PUSL3123

AI and Machine Learning

Coursework

Table of Contents

[1. Introduction 3](#_Toc121146666)

[1.1 Artificial Intelligence 3](#_Toc121146667)

[1.2 Machine Learning 4](#_Toc121146668)

[1.2.1 Example: Machine Learning 4](#_Toc121146669)

[2. Literature Review 5](#_Toc121146670)

[2.1 Supervised Learning 5](#_Toc121146671)

[2.2 Unsupervised learning 8](#_Toc121146672)

[2.3 Difference Between Supervised and Un-Supervised Learning 10](#_Toc121146673)

[3. Implementation 11](#_Toc121146674)

[3.1 Methodology 11](#_Toc121146675)

[3.2 Results and Analysis 11](#_Toc121146676)

[3.2.1 Results 11](#_Toc121146677)

[3.2.2 Analysis 15](#_Toc121146678)

[4. Discussion and Conclusion 18](#_Toc121146679)

[5. References 19](#_Toc121146680)

[6. Appendix 19](#_Toc121146681)

List of Figures

[Figure 1: AI Landscape 3](file:///D:\AI_ML\fishersiris\AI_ML%20coursework.docx#_Toc121146682)

[Figure 2: Deep Neural Network 3](#_Toc121146683)

[Figure 3: Spam Email Filter ML Program Diagram 5](#_Toc121146684)

[Figure 4: Regression chart 6](#_Toc121146685)

[Figure 5: Regression equation 7](#_Toc121146686)

[Figure 6: Car price regression plot 7](#_Toc121146687)

[Figure 7: Association Rule Learning Example 9](#_Toc121146688)

[Figure 8: Clustering graph 9](#_Toc121146689)

[Figure 9: Anomaly Detection 10](#_Toc121146690)

[Figure 10: Feedforward neural net 17](#_Toc121146691)

[Figure 11: Profile summary 18](#_Toc121146692)

[Figure 12: Profile summary 18](#_Toc121146693)

# Introduction

## Artificial Intelligence

*“Artificial intelligence leverages computers and machines to mimic the problem-solving and decision-making capabilities of the human mind.”* (Education, n.d.)

**Artificial intelligence** is a field that combines computer science and large datasets to solve problems. It also includes the subfields of machine learning and deep learning, which are commonly referenced in the context of artificial intelligence. AI algorithms are used in these areas to develop expert systems that make predictions or classifications based on input data.

There are two main types of AI: **weak** **AI** and **strong AI**. Where weak AI refers to AI that has been taught and focused on specific tasks related to most of the AI applications present today. IBM Watson, Apple’s Siri, and autonomous vehicles can be presented as examples. Strong AI is a theorized kind of AI in which a machine has an intelligence comparable to humans; it has a self-aware awareness capable of solving problems, learning, and planning for the future.

Diagram

Description automatically generated

Figure 1: AI Landscape

Diagram

Description automatically generatedMachine Learning and Deep Learning are the noteworthy main fields of AI, where deep learning is a sub-field of machine learning. Deep Learning is a subgroup of AI that seeks to create computer systems that learn using neural networks similar to those seen in the human brain. While machine learning is primarily concerned with optimization, deep learning is concerned with developing algorithms that simulate how neurons in the human brain function.

Figure 2: Deep Neural Network

One of the ultimate aims of AI is for computers to think and learn in the same way that humans do. Deep learning is a set of computational approaches that take us closer to that aim by using neural networks based on the decision-making process of the human brain.

There are various real-world applications of AI including, computer vision, natural language processing, chatbots, recommendation engines etc.

## Machine Learning

Machine learning is one of the liveliest areas in artificial intelligence. Machine learning algorithms allow computers to learn new things without being programmed. They use statistics to better understand the massive amounts of data we create daily. These newer algorithms help machines classify images, sounds, and videos. They can answer our questions, discover new drugs, and even write songs.

Most people immediately think of a robot when they hear the phrase "machine learning," whether it be a trustworthy but dangerous Terminator, depending on whom you ask. However, machine learning is already a reality, not just a sci-fi dream. In some specialized applications, such as optical character recognition, it has really existed for decades (OCR). However, the spam filter was the first machine learning (ML) application to truly go mainstream and improve the lives of hundreds of millions of people back in the 1990s. Although it isn't quite a self-aware Skynet, it does technically fall under the category of machine learning (in fact, it has learned so effectively that you hardly ever need to mark emails as spam anymore).(Géron, 2017)

There are major regions and most prominent landmarks in machine learning, such as supervised versus unsupervised learning, online versus batch learning, and instance-based versus model-based learning.

