# COMP 5212: MACHINE LEARNING PROJECT 1

Young Dae KWON (Student ID: 20236796)

12 March, 2018

## 1 INTRODUCTION

In this project, I explore the performances of various machine learning algorithms in different datasets. I use four machine learning algorithms such as Logistic Regression (LR), Linear Support Vector Machine (Linear SVM), Radial Base Function Support Vector Machine (RBF SVM), and Neural Networks (NNs). I investigate parameter settings of algorithms in each dataset with 5-fold Cross-Validation and present best parameter which shows best classification performance. Furthermore, I analyze the results of classifier with Area Under Curve (AUC), Confusion Matrix, and computation time during training and testing period.

Table 1: Descriptive statistics about Training datasets used in empirical study

| Dataset | # features | # Train | # Class 0 | # Class 1 | Mean (all features) | Std (all features) |
|---|---|---|---|---|---|---|
| Breast | 10 | 547 | 191 | 356 | -0.6 | 0.6 |
| Diabetes | 8 | 615 | 214 | 401 | -0.4 | 0.5 |
| Digits | 64 | 800 | 412 | 388 | 4.9 | 6.0 |
| Iris | 4 | 120 | 40 | 80 | 3.5 | 2.0 |
| Wine | 13 | 142 | 85 | 57 | 69.9 | 218.7 |

Table 2: Descriptive statistics about Testing datasets used in empirical study

| Dataset | # features | # Test | # Class 0 | # Class 1 | Mean (all features) | Std (all features) |
|---|---|---|---|---|---|---|
| Breast | 10 | 136 | 48 | 88 | -0.6 | 0.6 |
| Diabetes | 8 | 153 | 54 | 99 | -0.4 | 0.5 |
| Digits | 64 | 200 | 91 | 109 | 4.9 | 6.0 |
| Iris | 4 | 30 | 10 | 20 | 3.4 | 2.0 |
| Wine | 13 | 36 | 22 | 14 | 66.1 | 203.8 |

Five different datasets are used in this project such as Breast Cancer, Diabetes, Digit, Iris, and Wine. Table 1 and Table 2 show descriptive statistics of datasets for empirical studies. As in Table 1 and Table 2, mean and standard deviation of features in each dataset vary. Thus, I conduct standardization for all features before training classifiers.

I explain detailed process of empirical study on four machine learning algorithms and five different datasets which are used in our study. I elaborate which parameters are fixed and which parameters are changed and compare them according to different metrics such as Accuracy, F1 measure, AUC in Section 2, Section 3, Section 4, Section 5, respectively.

# 2 EMPIRICAL STUDY ON LOGISTIC REGRESSION

In this Section, I build the logistic regression model with SGDClassifier function in scikit-learn package. Investigations on logistic regression model are summarized as follows.

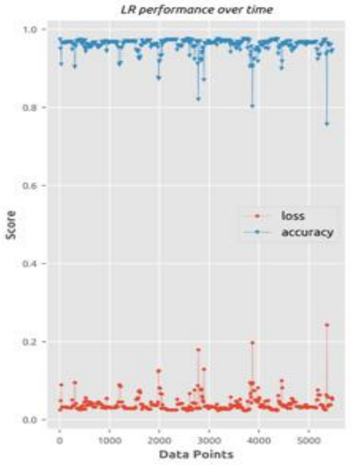
- Present the experiment settings of the logistic regression model
- Show the change in performance of the logistic regression model over time
- Report the accuracy of the logistic regression model on the training and test set
- Report the negative log loss and zero one loss of the logistic regression model on the training and test set
- Report the AUC / Precision / Recall / F1-score and confusion matrix
- Report the computation time for training

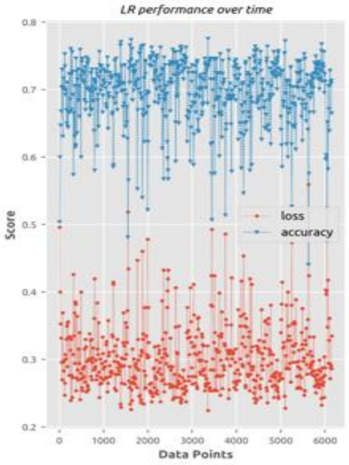
**Present the experiment settings of the logistic regression model**

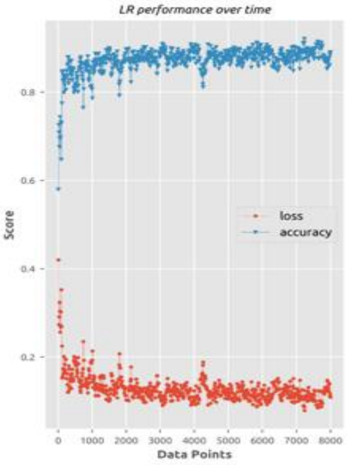
With 5-fold cross-validation method, I train many logistic regression models with different parameters. By choosing the model that shows the best accuracy score, we identify the appropriate model for a classification task. Following parameters are used for training the model. First, 3 different penalties are used such as L1-regularization, L2-regularization, and no penalty. Second, I vary the value of learning rate by factors of ten from $10^{-7}$ to $10^{5}$. As a result of 5-fold Cross-validation method, the parameters which enable the model to show the best accuracy score for each dataset are as follows:

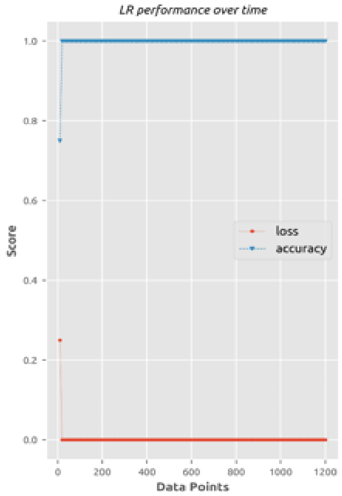
1) Breast-cancer : Penalty is **None** and Learning rate value is **$10^{-6}$**
2) Diabetes : Penalty is **L2-Regularization** and Learning rate value is **$10^{-1}$**
3) Digits : Penalty is **L2-Regularization** and Learning rate value is **$10^{-2}$**
4) Iris : Penalty is **L1-Regularization** and Learning rate value is **$10^{-7}$**
5) Wine : Penalty is **L1-Regularization** and Learning rate value is **$10^{-7}$**

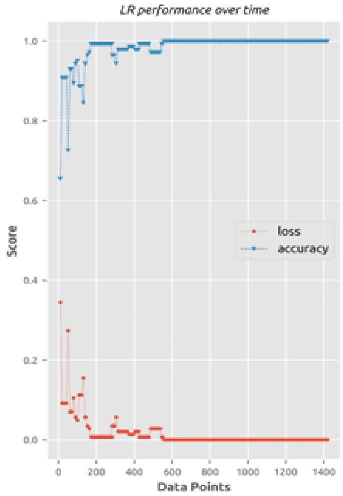
## Show the change in performance of the logistic regression model over time


