Advances in Nonlinear Variational Inequalities
ISSN: 1092-910X
Vol 27 No. 2 (2024)

A Comparative Analysis and Mathematical Modelling of Data Migration Tools to Evaluate Performance Parameters using Machine Learning Methods

Suchit A. Sapate[1], Ganesh R. Pathak[2]

[1]MIT School of Computing, MIT-ADT University, Pune, Maharashtra, INDIA
suchit.sapate2005@gmail.com
[2]MIT School of Computing, MIT-ADT University, Pune, Maharashtra, INDIA
ganesh.pathak@mituniversity.edu.in

Article History:

***Received**: 14-02-2024*

***Revised**: 24-04-2024*

***Accepted**: 14-05-2024*

Abstract:

Abstract: Transferring data between storage systems, platforms, or places is a common reason why data movement is an important part of modern computing. The effectiveness and dependability of these tasks can be greatly affected by how well data transfer tools work. Using machine learning techniques, we compare and contrast different data transfer tools in this work. First, we will talk about some important performance metrics that are needed to properly evaluate data transfer tools. Such elements include the type of data, its size, the network's speed, and other important factors like delay and flow. Next, we get data from different data transfer tools and prepare it before reviewing their performance in a range of situations and setups. The data is analyzed using machine learning techniques like classification, regression, and grouping. Finding groups of tools that work similarly is easier with clustering. This gives you information about which tools are best for different data transfer jobs. We can describe the link between performance factors and tool performance under different situations using regression analysis. The performance traits of tools help with classification, which makes it easier to choose the right tool for a transfer job. Our findings demonstrate that machine learning techniques can correctly evaluate and contrast data transfer tools by looking at how well they do certain tasks. The paper study shows how these approaches can help companies pick the best tool for their data transfer needs by giving them useful information about tool choices. Furthermore, our research shows how important it is to use machine learning when assessing data transfer tools and sets the stage for more research to be done in this area in the future.

Keywords: Data migration tools, Performance evaluation, Machine learning methods, Comparative analysis, Tool selection

1. INTRODUCTION

In today's computer world, data transfer is a necessary step that is often required by system updates, data center consolidations, or the use of cloud storage options. Traditionally, data transfer has been done by hand or with scripts, which can take a long time, make mistakes, and not work very well. In the past few years, there has been a growing interest in using machine learning (ML) techniques to make data transfer processes faster and more accurate. Moving data from one system to another is called data migration, and it is an important part of the IT field. It's often needed to move data to the cloud, upgrade systems, or combine data centers. The success of a data movement project relies on many things, such as how well the tools used for the transfer work [1]. The job of these tools is to make sure that data is sent quickly, correctly, and safely. Several things, like the type of data being moved, the size of the data set, and the network speed, can affect how well data movement tools work. Performance can also be affected by things like delay, flow, and the presence of data compression methods. When choosing the best data transfer tool for a job, it's important to look at how well each one works in a variety of situations [2]. We want to look at and compare different data transfer tools based on how well they work in this study. We mostly look at tools using machine learning techniques, which can tell you a lot about how well they work and help you choose the best one for a data transfer job. Our goal is to give you a full picture of all the data transfer tools out there by looking at performance aspects like data type, data size, network speed, and other important things. The main ideas behind data transfer and the things that affect how well a tool works.