*“Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed.” —Arthur Samuel, 1959*

### Example: Machine Learning

Machine Learning is fantastic for:

* Issues for which existing solutions necessitate extensive manual tweaking or lengthy lists of rules: one Machine Learning algorithm may frequently reduce code and perform better.
* Complex issues for which traditional methods offer no viable solutions at all can be solved utilizing the best machine learning techniques.
* A machine learning system can adapt to new data in changing situations.
* Gaining knowledge from massive amounts of data and complicated situations.

Machine learning systems are categorised into many different types based on how they trained (whether the program trained with human supervision or not), how they learn in the production environment, or they detect any patterns in the datasets given and more. however, all these different types of machine learning systems can be work together to achieve a desired goal.

# Literature Review

As previously mentioned, there are four main machine learning methods called supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning.

## Supervised Learning

Supervised learning is a machine learning method distinguished by the use of labelled datasets. These datasets are intended to train or "supervise" algorithms in accurately identifying data or forecasting outcomes. The model may test its accuracy and learn over time by using labelled inputs and outputs.

When it comes to data mining, supervised learning can be divided into two sorts of problems: classification and regression:

**Classification** issues employ an algorithm to accurately categorize test data, such as distinguishing apples from oranges. In the real world, supervised learning algorithms can be used to isolate spam from your email in a distinct folder. Classification methods include linear classifiers, support vector machines, decision trees, and random forests.

**Example:**

**The spam filter** is a very common example for classification, that runs on our email applications which is learning from the given examples of spam emails and normal (non-spam) emails. The examples that are used to learn by the machine learning program is called training dataset. As the result, it produces the ratio of accurately classified spam emails. The performance measure “accuracy is used to measure the results”.

A picture containing diagram

Description automatically generated

Figure 3: Spam Email Filter ML Program Diagram

**Regression** is another supervised learning technique that employs an algorithm to comprehend the link between dependent and independent variables. Regression models are useful for making predictions about numbers based on several data points, such as sales revenue forecasts for a certain company. Polynomial regression, logistic regression, and linear regression are some common regression algorithms. In regression, the task is to predict a target numeric value, such as price of a house. To train a regression algorithm, we need to give examples of houses, including both of its predictors and labels. (Predictors means the features such as location, type of the house, no of rooms, etc)

Chart, scatter chart

Description automatically generated

Figure 4: Regression chart

**Example:**

If we take the first example as a dataset with the output Y being the price of the vehicle and the input X being a collection of various qualities of various cars, such as production year, brand, mileage, and engine power.

To keep things straightforward, let's just focus on the car's manufacturing year. Next, we train the model to discover the weight parameters w and w 0 to suit the points and create our model:

Text

Description automatically generated

Figure 5: Regression equation

If we plot the problem by only considering one feature, which is the price of cars, the plot is presented as below.

Chart, scatter chart

Description automatically generated

Figure 6: Car price regression plot

**Example:**

For a business that operates in this industry, another regression example was run to determine the impact of temperature change on the sales of iced products. This enabled the business to change its production in response to any timeline (month, week, or day).

This is an actual instance of a regression that prevented a business from producing more or less goods than they are likely to sell.

Some classification and regression algorithms are interchangeable. For instance, Logistic Regression is frequently used for categorization since it may produce a value that represents the likelihood of falling into a specific category. Example: 20% chance of being spam.

***Algorithms: k-Nearest Neighbors, Linear Regression, Logistic Regression, Support Vector Machines (SVMs), Decision Trees, and Random Forests, Neural networks***

## Unsupervised learning

In **unsupervised learning**, training data is not labelled. Its like learning on their own without a teacher. These algorithms find hidden patterns in data without requiring human intervention. The following are the most important unsupervised learning algorithms.

**Clustering**

A data mining technique for categorizing unlabeled data based on similarities or differences for example, K-means clustering algorithms divide related data points into groups, where the K value defines the size and granularity of the grouping. This technique is useful for market segmentation, image compression, and other purposes.

***Algorithms: K-menas, Hierarchical cluster analysis, Expectation maximization***

**Example**:

We can look at a streaming service that wishes to improve its recommendation system for its users.