Breast-cancer


Diabetes

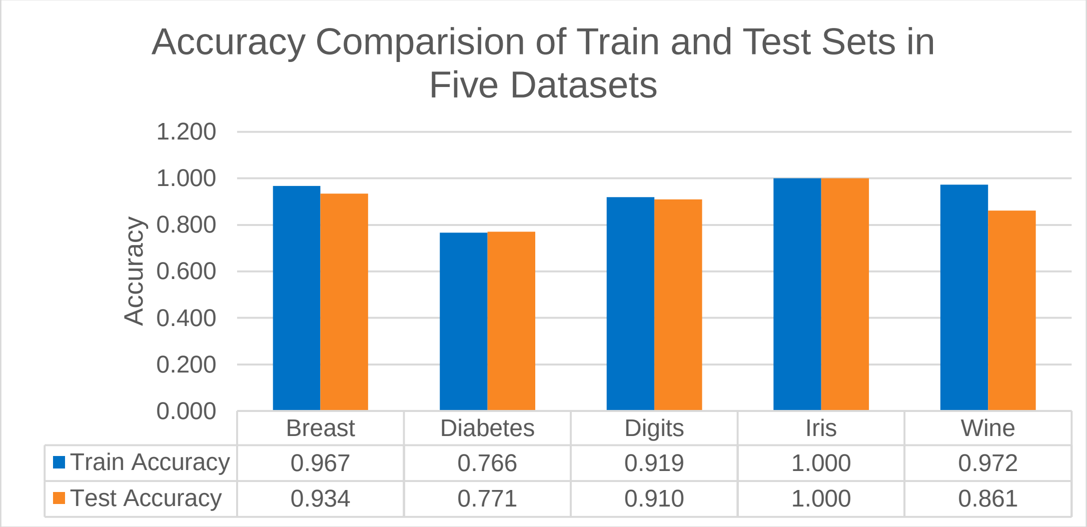

Digits


Iris


Wine

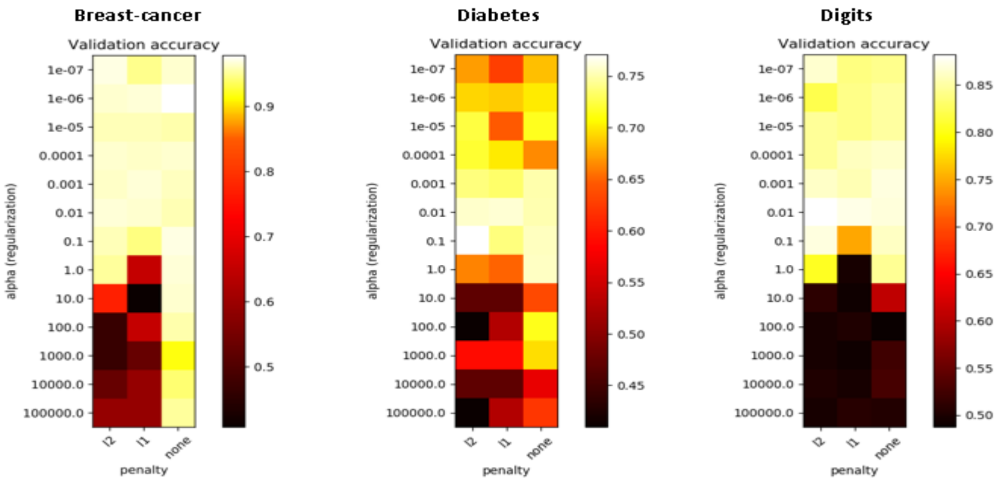
**Report the accuracy of the logistic regression model on the training and test set**

Accuracy results on five datasets are shown below.



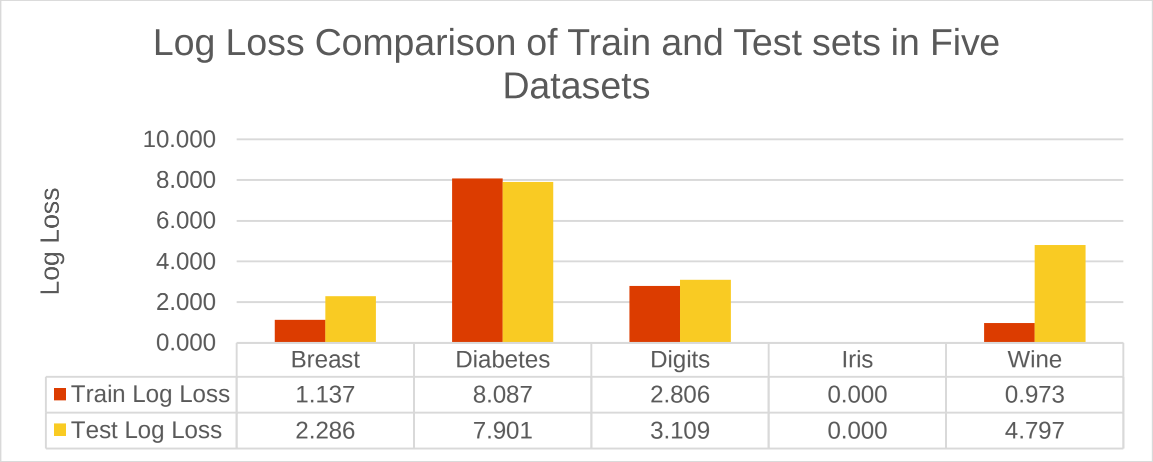
| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟦 Train Accuracy | 0.967 | 0.766 | 0.919 | 1.000 | 0.972 |
| 🟧 Test Accuracy | 0.934 | 0.771 | 0.910 | 1.000 | 0.861 |

Here I present 3 hitmaps which show accuracies of models with different parameters as an example.

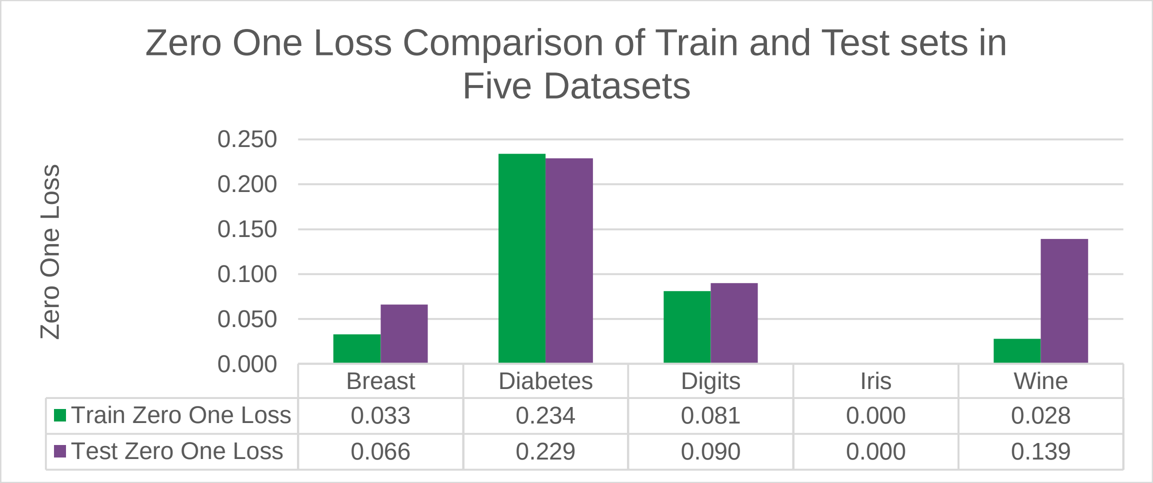


**Report the negative log loss and zero one loss of the logistic regression model on the training and test set**

Negative log loss and zero one loss results on five datasets are shown below.