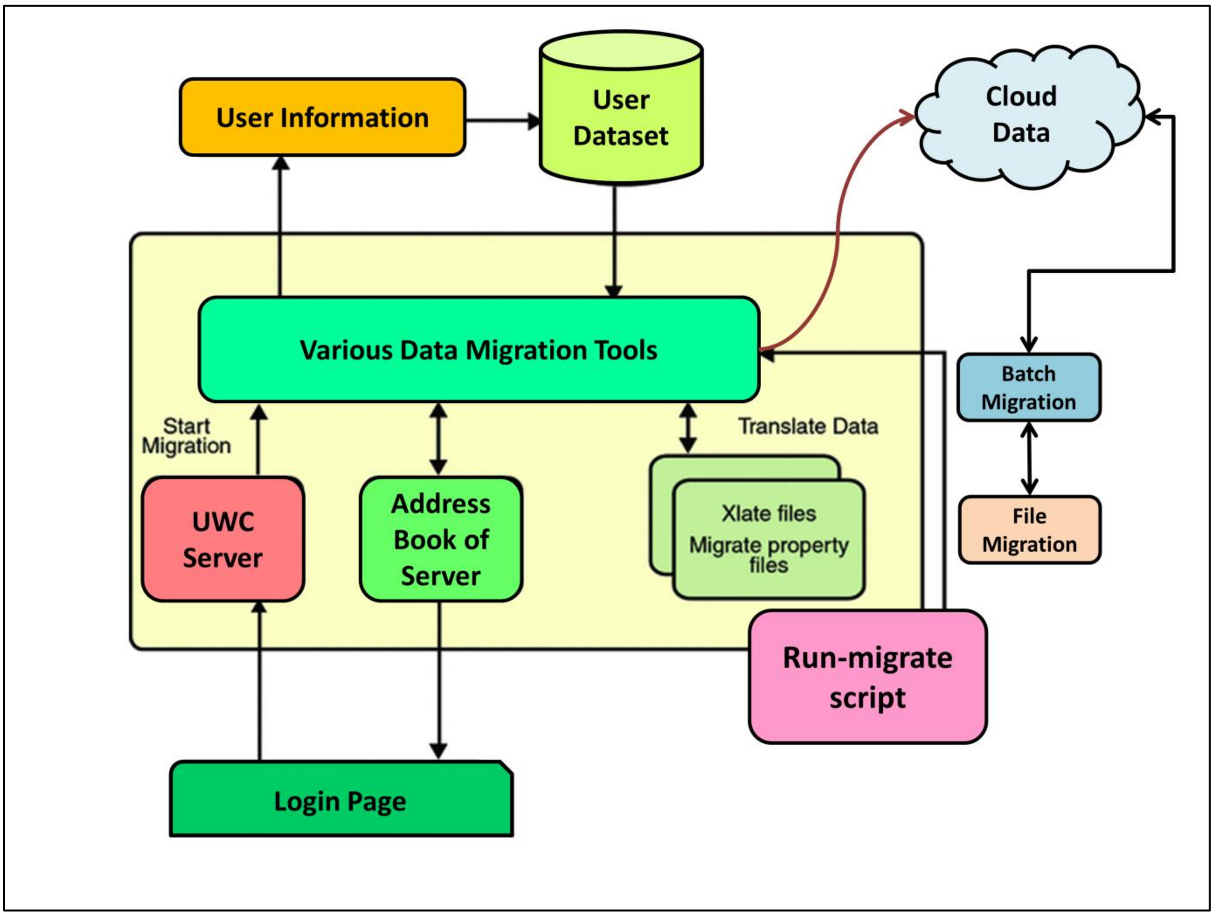


Figure 1: Overview of data Migration Process

Planning, carrying out, and checking the transfer process by hand is what most people do when they migrate data the old-fashioned way. To [3] use this method, you need to know a lot about both the source and goal systems, as well as the data that you want to move. Data mapping, processing, and checking are often done by hand or with the help of simple tools, which can be time-consuming and error-prone. Figure 1 shows a big picture of the whole data movement process, showing important steps from evaluating and planning to tracking after the transfer. The section on user information stresses how important it is for users to be involved and managed during the transfer process. The figure also talks about batch migration, highlighting how it can be used to quickly move big amounts of data. The cloud part talks about the pros and cons of moving data to the cloud, which is becoming more popular. Overall, the figure description covers all the important things to think about and the complicated parts of moving data successfully, giving a full picture of the process. Also, the old way of doing things might not work well with big or complicated data sets, which could make moving take longer and make it more likely that data will be lost or changed. ML-based data transfer, on the other hand, has several benefits over older methods. ML systems can quickly and correctly look at a lot of data, which helps find trends, outliers, and relationships that could affect the transfer process [4]. ML can also handle many parts of data transfer, such as data analysis, mapping, transformation, and validation. This cuts down on the work that needs to be done by hand and makes things run more smoothly overall. There are many problems that can happen during data transfer, such as data loss, data corruption, and data that doesn't match up. If you use the wrong data transfer tools, these problems can get worse. So, it's important to test data transfer tools fully before putting them to use in a live setting. Different things, like the type of data being moved, the size of the data set, and the network speed, can affect how well data movement tools work [5]. Performance can also be affected by things like delay, flow, and the presence of data compression methods. When choosing the best data transfer tool for a job, it's important to look at how well each one works in a variety of situations. Machine learning helps businesses choose the right data transfer tools and complete successful data migration projects. We are using machine learning to look at and compare data transfer tools based on how well they do certain tasks. We want to help groups choose the best data transfer tool for their needs by giving them a full review of all the available tools [6].

One of the best things about the ML method is that it can be changed to fit new transfer needs and data types. Machine learning systems can learn from past migrations and use what they've learned to make future migrations better [7]. ML algorithms can, for instance, change the rules for data mapping and transformation automatically when the source or target system changes [9]. This lowers the chance of mistakes and makes sure the data is correct. The ML method can also handle complicated data types and structures, which is another benefit. ML systems can work with text or video files that aren't organized or are only partially organized, which can be hard to move using old-fashioned methods. ML can also find and fix possible data quality problems, like missing