If we have meaningful data from customers, such as time spent watching movies, movie genres, and which unique television episodes they viewed, we can utilize clustering methods to classify them and determine who deserves more attention in marketing campaigns.

**Association Rule Learning**

Association Rule learning is an unsupervised learning technique that applies several criteria to discover connections between variables in a given dataset. The "Customers Who Bought This Item Also Bought" recommendation engine and market basket analysis both regularly employ these techniques.

***Algorithms: Apriori, Eclat***

**Example**:

The basket analysis is the most popular application scenario for this technique, in which we want to know if a consumer who buys product X is likely to also buy product Y. Typically, three metrics are calculated in association rules. The first is the usual conditional probability P(Y|X), which is the confidence. To conclude that the rule is strong enough, we want the value to be as close to one as possible and larger than P(Y).

It is also important to analyse the rule's support because, even with a stronger support rule, we need to know if the number of customers is very large:

Table

Description automatically generated with medium confidence

Figure 7: Association Rule Learning Example

**Example:** Suggesting a movie based on the past movies, or TV shows that the viewer watched.

**Visualization and Dimensionality Reduction**

A learning method applied when there are too many characteristics (or dimensions) in a dataset. It keeps the data integrity intact while bringing the amount of data inputs down to a bearable level. This method is frequently applied during the pre-processing of data, such as when autoencoders clean up visual data to produce better-looking images.

***Algorithms: Principal Component Analysis (PCA), Kernel PCA, Locally Linear Embedding (LLE), t-distributed Stochastic Neighbour Embedding (t-SNE)***

**Example:**

Consider the scenario when you have access to a lot of visitor data for your blog. If you want to find groups of similar visitors, you might wish to use a clustering technique. You never need to inform the algorithm which group a visitor belongs to because it figures it out on its own. It might be noted, for instance, that 20% of your site's visitors are youthful sci-fi fans who visit on the weekends, whereas 40% of your visitors are guys who enjoy comic books and read your blog most often in the evening. A hierarchical clustering technique may divide each group into more compact groups. You may use this to tailor your posts to each group.

Diagram

Description automatically generated

Figure 8: Clustering graph

Unsupervised learning algorithms can aid businesses in comprehending significant amounts of fresh, unlabelled data. These algorithms search for patterns in the data similarly to supervised learning, but the distinction is that the data is not known.

**Example:** Gathering vast volumes of data about a particular disease can aid practitioners in understanding symptom patterns and connecting them to patient outcomes. To classify all the data sources connected to a condition like diabetes would take too much time. Consequently, a supervised learning strategy may take longer to establish results than an unsupervised learning one.

**Example: Anomaly Detection**

detecting anomalous credit card transactions to avoid fraud, detecting manufacturing flaws, or automatically removing outliers from a dataset before feeding it to another learning algorithm are all examples of machine learning. The system has been trained on typical occurrences, and when it sees a new one, it can evaluate if it seems normal or if it is likely an anomaly.

**Chart, scatter chart

Description automatically generated**

Figure 9: Anomaly Detection

## Difference Between Supervised and Un-Supervised Learning

The main difference between supervised learning and unsupervised learning is **labelled data**. Simply said, an unsupervised learning algorithm does not employ labelled input and output data. Supervised learning does.

When using supervised learning, the algorithm iteratively predicts the data and modifies for the proper response in order to "learn" from the training dataset. Unsupervised learning models are more likely to be inaccurate than supervised learning models, but supervised learning techniques need human interaction up front to identify the data correctly. A supervised learning model, for instance, can forecast how long your commute will be based on the time of day, the weather, and other factors. But first, you'll need to teach it that travel time increases in rainy conditions.

Contrarily, unsupervised learning models operate independently to identify the underlying structure of unlabelled data. Keep in mind that they still need some human involvement for output variable validation. An unsupervised learning model, for instance, can determine that online shoppers frequently buy bundles of products at once. However, a data analyst would need to confirm that grouping baby clothes with a selection of diapers, applesauce, and Sippy cups makes sense. (Rose, 2018)

# Implementation

## Methodology

A direct approach in matlab was followed to load data, prepare training and test data, followed by implementing the neural network setup.