Log Loss Comparison of Train and Test sets in Five Datasets

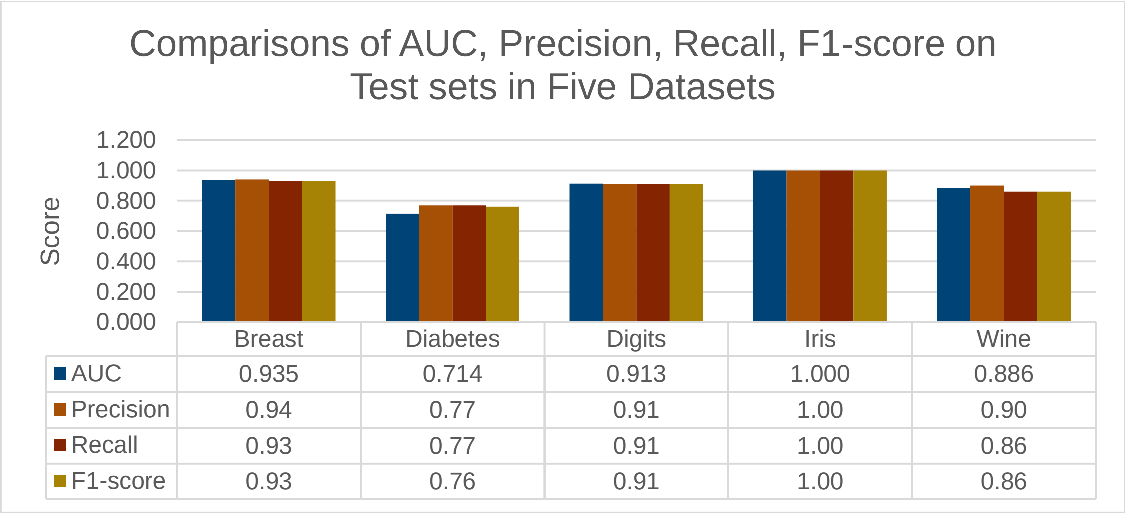
| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟥 Train Log Loss | 1.137 | 8.087 | 2.806 | 0.000 | 0.973 |
| 🟨 Test Log Loss | 2.286 | 7.901 | 3.109 | 0.000 | 4.797 |


Zero One Loss Comparison of Train and Test sets in Five Datasets

| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟩 Train Zero One Loss | 0.033 | 0.234 | 0.081 | 0.000 | 0.028 |
| 🟪 Test Zero One Loss | 0.066 | 0.229 | 0.090 | 0.000 | 0.139 |

**Report the AUC / Precision / Recall / F1-score and confusion matrix**

Results on AUC, Precision, Recall, F1-score, and confusion matrix are presented.

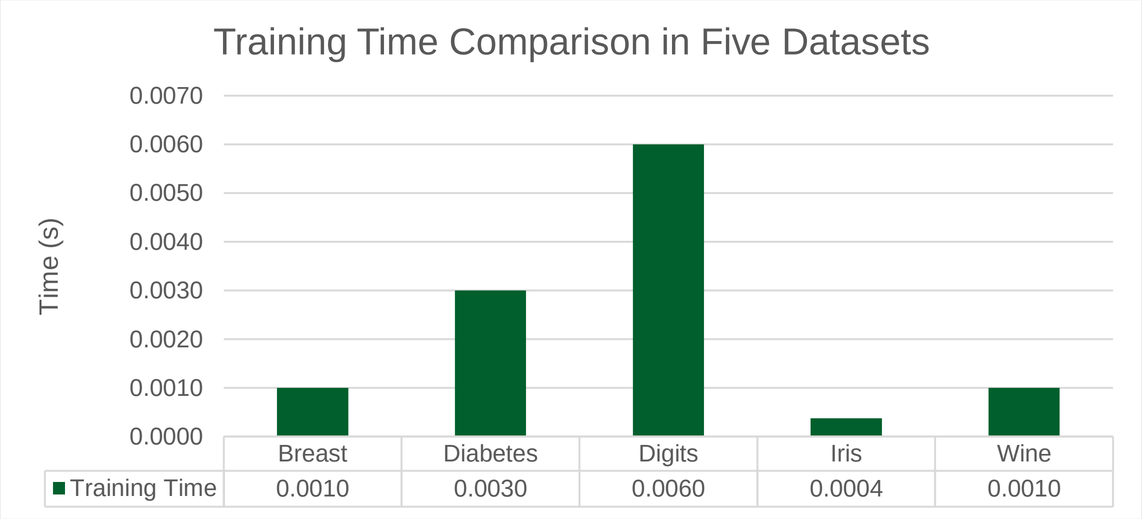

Comparisons of AUC, Precision, Recall, F1-score on Test sets in Five Datasets

| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟦 AUC | 0.935 | 0.714 | 0.913 | 1.000 | 0.886 |
| 🟧 Precision | 0.94 | 0.77 | 0.91 | 1.00 | 0.90 |
| 🟫 Recall | 0.93 | 0.77 | 0.91 | 1.00 | 0.86 |
| 🟨 F1-score | 0.93 | 0.76 | 0.91 | 1.00 | 0.86 |

**Report the computation time for training**

Execution time is measured on a machine with the following characteristics.

- Processor: Intel® Core™ i5-7600 CPU @ 3.50GHz
- CPU(s) : 4
- Architecture : x86_64
- Memory(RAM) : 8.0 GB
- Python virtual environment : python 3.5.4 :: Anaconda



| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟩 Training Time | 0.0010 | 0.0030 | 0.0060 | 0.0004 | 0.0010 |

## 3 EMPIRICAL STUDY ON LINEAR SVM

In this Section, I build the Linear SVM model with SVC function in scikit-learn package. Investigations on Linear SVM model are summarized as follows.

