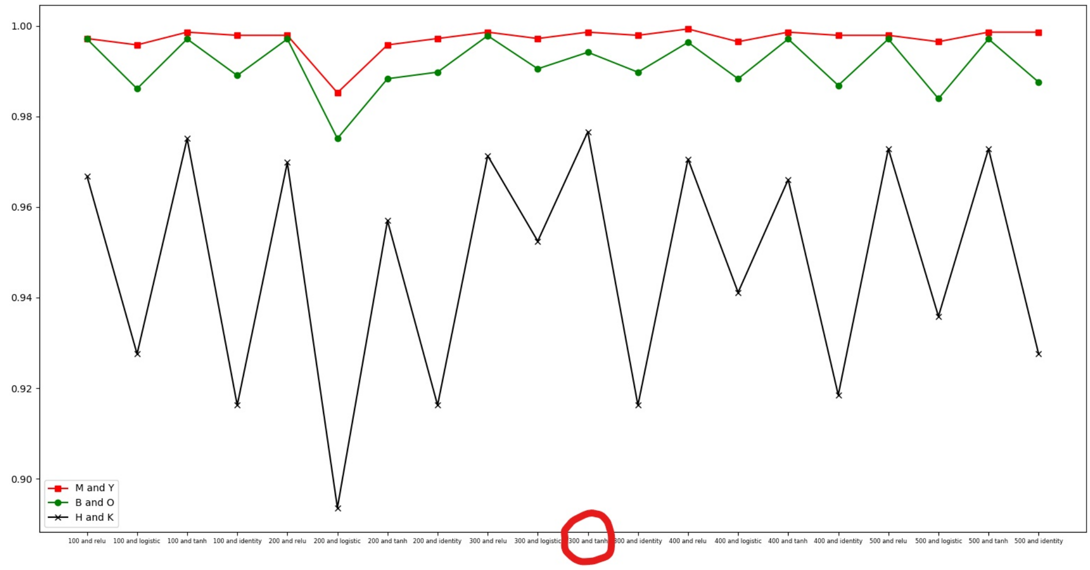
5.Artificial Neural Network

Artificial neural network (ANN) is a computational model that consists of several processing elements that receive inputs and deliver outputs based on their predefined activation functions.

The advantage of ANN: Can fit just about any model with high accuracy.

The disadvantage of ANN: Is difficult to interpret by a client with no data science or machine learning background. Have to consume more running time.

I tuned two hyperparameters, max iteration and activation. For the max iteration, I tried 100-500. For the kernel, I tried "Relu", "Logistic", "Tanh" and "Identity". The accuracy of those three binary problems are shown below:



We can see that when the hyperparameters are 300 and "Relu", the accuracy is relatively better.