Dataset: fisheriris.mat

Training data: 60%

Test Data: 40%

Neural Network: Feedforward Recognition Neural Network

% Load Fisher’s IRIS Dataset

load iris\_dataset irisInputs irisTargets

inputs = irisInputs;

targets = irisTargets;

%  Define Hidden Layer Size

    hiddenLayerSize = 5;

% Buid the Neural Network

    net =feedforwardnet(hiddenLayerSize);

    net.divideParam.trainRatio = 60/100;

    net.divideParam.valRatio = 0/100;

    net.divideParam.testRatio = 40/100;

Build the neural network and divide train, test data and target ratios.

% Train the Neural Network 4 times with 4 different hiddenLayerSizes

    for index = 0:3

        for iteration = 0:9

            [net, tr] = train(net,inputs,targets);

            outputs = net(inputs);

            e = gsubtract(targets,outputs);

            performance = perform(net,targets,outputs);

## Results and Analysis

### Results

iris\_classification

Hidden Layer Size =5 Accuracy = 96.667%

Correct Classification Percentage : 91.666667%

Incorrect Classification Percentage: 8.333333%

-----------------------------------------------------

Hidden Layer Size =5 Accuracy = 98.000%

Correct Classification Percentage : 95.000000%

Incorrect Classification Percentage: 5.000000%

-----------------------------------------------------

Hidden Layer Size =5 Accuracy = 96.667%

Correct Classification Percentage : 91.666667%

Incorrect Classification Percentage: 8.333333%

-----------------------------------------------------

Hidden Layer Size =5 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =5 Accuracy = 98.000%

Correct Classification Percentage : 95.000000%

Incorrect Classification Percentage: 5.000000%

-----------------------------------------------------

Hidden Layer Size =5 Accuracy = 96.667%

Correct Classification Percentage : 91.666667%

Incorrect Classification Percentage: 8.333333%

-----------------------------------------------------

Hidden Layer Size =5 Accuracy = 97.333%

Correct Classification Percentage : 93.333333%

Incorrect Classification Percentage: 6.666667%

-----------------------------------------------------

Hidden Layer Size =5 Accuracy = 97.333%

Correct Classification Percentage : 93.333333%

Incorrect Classification Percentage: 6.666667%

-----------------------------------------------------

Hidden Layer Size =5 Accuracy = 98.000%

Correct Classification Percentage : 95.000000%

Incorrect Classification Percentage: 5.000000%

-----------------------------------------------------

Hidden Layer Size =5 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer 1.666667e+00 --> Average Accuracy:

------------------Starting---------------------

Hidden Layer Size =10 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =10 Accuracy = 98.667%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =10 Accuracy = 98.667%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =10 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =10 Accuracy = 98.667%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =10 Accuracy = 98.667%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =10 Accuracy = 98.000%

Correct Classification Percentage : 95.000000%

Incorrect Classification Percentage: 5.000000%

-----------------------------------------------------

Hidden Layer Size =10 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =10 Accuracy = 98.000%

Correct Classification Percentage : 95.000000%

Incorrect Classification Percentage: 5.000000%

-----------------------------------------------------

Hidden Layer Size =10 Accuracy = 98.000%

Correct Classification Percentage : 95.000000%

Incorrect Classification Percentage: 5.000000%

-----------------------------------------------------

Hidden Layer 5 --> Average Accuracy:

------------------Starting---------------------

Hidden Layer Size =15 Accuracy = 98.667%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =15 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =15 Accuracy = 100.000%

Correct Classification Percentage : 100.000000%

Incorrect Classification Percentage: 0.000000%

-----------------------------------------------------

Hidden Layer Size =15 Accuracy = 97.333%

Correct Classification Percentage : 93.333333%

Incorrect Classification Percentage: 6.666667%

-----------------------------------------------------

Hidden Layer Size =15 Accuracy = 96.667%

Correct Classification Percentage : 91.666667%

Incorrect Classification Percentage: 8.333333%

-----------------------------------------------------

Hidden Layer Size =15 Accuracy = 98.000%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =15 Accuracy = 98.000%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =15 Accuracy = 98.000%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =15 Accuracy = 99.333%

Correct Classification Percentage : 100.000000%

Incorrect Classification Percentage: 0.000000%

-----------------------------------------------------

Hidden Layer Size =15 Accuracy = 99.333%

Correct Classification Percentage : 100.000000%

Incorrect Classification Percentage: 0.000000%

-----------------------------------------------------

Hidden Layer 0 --> Average Accuracy:

------------------Starting---------------------

Hidden Layer Size =20 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =20 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =20 Accuracy = 98.667%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =20 Accuracy = 98.667%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =20 Accuracy = 98.667%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =20 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =20 Accuracy = 99.333%