- Present the experiment settings of the Linear SVM model
- Report the accuracy of the Linear SVM model on the training and test set
- Report the zero one loss of the Linear SVM model on the training and test set
- Report the AUC / Precision / Recall / F1-score and confusion matrix
- Report the computation time for training

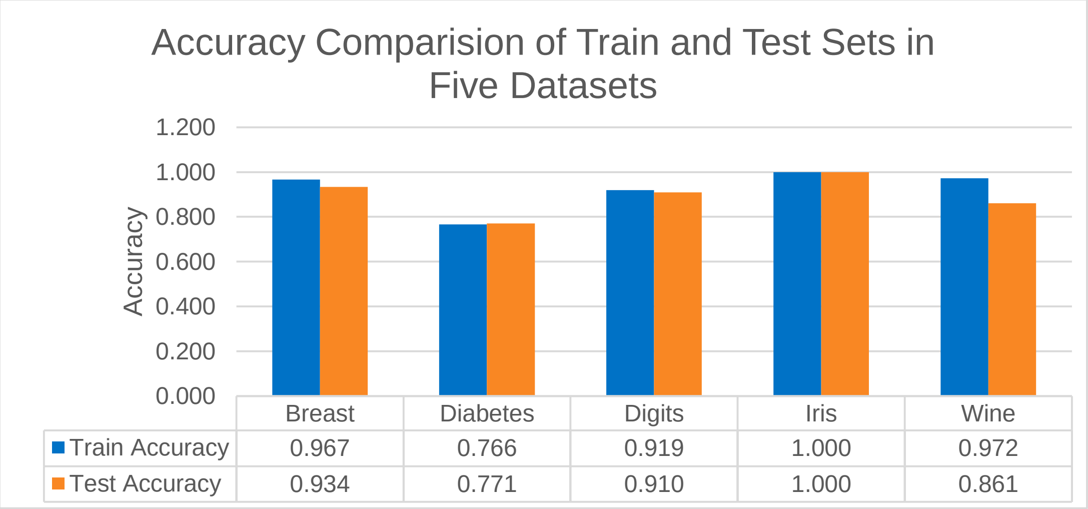
### Present the experiment settings of the Linear SVM model

With 5-fold cross-validation method, I train many Linear SVM models with different parameters. By choosing the model that shows the best accuracy score, an appropriate model for a classification task can be identified. Following parameters are used for training the model. I vary the value of C values by factors of ten from $10^{-7}$ to $10^{5}$. As a result of 5-fold Cross-validation method, the parameters which enable the model to show the best accuracy score for each dataset are as follows:

1) Breast-cancer : C value is **$10^{-1}$**
2) Diabetes : C value is **$10^{-2}$**
3) Digits : C value is **$10^{-1}$**
4) Iris : C value is **$10^{-2}$**
5) Wine : C value is **1.0**

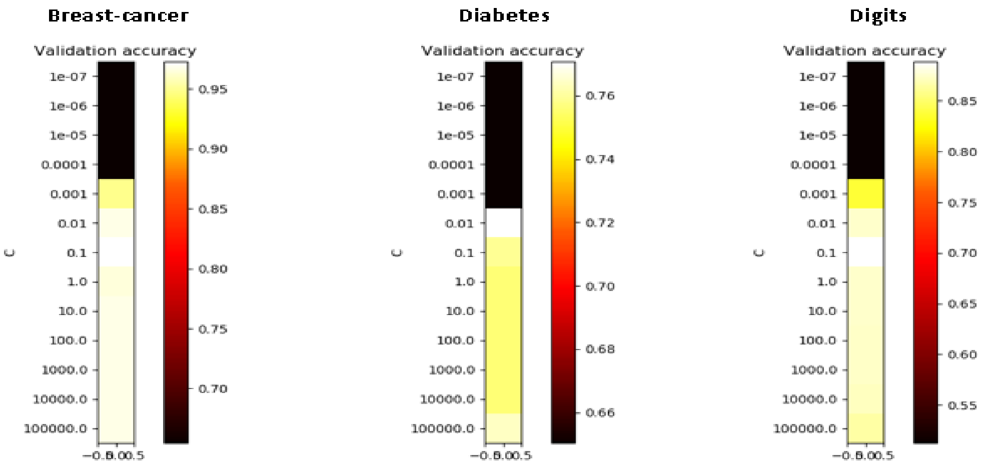
### Report the accuracy of the Linear SVM model on the training and test set

Accuracy results on five datasets are shown below.



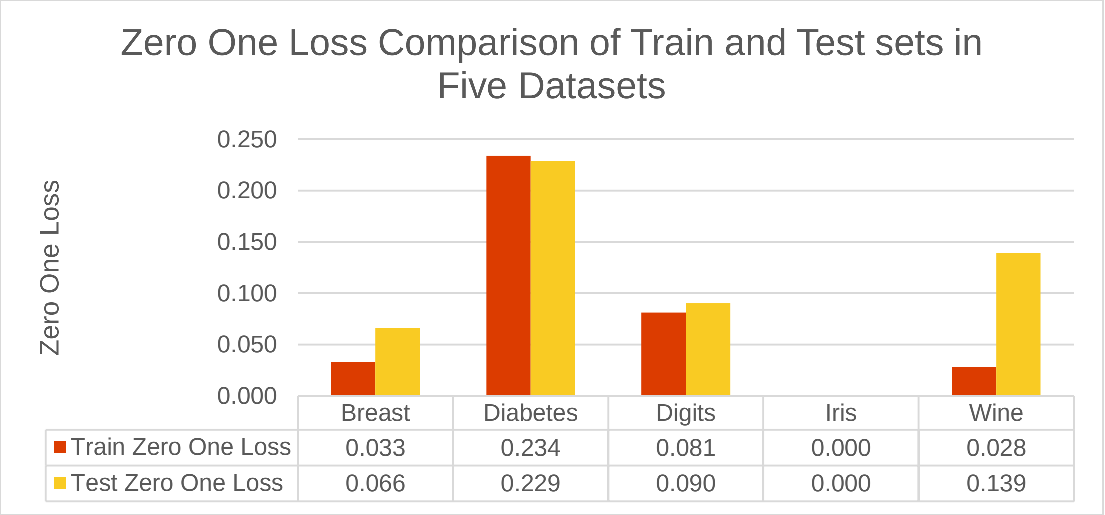
| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟦 Train Accuracy | 0.967 | 0.766 | 0.919 | 1.000 | 0.972 |
| 🟧 Test Accuracy | 0.934 | 0.771 | 0.910 | 1.000 | 0.861 |

Here I present 3 hitmaps which show accuracies of models with different parameters as an example.