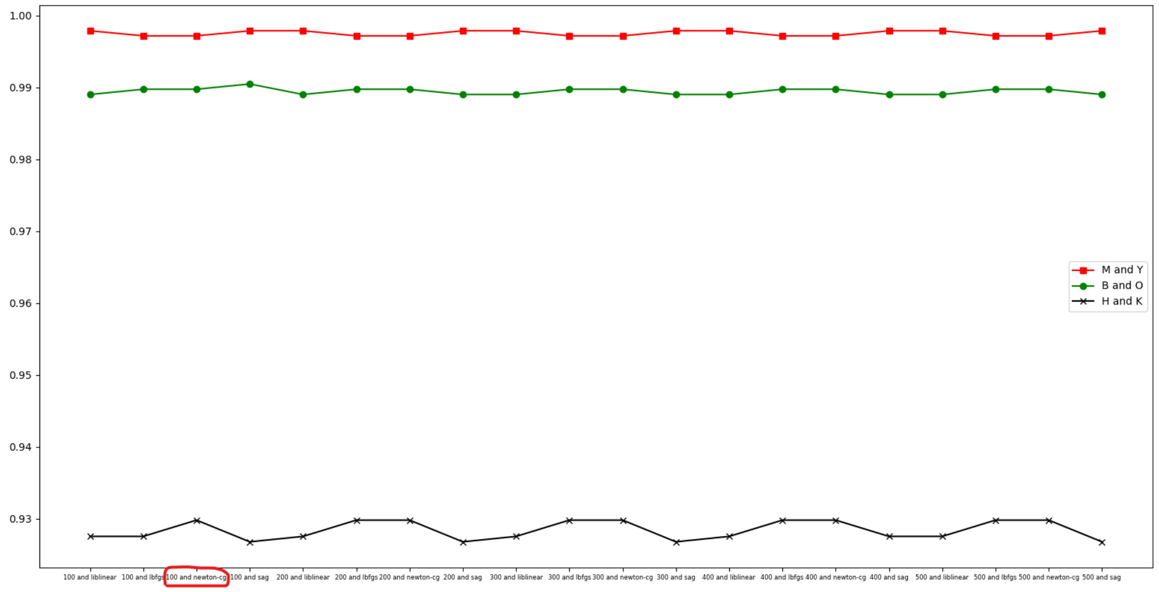
6. Logistic Regression

Logistic regression is a statistical analysis method to predict a binary outcome, such as yes or no, based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

The advantage of Logistic regression: Logistic regression is easier to implement, interpret, and very efficient to train. It makes no assumptions about distributions of classes in feature space.

The disadvantage of Logistic regression: If the number of observations is lesser than the number of features, Logistic Regression should not be used, otherwise, it may lead to overfitting.

I tuned two hyperparameters, max iteration and solver. For the max iteration, I tried 100-500. For the solver, I tried "Liblinear", "Lbfgs", "Newton-cg" and "Sag". The accuracy of those three binary problems are shown below:



We can see that when the hyperparameters are 100 and "Newton-cg", the accuracy is relatively better.

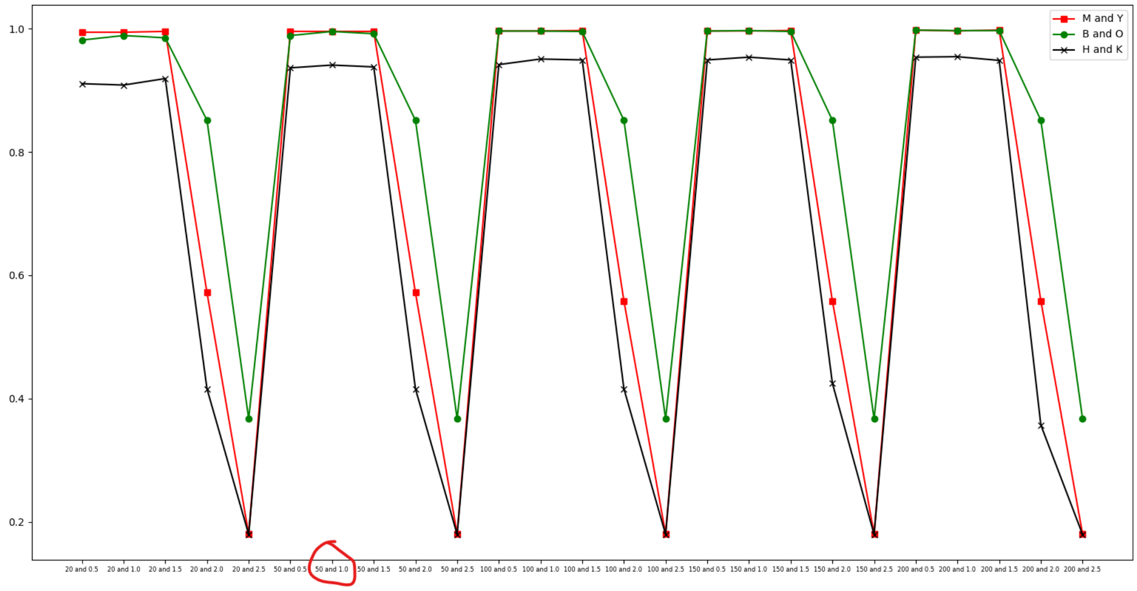
7. AdaBoost

AdaBoost is an ensemble learning method (also known as “meta-learning”) which was initially created to increase the efficiency of binary classifiers. AdaBoost uses an iterative approach to learn from the mistakes of weak classifiers, and turn them into strong ones.

The advantage of AdaBoost: AdaBoost is less prone to overfitting as the input parameters are not jointly optimized. The accuracy of weak classifiers can be improved by using AdaBoost.