Correct Classification Percentage : 100.000000%

Incorrect Classification Percentage: 0.000000%

-----------------------------------------------------

Hidden Layer Size =20 Accuracy = 99.333%

Correct Classification Percentage : 98.333333%

Incorrect Classification Percentage: 1.666667%

-----------------------------------------------------

Hidden Layer Size =20 Accuracy = 98.667%

Correct Classification Percentage : 96.666667%

Incorrect Classification Percentage: 3.333333%

-----------------------------------------------------

Hidden Layer Size =20 Accuracy = 100.000%

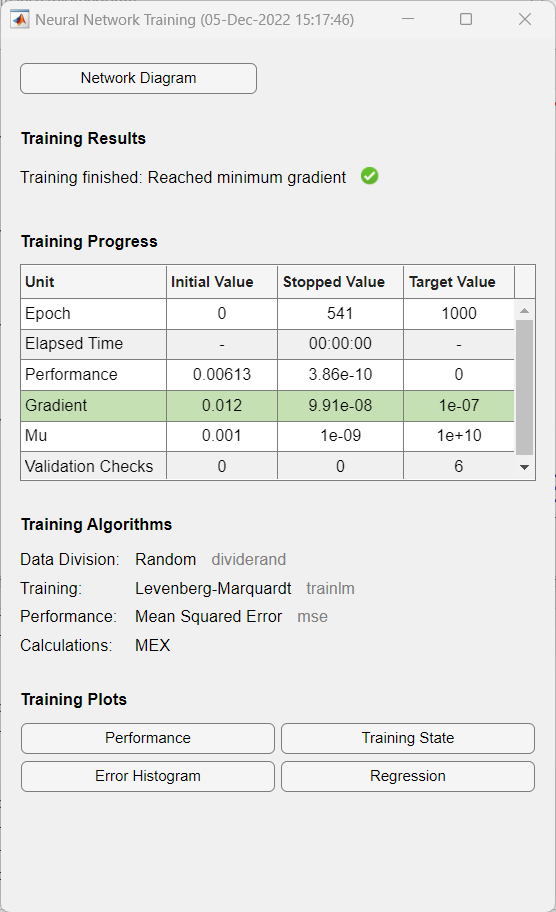
Correct Classification Percentage : 100.000000%

Incorrect Classification Percentage: 0.000000%

-----------------------------------------------------

### Analysis

Each time the result is different because we have changed the no of hidden layers of the neural network.