**Report the negative log loss and zero one loss of the Linear SVM model on the training and test set**

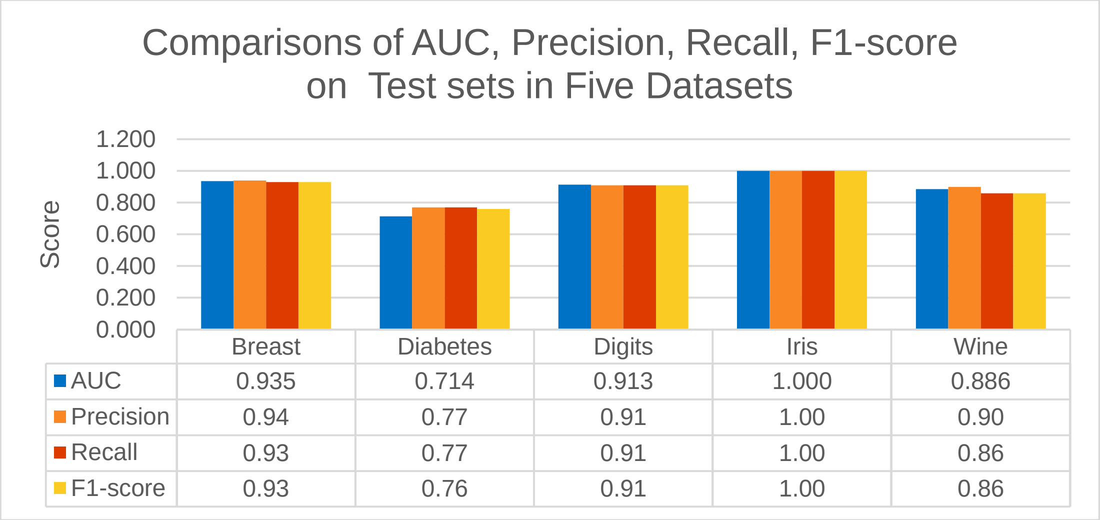
Zero one loss results on five datasets are shown below.



| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟥 Train Zero One Loss | 0.033 | 0.234 | 0.081 | 0.000 | 0.028 |
| 🟨 Test Zero One Loss | 0.066 | 0.229 | 0.090 | 0.000 | 0.139 |

**Report the AUC / Precision / Recall / F1-score and confusion matrix**

Results on AUC, Precision, Recall, F1-score, and confusion matrix are presented.

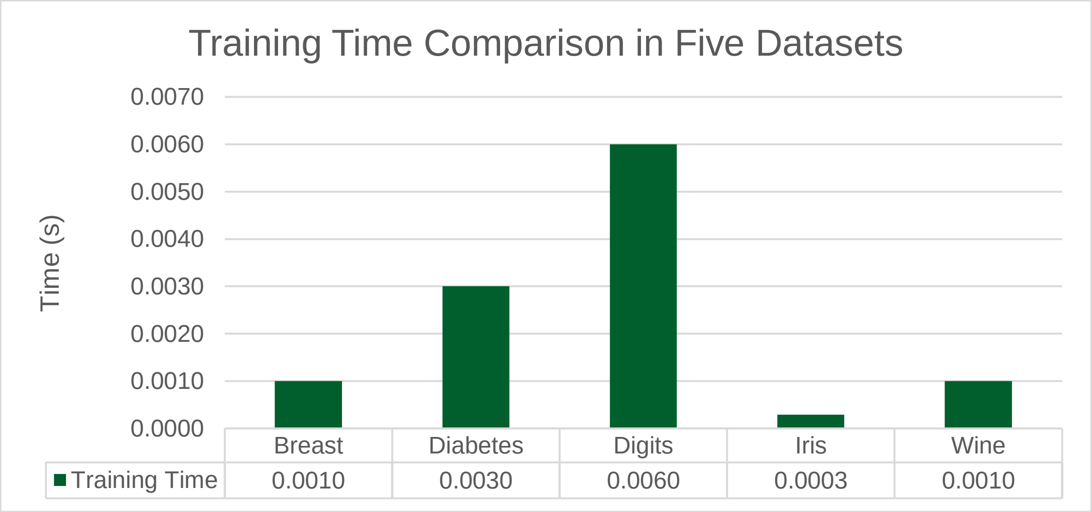


| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟦 AUC | 0.935 | 0.714 | 0.913 | 1.000 | 0.886 |
| 🟧 Precision | 0.94 | 0.77 | 0.91 | 1.00 | 0.90 |
| 🟥 Recall | 0.93 | 0.77 | 0.91 | 1.00 | 0.86 |
| 🟨 F1-score | 0.93 | 0.76 | 0.91 | 1.00 | 0.86 |

**Report the computation time for training**

Execution time is measured on a machine with the following characteristics.

- Processor: Intel® Core™ i5-7600 CPU @ 3.50GHz
- CPU(s) : 4
- Architecture : x86_64
- Memory(RAM) : 8.0 GB
- Python virtual environment : python 3.5.4 :: Anaconda



| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟩 Training Time | 0.0010 | 0.0030 | 0.0060 | 0.0003 | 0.0010 |

# 4 EMPIRICAL STUDY ON RBF SVM

In this Section, I build the RBF SVM model with SVC function in scikit-learn package. Investigations on RBF SVM model are summarized as follows.

- Present the experiment settings of the RBF SVM model
- Report the accuracy of the RBF SVM model on the training and test set
- Report the zero one loss of the RBF SVM model on the training and test set
- Report the AUC / Precision / Recall / F1-score and confusion matrix
- Report the computation time for training

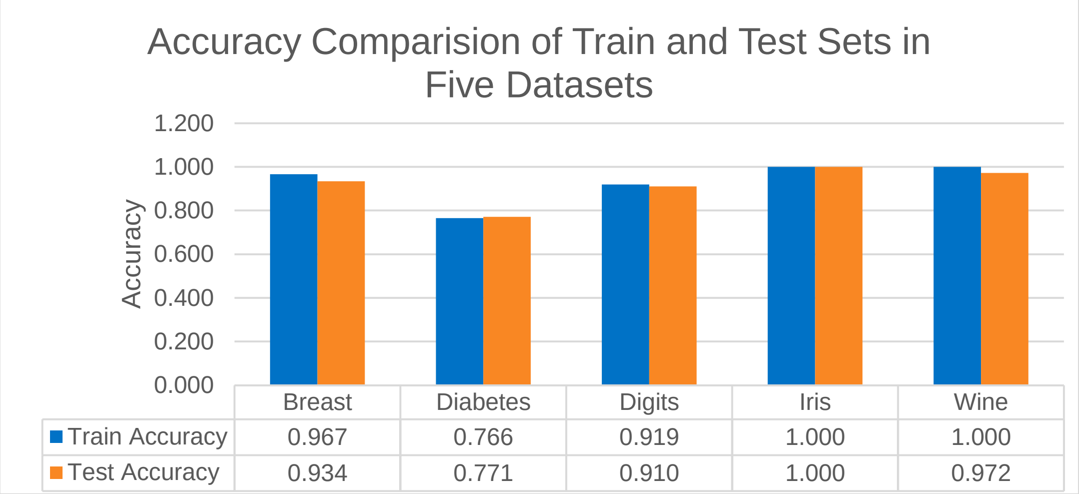
**Present the experiment settings of the RBF SVM model**

With 5-fold cross-validation method, I train many RBF SVM models with different parameters. By choosing the model that shows the best accuracy score, we identify the appropriate model for a classification task. Following parameters are used for training the model. First, 9 different gamma values are used such as [0.001,0.005,0.01,0.05,0.1,0.5,1,2,3]. Second, I vary the value of C by factors of ten from $10^{-7}$ to $10^{5}$. As a result of 5-fold Cross-validation method, the parameters which enable the model to show the best accuracy score for each dataset are as follows:

1) Breast-cancer :  Gamma value is **0.005** and C value is **10**
2) Diabetes : Gamma value is **0.005** and C value is **1.0**
3) Digits : Gamma value is **0.05** and C value is **1.0**
4) Iris : Gamma value is **0.05** and C value is **$10^{-1}$**
5) Wine : Gamma value is **0.1** and C value is **1.0**

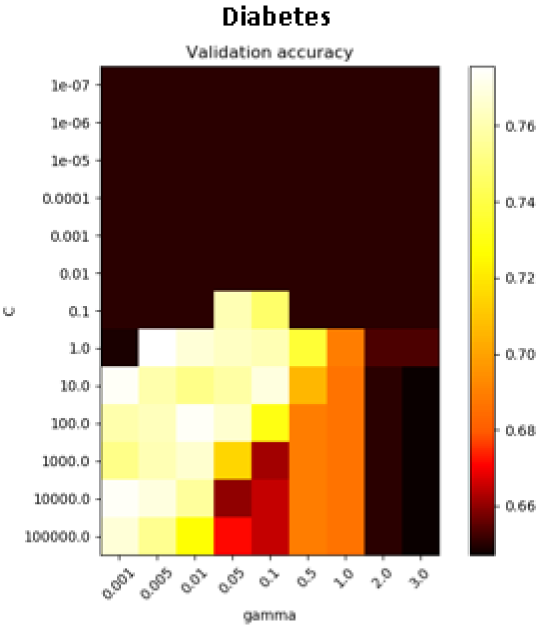
**Report the accuracy of the RBF SVM model on the training and test set**

Accuracy results on five datasets are shown below.



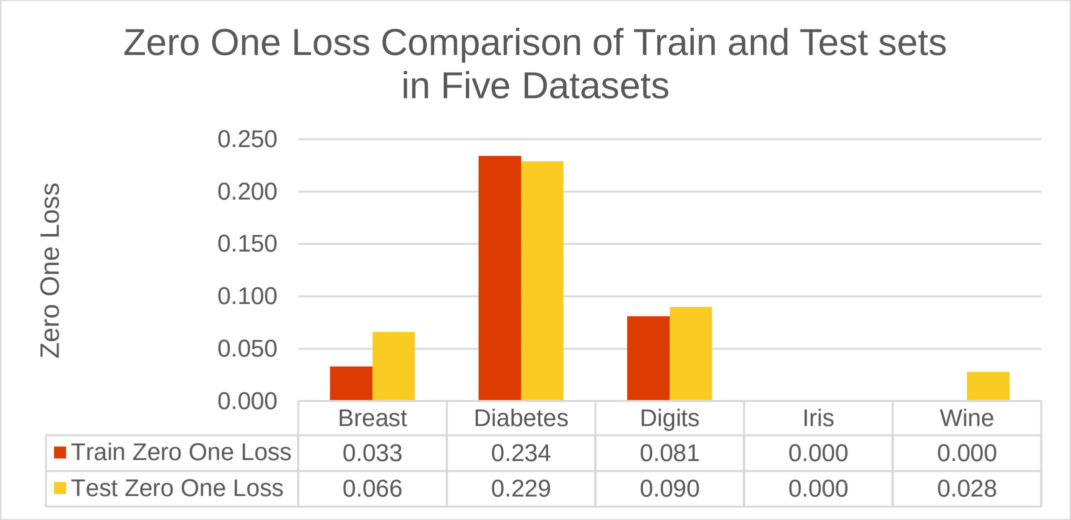
| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟦 Train Accuracy | 0.967 | 0.766 | 0.919 | 1.000 | 1.000 |
| 🟧 Test Accuracy | 0.934 | 0.771 | 0.910 | 1.000 | 0.972 |

Here I present 1 hitmap which shows accuracies of models with different parameters as an example.



**Report the negative log loss and zero one loss of the RBF SVM model on the training and test set**

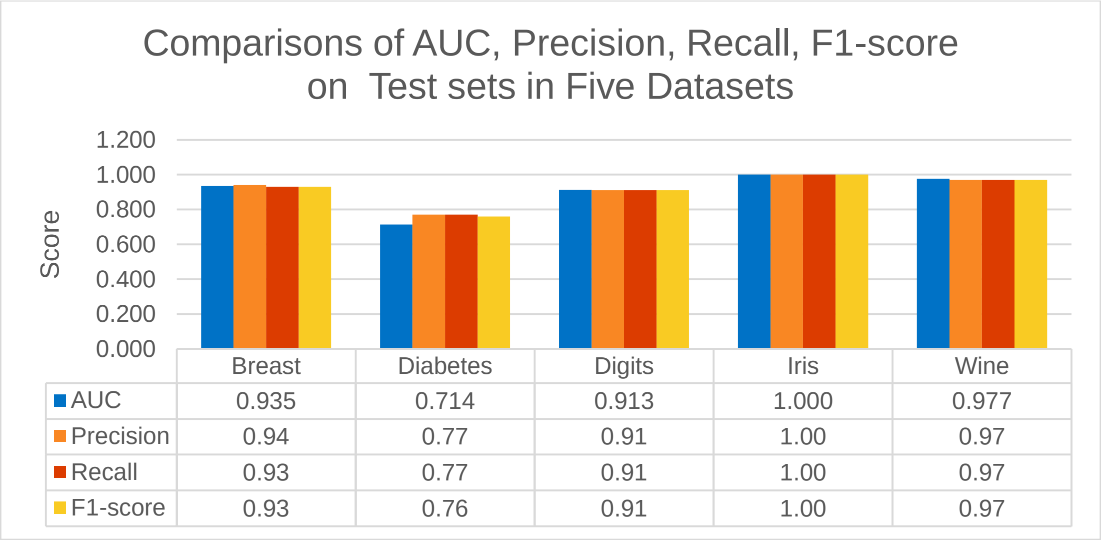
Negative log loss and zero one loss results on five datasets are shown below.



| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟥 Train Zero One Loss | 0.033 | 0.234 | 0.081 | 0.000 | 0.000 |
| 🟨 Test Zero One Loss | 0.066 | 0.229 | 0.090 | 0.000 | 0.028 |

**Report the AUC / Precision / Recall / F1-score and confusion matrix**

Results on AUC, Precision, Recall, F1-score, and confusion matrix are presented.