or incorrect data, before the transfer process starts. This lowers the chance that data will be lost or damaged [10]. With using machine learning to move data has many benefits over older methods, such as being more efficient, scalable, and flexible. ML algorithms can help organizations speed up their data transfer processes, lower the chance of mistakes, and make sure the security of their data while it's being moved. And as machine learning keeps getting better, it will probably become a bigger part of data transfer in the future, making it easier and faster for businesses to move their data.

2. RELATED WORK

Moving data is an important part of modern computing, and how well data migration tools work can have a big effect on how well migration projects go. Traditional data movement includes moving data from one system to another through a number of human steps and automated steps. This method has been used for a long time and is still often used today, especially in smaller businesses or for easier transfer jobs. In the old method, one of the most important steps is data analysis, which looks at the source data to figure out its structure, style, and quality. A migration plan is made with this information. This plan shows the steps that need to be taken to move the data to the target system [11].

The real data transfer process starts once the move plan is made. Usually, this means getting the data from the source system, changing it into a file that the target system can understand, and then putting it into the target system. This process can take a long time and require a lot of work, especially if the data sets are big or complicated. Another problem with the old way of doing things is that data can get lost or messed up. There is a bigger chance of mistakes happening during the transfer process because most of it has to be done by hand. This can cause data to be lost or damaged, which can make fixing the problem take longer and cost more. One more thing is that the old way of moving data might not work well for big or complicated images [12]. With the old way of doing things, it might be hard for companies to finish moves on time and efficiently as data amounts keep growing and migration needs get more complicated. The standard way of moving data does have some benefits, even with these problems. Like, it might be less expensive than using specialized data transfer tools or services, especially for smaller businesses or moving jobs that aren't too complicated. Recently, people have become more interested in using machine learning (ML) techniques to check how well data transfer tools work. Several studies have looked into this subject, using different ML methods and focusing on different parts of data movement [13]. Evaluating data transfer tools based on how well they work is one of the most important areas of study in this field, for example, did a study that looked at how well different data transfer tools worked based on things like the type of data, the size of the data, and the network speed. They put tools together that worked the same way by using grouping and regression analysis to guess how the tools would work in different situations. It [14] also came up with a way to evaluate data transfer tools that uses machine learning techniques. They got information from a number of different transfer projects and used algorithms to put tools into groups based on how well they worked. Their work showed that machine learning can be used to evaluate data transfer tools and showed how important it is to look at a number of performance factors when assessing a tool.