The advantage of AdaBoost: it needs a quality dataset. Noisy data and outliers have to be avoided before adopting an AdaBoost algorithm.

I tuned two hyperparameters, n_estimators and learning rate. For the n_estimators, I tried 20, 50, 100, 150 and 200. For the learning rate, I tried 0.5, 1.0, 1.5, 2.0 and 2.5. The accuracy of those three binary problems are shown below:

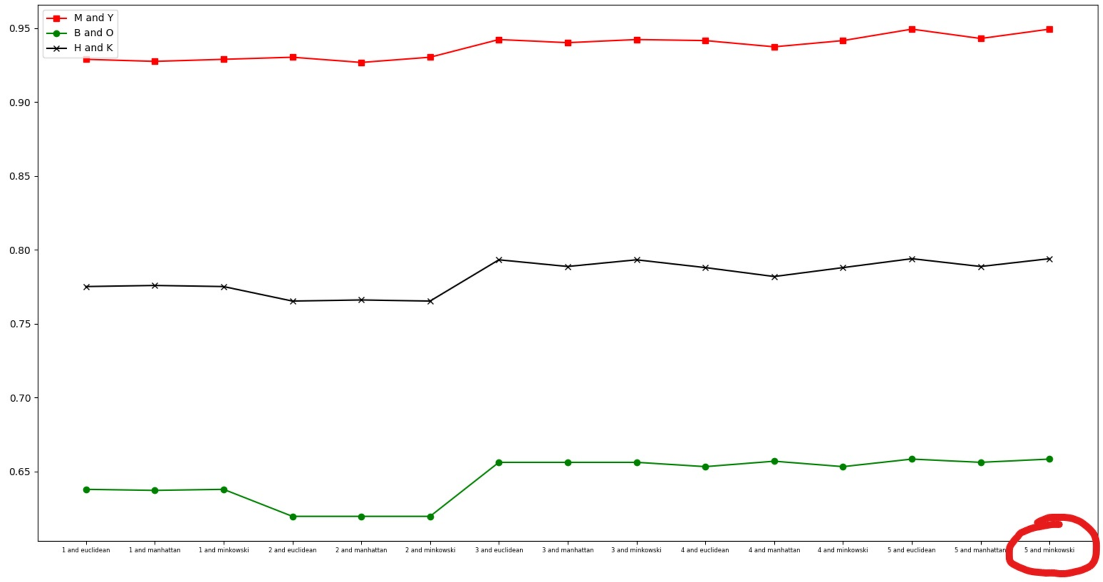


We can see that when the hyperparameters are 50 and 1.0, the accuracy is relatively better.

Dimension reduction

1. Low Variance

I used Low Variance method on KNN model. Eliminate the low variance dimension in the data, from 16 to 4. I tuned the same two hyperparameters as KNN without dimension reduction, n_neighbors and metric. The accuracy of those three binary problems are shown below:

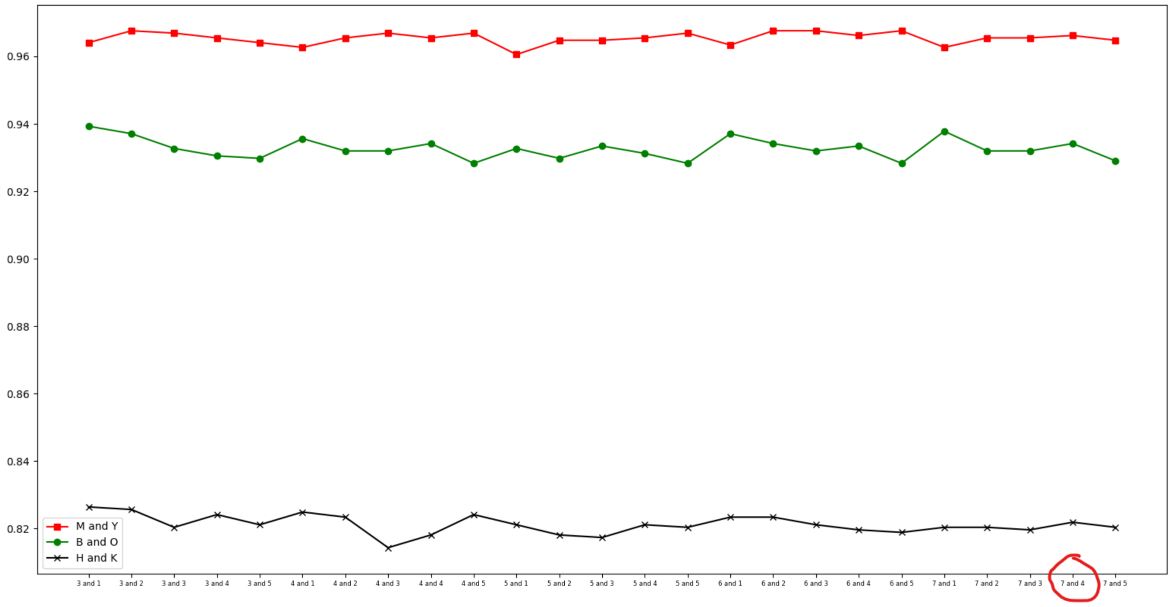


We can see that different from the model without dimension reduction, when the hyperparameters are 5 and "Minkowski", the accuracy is relatively better.

2. Factor Analysis

Factor analysis is a powerful data reduction technique that enables researchers to investigate concepts that cannot easily be measured directly. By boiling down a large number of variables into a handful of comprehensible underlying factors, factor analysis results in easy-to-understand, actionable data.

I used Factor Analysis on Decision Tree model. I tuned two hyperparameters, min_samples_split and min_samples_leaf. The accuracy of those three binary problems are shown below:

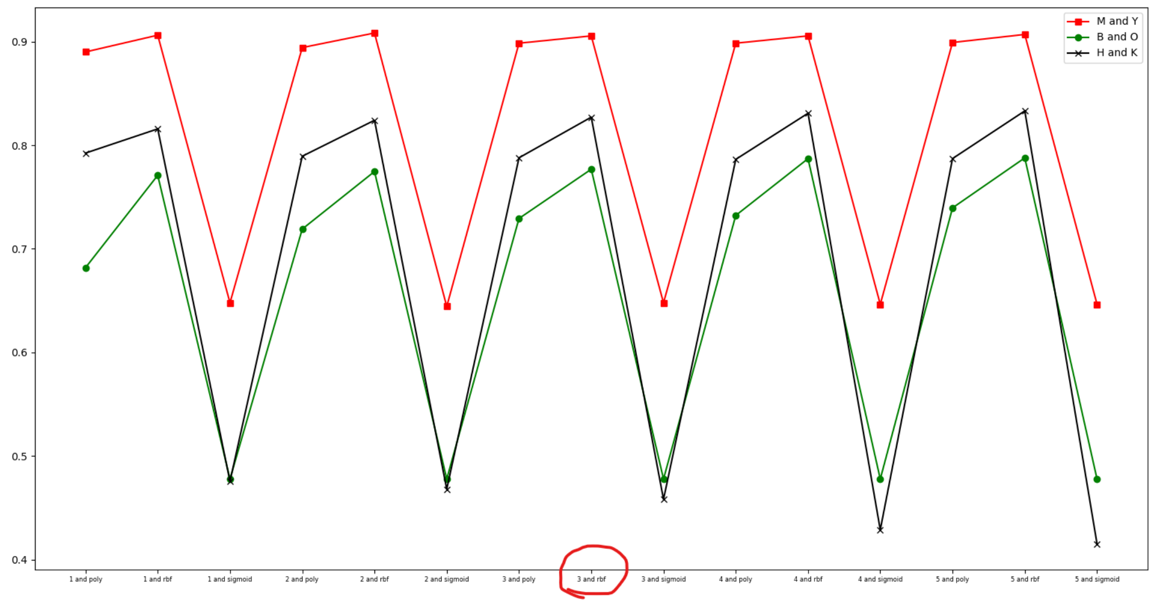


We can see that different from the model without dimension reduction, when the hyperparameters are 7 and 4, the accuracy is relatively better.

3. High Correlation

Correlation is a term that refers to the strength of a relationship between two variables where a strong, or high, correlation means that two or more variables have a strong relationship with each other while a weak or low correlation means that the variables are hardly related.