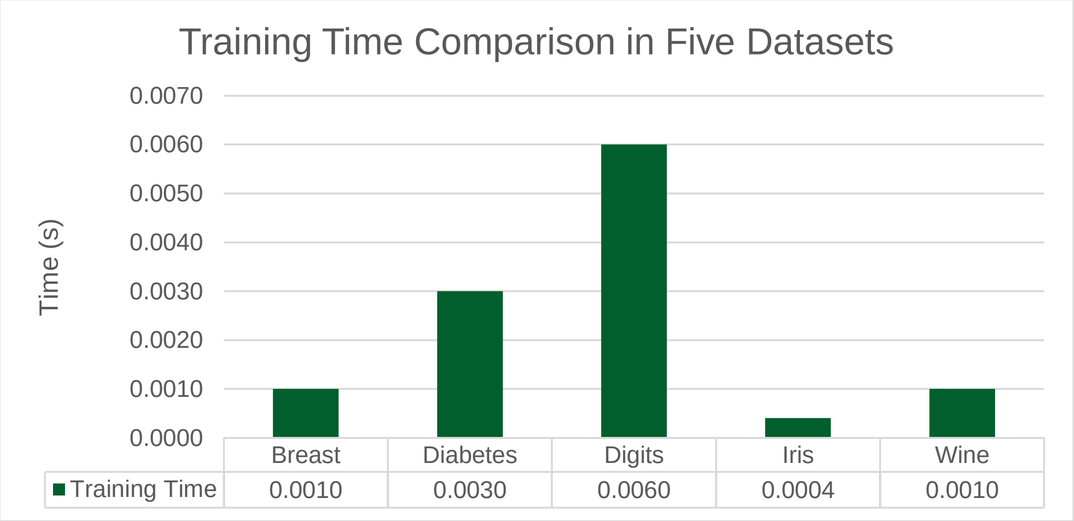
| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟦 AUC | 0.935 | 0.714 | 0.913 | 1.000 | 0.977 |
| 🟧 Precision | 0.94 | 0.77 | 0.91 | 1.00 | 0.97 |
| 🟥 Recall | 0.93 | 0.77 | 0.91 | 1.00 | 0.97 |
| 🟨 F1-score | 0.93 | 0.76 | 0.91 | 1.00 | 0.97 |

**Report the computation time for training**

Execution time is measured on a machine with the following characteristics.

- Processor: Intel® Core™ i5-7600 CPU @ 3.50GHz
- CPU(s) : 4

- Architecture : x86_64
- Memory(RAM) : 8.0 GB
- Python virtual environment : python 3.5.4 :: Anaconda



# 5 EMPIRICAL STUDY ON NN

In this Section, I build the NNs model with MLPClassifier function in scikit-learn package. Investigations on NNs model are summarized as follows.

- Present the experiment settings of the NNs model
- Report the accuracy of the NNs model on the training and test set
- Report the zero one loss of the NNs model on the training and test set
- Report the AUC / Precision / Recall / F1-score and confusion matrix
- Report the computation time for training

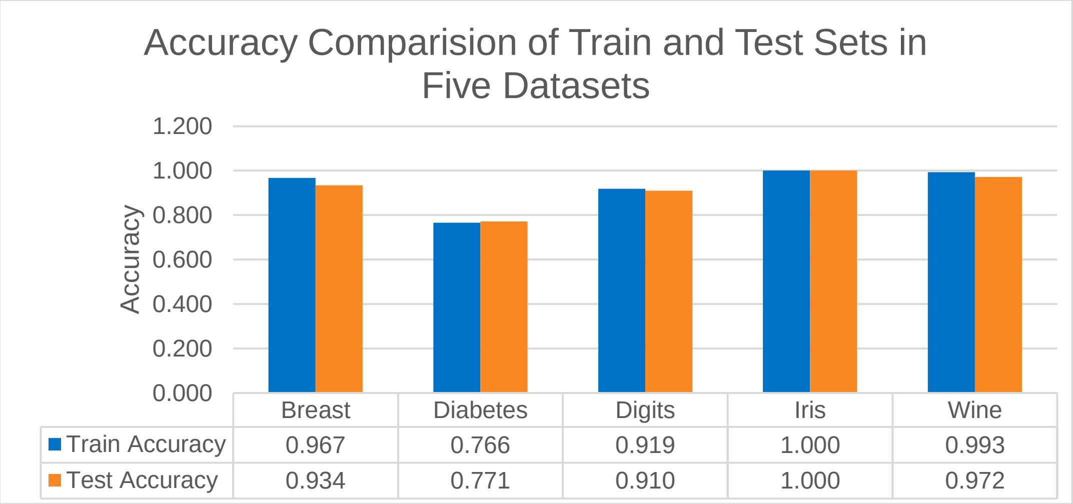
**Present the experiment settings of the NNs model**

With 5-fold cross-validation method, I train many NNs models with different parameters. By choosing the model that shows the best accuracy score, we identify the appropriate model for a classification task. Following parameters are used for training the model. First, 4 different initial learning rates are used such as [0.0001, 0.001, 0.01, 0.1]. Second, I vary the number of hidden layer sizes such as [1,2,3,4,5,6,7,8,9,10,16,32]. As a result of 5-fold Cross-validation method, the parameters which enable the model to show the best accuracy score for each dataset are as follows:

1) Breast-cancer : Initial learning rate is **0.1** and hidden layer size is **1**
2) Diabetes : Initial learning rate is **0.1** and hidden layer size is **9**
3) Digits : Initial learning rate is **0.1** and hidden layer size is **16**
4) Iris : Initial learning rate is **0.01** and hidden layer size is **2**
5) Wine : Initial learning rate is **0.01** and hidden layer size is **3**

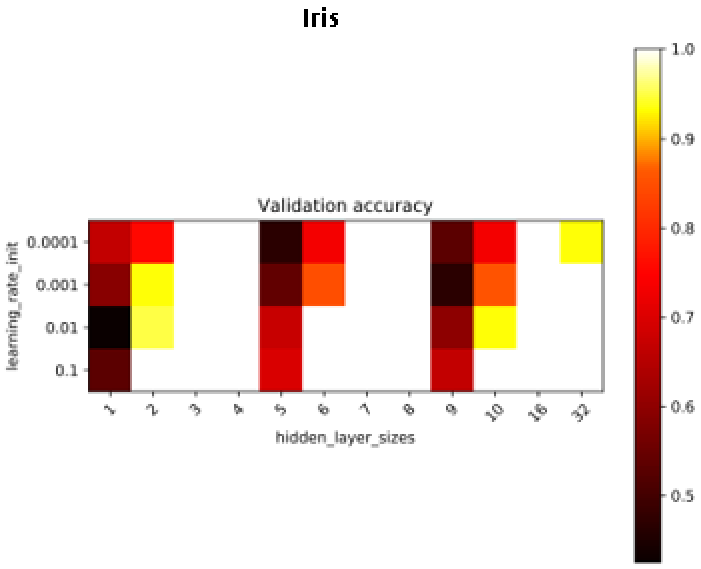
**Report the accuracy of the NNs model on the training and test set**

Accuracy results on five datasets are shown below.