ML has been used to do more than just evaluate performance. It has also been used to make data transfer processes faster and more reliable. For instance, author suggested using reinforcement learning to find the best ways for moving data [15]. Their method used old migration data to teach a model that could figure out the best migration path for any set of data.

This cut down on migration times and made the whole process more efficient. Another area of study is how to use ML to plan and carry out data migrations. For instance, suggested a method for using ML techniques to automate chores like moving data [16]. Their framework looked at data from both the source and target systems to make migration plans and tasks that were carried out automatically. This [17] cut down on the need for human assistance and made migration more efficient. Even with these improvements, there are still some problems and restrictions with using ML to test data transfer tools. One of the biggest problems is that there aren't any regular ways to measure how well data transfer tools work. It's hard to compare data from different studies because most of them use their own measures or success indicators. Another problem is that there isn't a lot of data available to train ML models. It can be hard to gather and share data for study because data transfer projects are often complicated and involve private data. This makes it harder for ML models that were learned on small samples to be used in other situations [18].

Also, because data transfer projects are so complicated, it's hard to make machine learning models that can correctly predict how tools will work in every situation [8]. ML models are usually based on old data, which might not fully show how complicated transfer projects are in the real world. AI is a great way to test data transfer tools, but there are some problems that need to be fixed. As machine learning keeps getting better and more data comes out, researchers and practitioners are likely to come up with more complex models and ways to test data transfer tools. This will make the process of moving data faster and more reliably. In Table 1, discussed about the work that is likely to be in a study paper. It probably has sections with information like the title of the work, the writers, the place where it was published, the year, and a short summary of what the work adds or finds. This table 1 makes it easy for readers to quickly understand the current state of research, find important works in the field, and understand how and why the current study or project fits into the bigger picture of research.

Table 1: Summary of Related Work

Approach	**Algorithm**	**Type of Data Transfer Tool**	**Application**	**Scope**
Clustering	K-means clustering	General purpose data migration tools	Evaluate performance based on data type, size, network bandwidth	Comparative analysis of data migration tools
Classification	Random Forest	Data migration tools	Categorize tools based on performance profiles	Evaluation of data migration tools
Reinforcement learning	Q-learning	Data migration optimization tools	Optimize data migration paths	Performance optimization of data migration processes
Automation	Decision trees	Data migration automation tools	Automate data migration tasks	Improve efficiency of data migration processes
Predictive modeling	LSTM (Long Short-Term Memory)	Data replication tools	Predict performance under different conditions	Predictive analysis of data migration tool performance

Pattern recognition	Support Vector Machines (SVM)	Data synchronization tools	Identify patterns in migration processes	Improve data synchronization efficiency
Anomaly detection	Isolation Forest	Data validation tools	Detect anomalies in migration processes	Enhance data validation in data migration
Regression analysis	Linear regression	Data transformation tools	Predict performance based on data characteristics	Improve data transformation efficiency
Optimization	Genetic algorithms	Data deduplication tools	Optimize data deduplication processes	Enhance efficiency of data deduplication
Simulation	Monte Carlo simulation	Data compression tools	Simulate data compression processes	Evaluate performance under different compression scenarios
Ensemble modeling	Random Forest, Gradient Boosting	Hybrid data migration tools	Improve overall tool performance	Enhance performance of hybrid data migration tools
Deep learning	Convolutional Neural Networks (CNN)	Image-based data migration tools	Evaluate performance of image-based migration tools	Assess image-based data migration tool performance
Meta-learning	Meta-gradient reinforcement learning	Meta-learning based data migration tools	Improve adaptability of data migration tools	Enhance adaptability of data migration tools
Evolutionary algorithms	Genetic algorithms	Evolutionary data migration tools	Optimize migration strategies over time	Improve migration strategy optimization