A picture containing text, screenshot, electronics, jack

Description automatically generated

Figure 10: Feedforward neural net

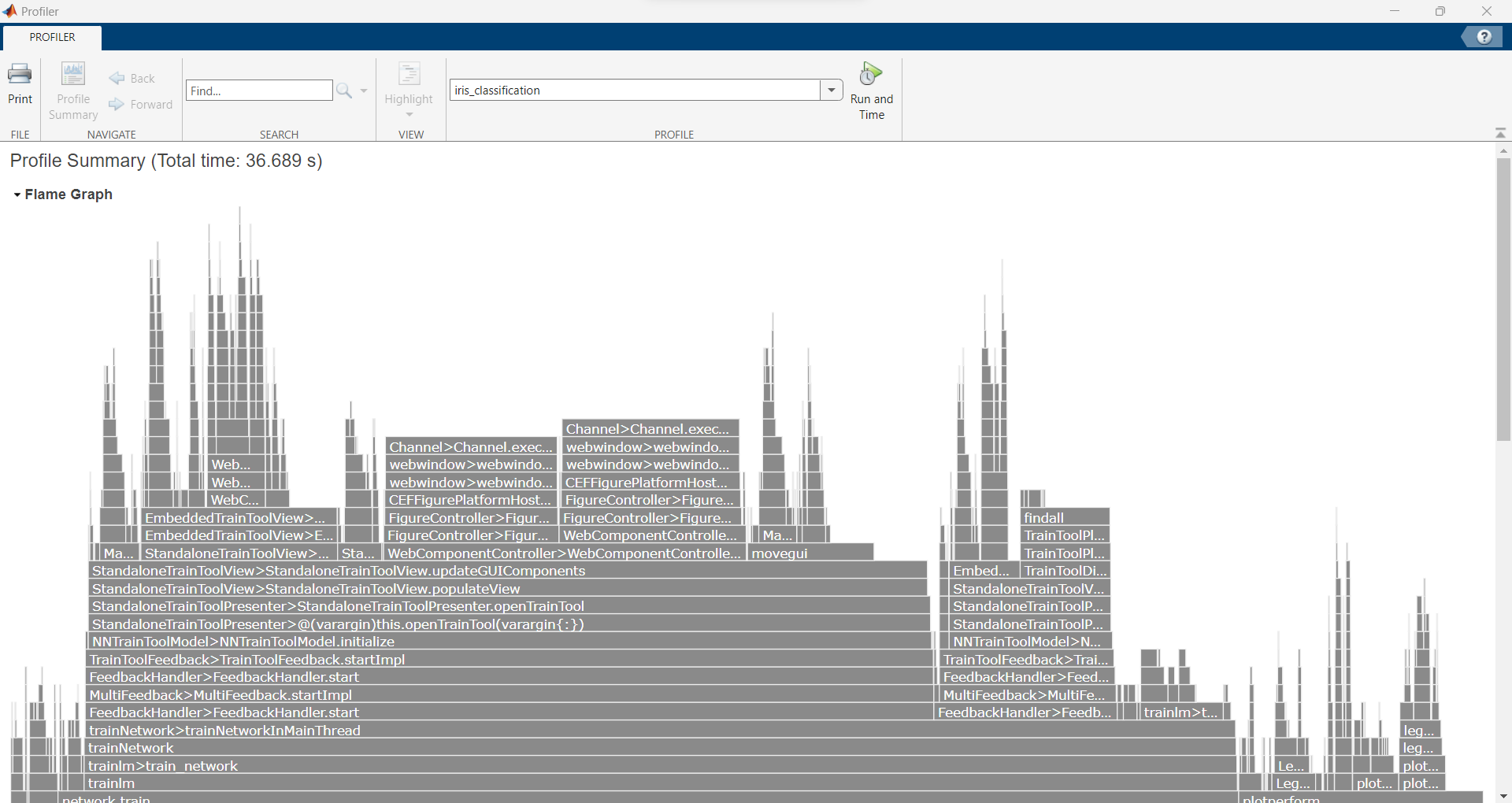


Figure 11: Profile summary

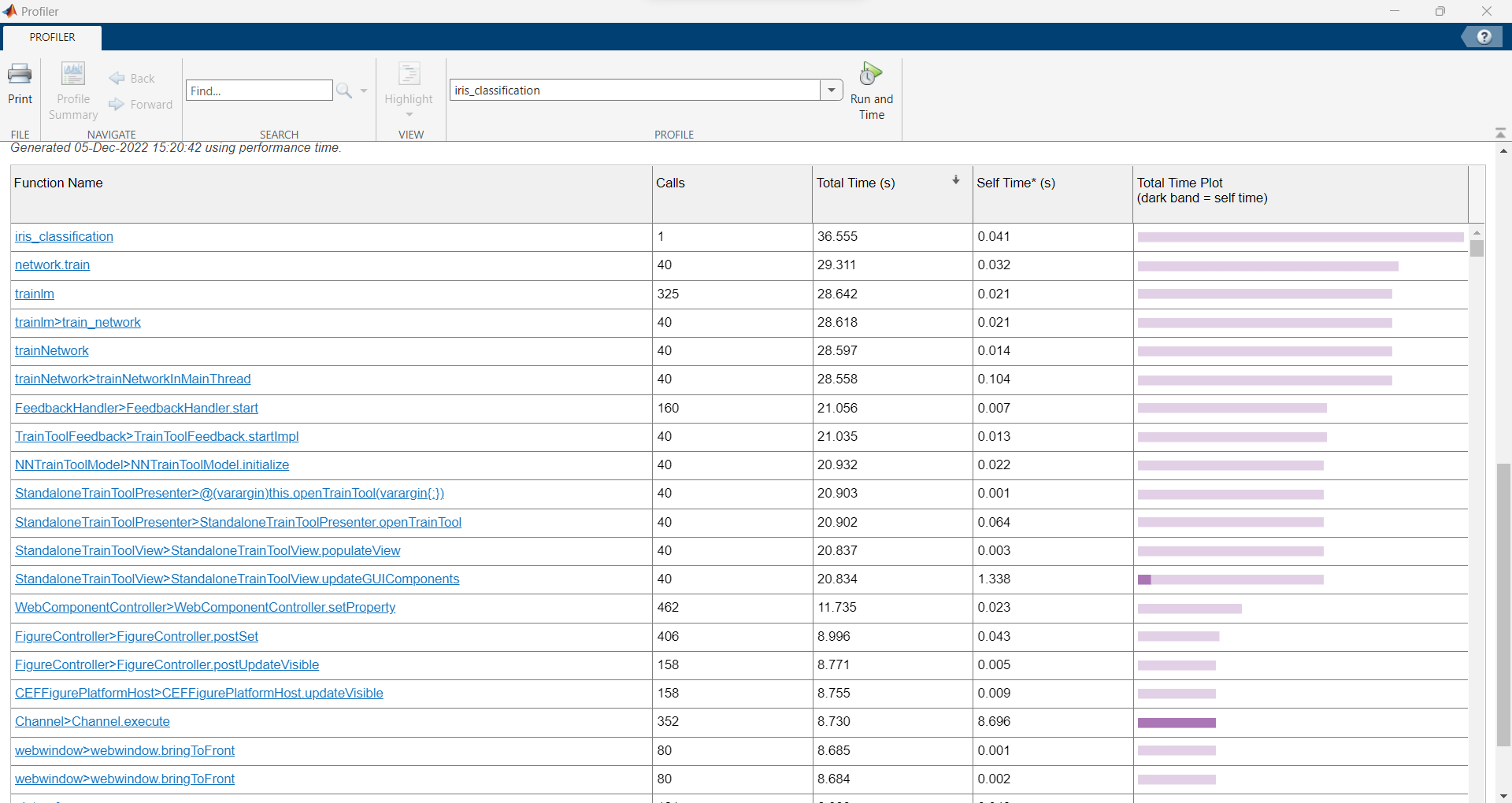


Figure 12: Profile summary

# Discussion and Conclusion

Overall, going through coursework has given the exposure to the theoretical and practical knowledge of machine learning techniques. During the neural network implementation, there was a quite a challenge on finding learning materials and proper documentations. Also, the fisher’s iris dataset was updated as well. However, model results were better using feedforward neural net.

# References

Education, I. C., n.d. *What is Artificial Intelligence (AI)?.* [Online]   
Available at: https://www.ibm.com/cloud/learn/what-is-artificial-intelligence

Rose, D., 2018. *Artificial Intelligence Foundations: Machine Learning,* s.l.: LinkedIn Learning.

Géron, A. (2017). Hands-on Machine Learning. In O’Reilly Media, Inc (Vol. 53, Issue 9).

# Appendix

% Load Dataset

load iris\_dataset irisInputs irisTargets

inputs = irisInputs;

targets = irisTargets;

%  Define Hidden Layer Size

    hiddenLayerSize = 5;

    accuracySum = 0;

% Buid the Neural Network

    net =feedforwardnet(hiddenLayerSize);

    net.divideParam.trainRatio = 60/100;

    net.divideParam.valRatio = 0/100;

    net.divideParam.testRatio = 40/100;