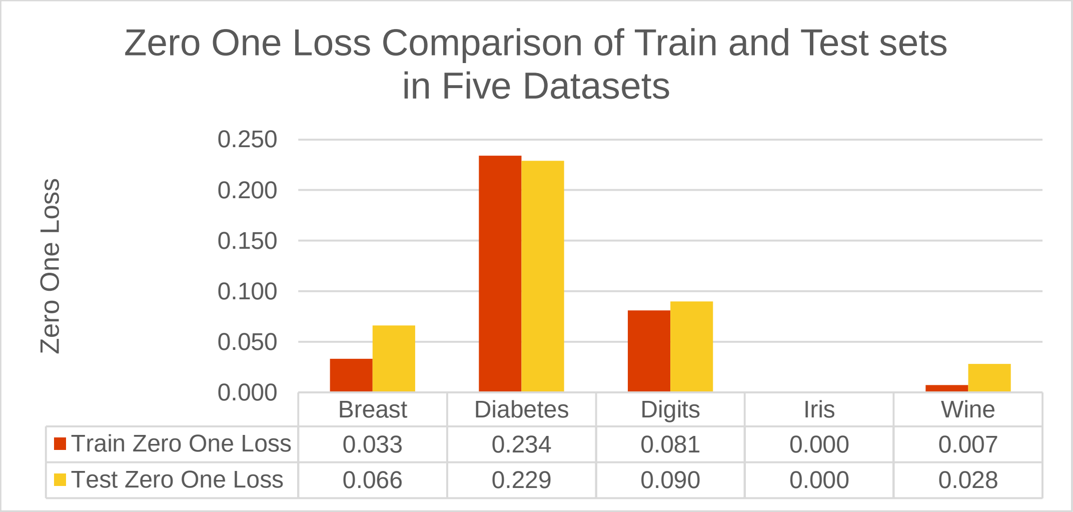
| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟦 Train Accuracy | 0.967 | 0.766 | 0.919 | 1.000 | 0.993 |
| 🟧 Test Accuracy | 0.934 | 0.771 | 0.910 | 1.000 | 0.972 |

Here I present 1 hitmap which shows accuracies of models with different parameters as an example.



**Report the negative log loss and zero one loss of the NNs model on the training and test set**

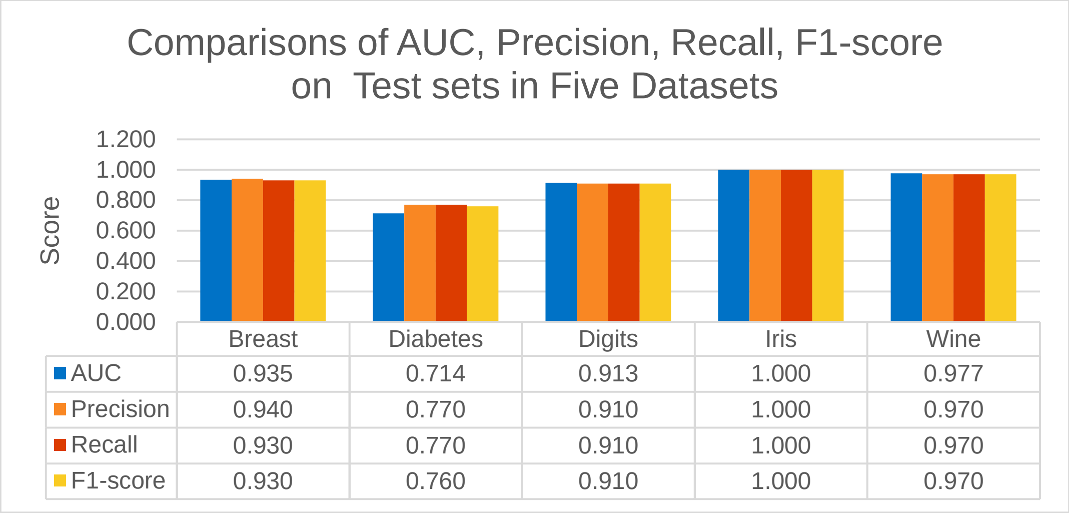
Negative log loss and zero one loss results on five datasets are shown below.



| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟥 Train Zero One Loss | 0.033 | 0.234 | 0.081 | 0.000 | 0.007 |
| 🟨 Test Zero One Loss | 0.066 | 0.229 | 0.090 | 0.000 | 0.028 |

**Report the AUC / Precision / Recall / F1-score and confusion matrix**

Results on AUC, Precision, Recall, F1-score, and confusion matrix are presented.

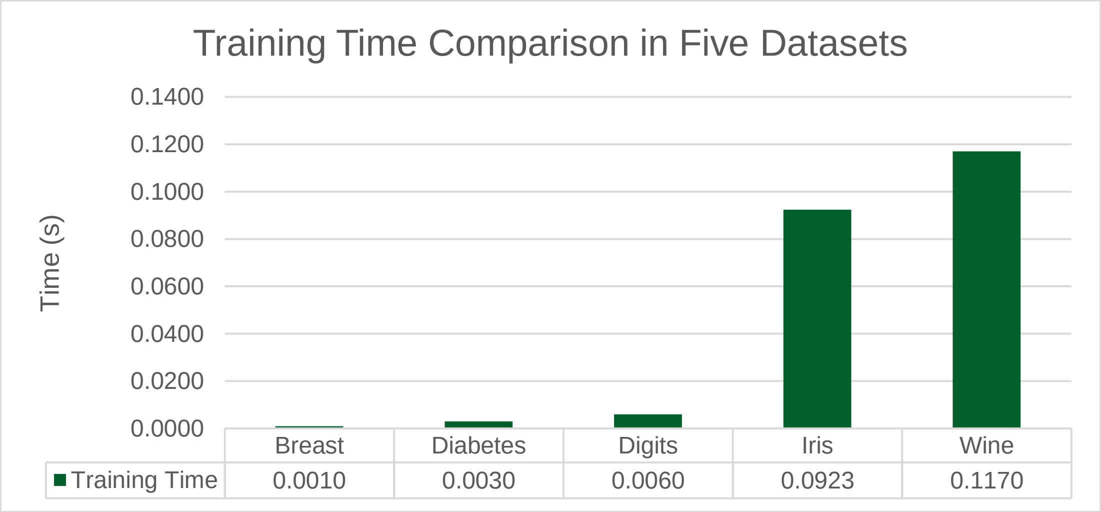


| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟦 AUC | 0.935 | 0.714 | 0.913 | 1.000 | 0.977 |
| 🟧 Precision | 0.940 | 0.770 | 0.910 | 1.000 | 0.970 |
| 🟥 Recall | 0.930 | 0.770 | 0.910 | 1.000 | 0.970 |
| 🟨 F1-score | 0.930 | 0.760 | 0.910 | 1.000 | 0.970 |

**Report the computation time for training**

Execution time is measured on a machine with the following characteristics.

- Processor: Intel® Core™ i5-7600 CPU @ 3.50GHz
- CPU(s) : 4
- Architecture : x86_64
- Memory(RAM) : 8.0 GB
- Python virtual environment : python 3.5.4 :: Anaconda



| | Breast | Diabetes | Digits | Iris | Wine |
|---|---|---|---|---|---|
| 🟩 Training Time | 0.0010 | 0.0030 | 0.0060 | 0.0923 | 0.1170 |