I used High Correlation on SVM model. I tuned two hyperparameters, C value and kernel. The accuracy of those three binary problems are shown below:

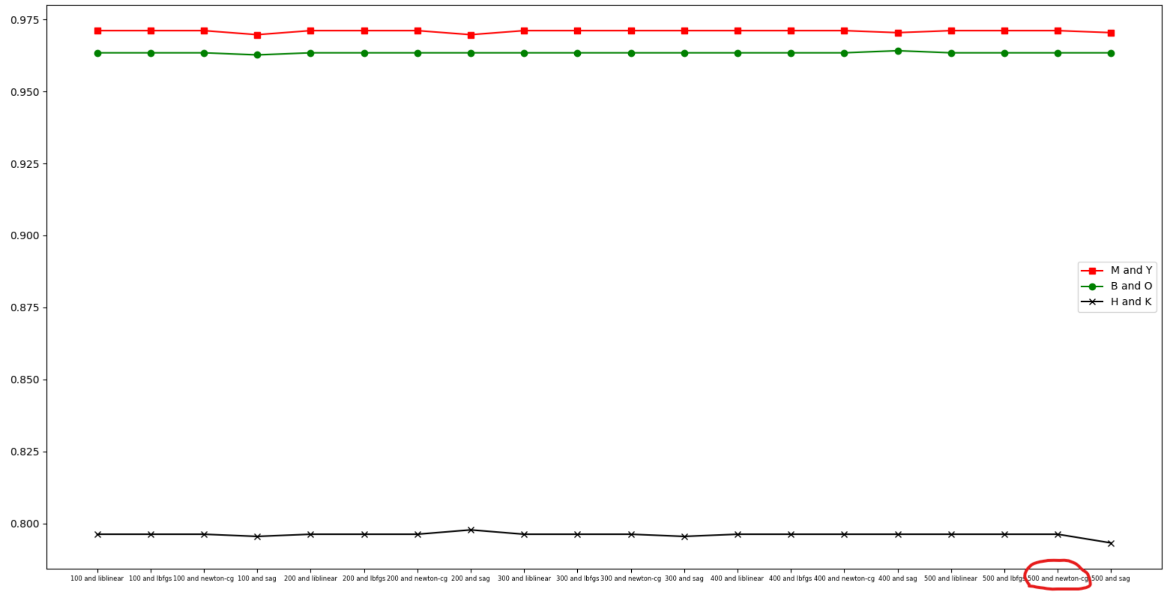


We can see that different from the model without dimension reduction, when the hyperparameters are 3 and "RBF", the accuracy is relatively better.

4. ICA

Independent Component Analysis (ICA) is a machine learning technique to separate independent sources from a mixed signal. Unlike principal component analysis which focuses on maximizing the variance of the data points, the independent component analysis focuses on independence.

I used ICA on Logistic Regression model. I tuned two hyperparameters, max iteration and solver. The accuracy of those three binary problems are shown below:

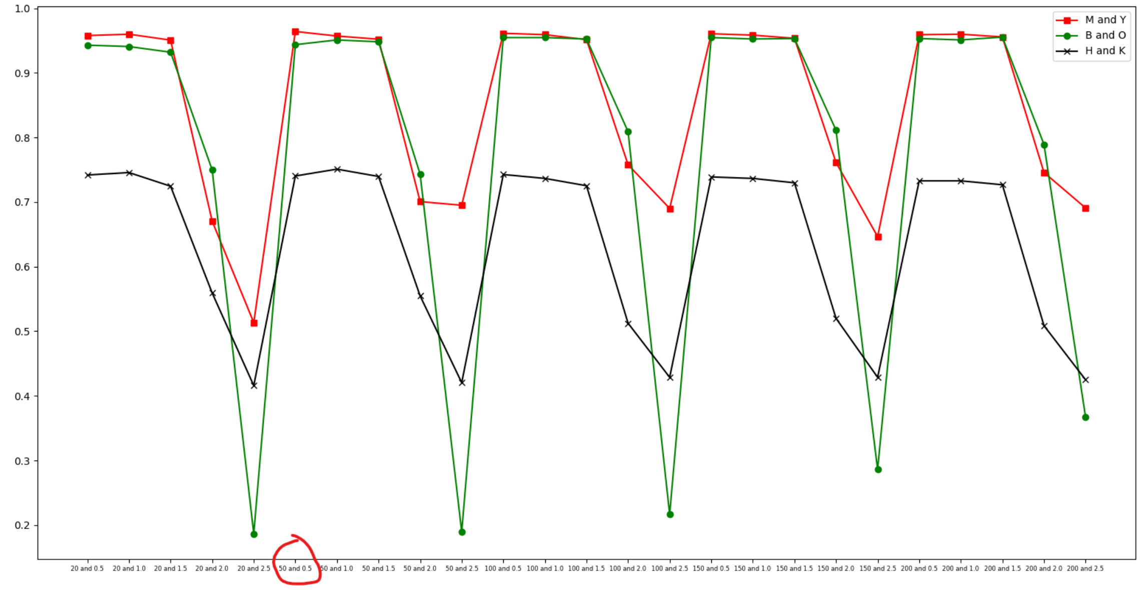


We can see that different from the model without dimension reduction, when the hyperparameters are 500 and "Newton-cg", the accuracy is relatively better.

5. SVD

In linear algebra, the Singular Value Decomposition (SVD) of a matrix is a factorization of that matrix into three matrices. It has some interesting algebraic properties and conveys important geometrical and theoretical insights about linear transformations. It also has some important applications in data science.

I used SVD on AdaBoost model. I tuned two hyperparameters, n_estimators and learning_rate. The accuracy of those three binary problems are shown below:

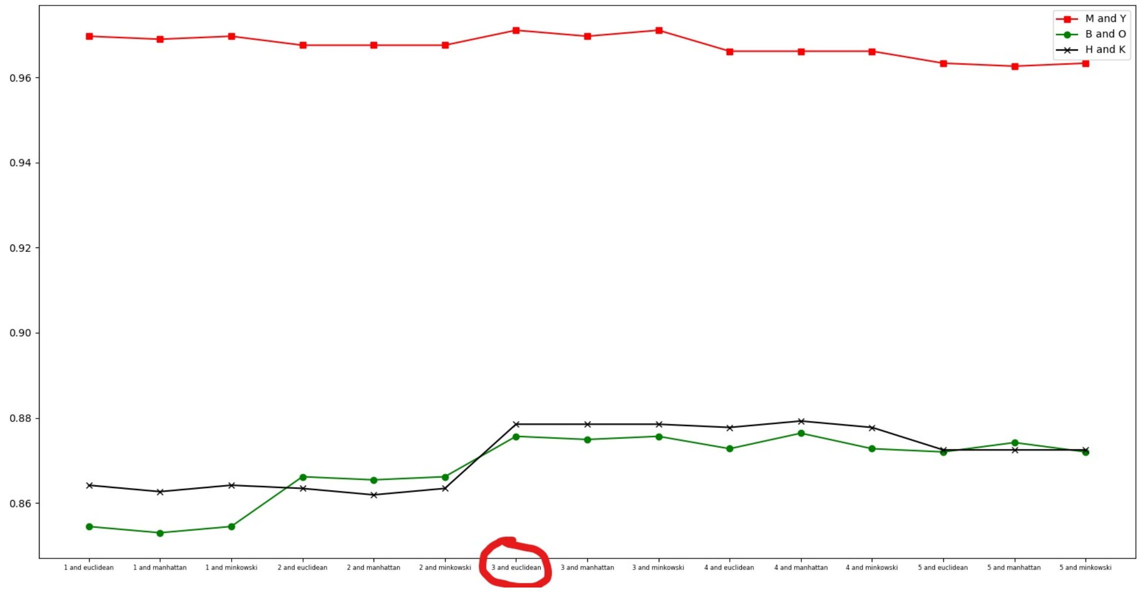


We can see that different from the model without dimension reduction, when the hyperparameters are 50 and 0.5, the accuracy is relatively better.

6. Random Forest Reduction

Random forests is a tree-based model which is widely used for regression and classification tasks on non-linear data. It can also be used for feature selection with its built-in feature_importances_ attribute which calculates feature importance scores for each feature based on the 'gini' criterion (a measure of the quality of a split of internal nodes) while training the model.