% Train the Neural Network 4 times with 4 different hiddenLayerSizes

    for index = 0:3

        for iteration = 0:9

            [net, tr] = train(net,inputs,targets);

            outputs = net(inputs);

            e = gsubtract(targets,outputs);

            performance = perform(net,targets,outputs);

            tind = vec2ind(targets);

            yind = vec2ind(outputs);

            percentErrors = sum(tind ~= yind)/numel(tind);

            acc = 100 \* (1 - percentErrors);

            fprintf('Hidden Layer Size =%d  Accuracy = %.3f%% \n',hiddenLayerSize, acc);

            accuracySum = accuracySum + acc;

       %    view(net);

            plotperform(tr);

            testX = inputs(:,tr.testInd);

            testT = targets(:,tr.testInd);

            testY = net(testX);

            testIndices = vec2ind(testY);

    %       plotconfusion(testT,testY);

    %       [c,cm] = confusion(testT,testY);

            fprintf('Correct Classification Percentage  : %f%%\n', 100\*(1-c));

            fprintf('Incorrect Classification Percentage: %f%%\n', 100\*c);

            fprintf('-----------------------------------------------------\n');

        end

        average = accuracySum/10;

        fprintf('Hidden Layer %d --> Average Accuracy:\n', 100\*c);

        hiddenLayerSize = hiddenLayerSize + 5;

        fprintf('------------------Starting---------------------\n');

    end