I used Random Forest Reduction on KNN model. I tuned two hyperparameters, n_neighbors and metric. The accuracy of those three binary problems are shown below:

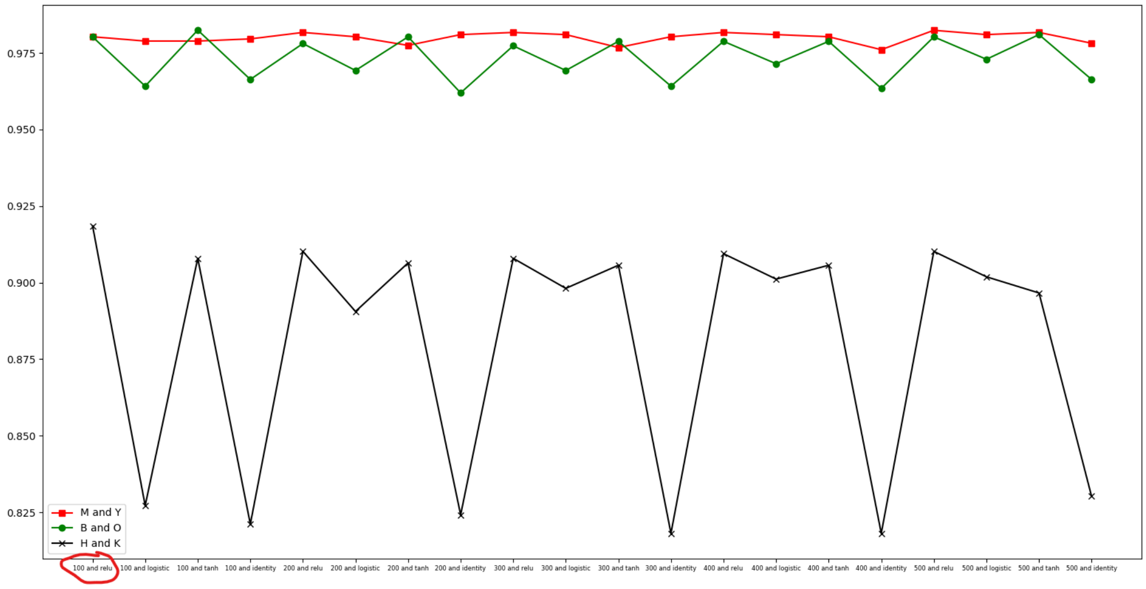


We can see that different from the model without dimension reduction, when the hyperparameters are 3 and "Euclidean", the accuracy is relatively better.

7. PCA

Principal component analysis (PCA) is the process of computing the principal components and using them to perform a change of basis on the data, sometimes using only the first few principal components and ignoring the rest.

I used PCA on ANN model. I tuned two hyperparameters, max iteration and activation. The accuracy of those three binary problems are shown below:

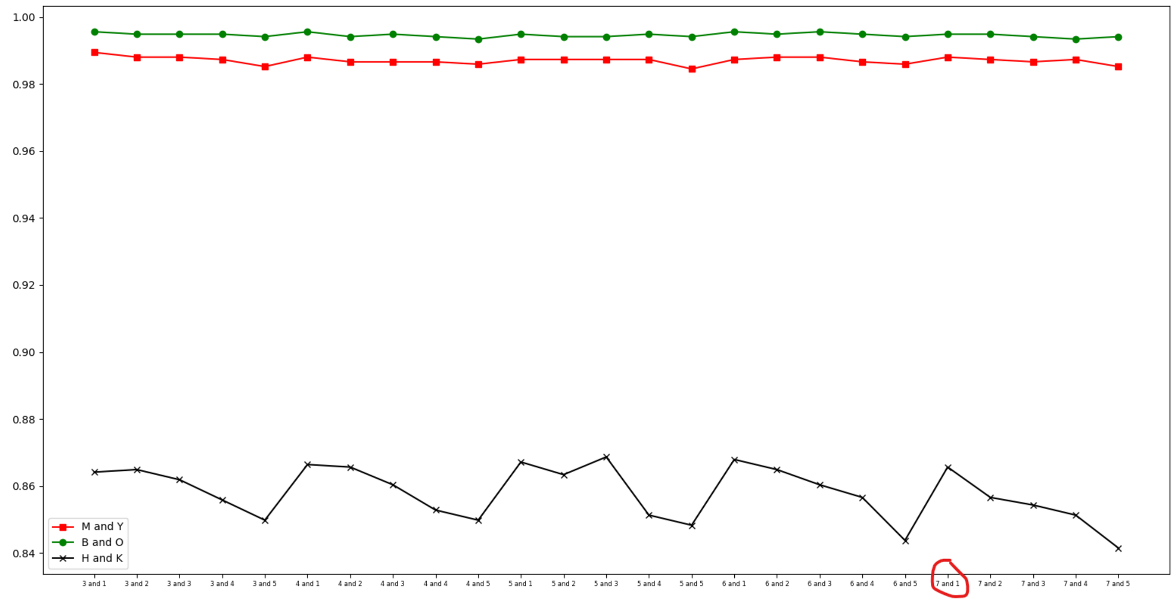


We can see that different from the model without dimension reduction, when the hyperparameters are 100 and "Relu", the accuracy is relatively better.

8. ISOMAP

Isomap is a non-linear dimensionality reduction method based on the spectral theory which tries to preserve the geodesic distances in the lower dimension. Isomap starts by creating a neighborhood network. After that, it uses graph distance to the approximate geodesic distance between all pairs of points.

I used Isomap on Random Forest model. I tuned two hyperparameters, min_samples_split and min_samples_leaf. The accuracy of those three binary problems are shown below:



We can see that different from the model without dimension reduction, when the hyperparameters are 7 and 1, the accuracy is relatively better.

Discussion

The performance and run time of different classifiers without dimension reduction

The accuracy of these three classification problems and run time of those different classifiers without dimension reduction is shown blow:

	H and K accuracy	M and Y accuracy	B and O accuracy	run time
KNN	0.939189	1.000000	1.000000	0.022490500000000019
Decision tree	0.891892	1.000000	1.000000	0.011859000000000006
Random Forest	0.966216	1.000000	1.000000	0.3296030999999999
SVM	0.986486	1.000000	1.000000	0.036032300000000002
Artificial Neural Network	0.993243	1.000000	1.000000	5.9047111999999995
Logistic Regression	0.939189	1.000000	0.986842	0.10684510000000014
AdaBoost	0.932432	1.000000	1.000000	0.25759430000000005

Note: The running time means how long it takes the model to train and predict.

The performance and run time of different classifiers with dimension reduction

The accuracy of these three classification problems and run time of those different classifiers with dimension reduction is shown blow:

	H and K accuracy	M and Y accuracy	B and O accuracy	run time
KNN (Low Variance)	0.790541	0.955696	0.638158	0.01784009999999991
Decision tree (Factor Analysis)	0.831081	0.981013	0.927632	0.01316689999999987
Random Forest (Isomap)	0.831081	0.993671	0.993421	0.39926739999999983
SVM (High Correlation)	0.858108	0.936709	0.769737	0.09900829999999994
Artificial Neural Network (PCA)	0.898649	0.993671	1.000000	6.3602929
Logistic Regression (ICA)	0.831081	0.993671	0.973684	0.01337090000000018
AdaBoost (SVD)	0.750000	0.917722	0.980263	0.2619691000000002

Note: The running time means how long it takes the model to train and predict.

Lessons learned

From these tables, we can see that before the dimension reduction, most of the classifiers got a good performance in all three classification problems. Artificial Neural Network did a best performance among them. So for this problem I would choose Artificial Neural Network model.

After implementing different dimension reduction methods, the accuracy became a little worse for most of the models and problems. Running times were increased tinny number as well. But all of these loss of accuracy and run time were in acceptable range. They still did a great job on those problems.

If I was given this same task for a new dataset, I would choose use PCA for dimension reduction and use ANN as classification model. Using PCA and ANN seems to be able to get a great accuracy score although the running time may be a little higher.

I learned a lot from this project. I got a deeper understanding of many classification models and different dimension reduction methods. I learned that if I want to do some more classification problems in the future, I should choose the most suitable dimension reduction methods and classification models since they all have different advantages and disadvantages.