3. DATASET USED

A. MINIST

It is common to use the MNIST collection in computer vision and machine learning. Assembled from 70,000 grayscale pictures of scribbled numbers from 0 to 9, each one measuring 28x28 pixels. An image training set of 60,000 pictures and an image test set of 10,000 images make up the collection. Researchers often use the MNIST dataset to see how well machine learning methods work, especially when it comes to recognizing numbers. For training and testing their models, researchers and practitioners use this dataset to see how their results compare to those of other methods. Because each picture in the MNIST collection is tagged with the number it represents, it can be used for guided learning tasks. The information is usually clean and well-organized, which makes it simple for both new and experienced researchers to work with [23]. The MNIST collection is useful in part because it is simple and easy to use. Due to their small size and single number, the pictures are perfect for trying basic image processing and classification methods. Furthermore, the dataset is well-documented, with clear labels and a standard file that makes it simple to download and use for study reasons. With its shared standard, the MNIST dataset is an important dataset for analyzing and comparing different methods. With its simplicity and ease of use, it's a great place for students and professionals to start learning about computer vision and pattern recognition.

B. BEEF

The BEEF dataset, which is also called the LOCBEEF dataset, is a set of pictures of local Aceh beef that were taken between 7:00 a.m. and 22:00 p.m. There are 3268 pictures in the file, which are split into two groups: fresh and rotten. There are also learn and test folders within these groups [24]. There are 2228 pictures in the train directory, and each folder has 1114 pictures. In the same way, there are 980 pictures in the test directory, with 490 images in each folder. Different sizes of pixels are used for the images in the collection, such as 176 x 144 pixels, 320 x 240 pixels, 640 x 480 pixels, 720 x 480 pixels, 720 x 720 pixels, 1280 x 720 pixels, 1920 x 1080 pixels, 2560 x 1920 pixels, 3120 x 3120 pixels, 3264 x 2248 pixels, and 4160 x 3120 pixels. Using the deep learning method of Convolutional Neural Networks (CNN), the dataset was used for classification, especially to tell the difference between fresh and spoiled beef. The dataset was split into two parts: training and testing. The training set has 70% of the data and the testing set has 30%. The pictures in the collection can be used with the Resnet50 design, which is a well-known deep learning model for sorting images into groups. Overall, the BEEF dataset is a useful tool for both students and professionals working in computer vision and picture classification, especially when it comes to judging the quality of food.

4. SYSTEM CONFIGURATION

A. Microsoft Azure:

The Microsoft Azure cloud computing platform lets you build, launch, and manage apps and services across Microsoft's worldwide network of data centers. It does this by providing a variety of services and tools. Azure Machine Learning is a cloud-based tool for building, training, and delivering machine learning models. It is one of the most important services that Azure provides. The Azure Machine Learning Studio is an Integrated Development Environment (IDE) that you can use on the web to create and use machine learning models on Azure. It has a drag-and-drop interface that makes it simple for people to make machine learning systems without having to write any code [19]. To make machine learning processes that are more complicated, users can also write their own tools in languages like Python and R. There are many tools and features in Azure Machine Learning Studio that can be used to create and use machine learning models. Some of these are data preparation tools that clean and change data, different machine learning methods for teaching models, and tools that check and compare how well different models are doing.

B. Configuration Details

One great thing about Azure Machine Learning Studio is that it can be expanded. It is easy for users to make their machine learning projects bigger so that they can work with big datasets and complicated models [20]. Azure Machine Learning Studio works with other Azure services, like Azure Cosmos DB, to store and manage data. The Azure Free Tier provides 5GB of storage and 400 Request Units per second of Azure Cosmos DB per month for free.

5. METHODOLOGY

We suggest a way with a few important steps that can be used to measure the success of data transfer tools that use machine learning. These steps are getting the data, preparing it, choosing the features, training the model, and evaluating it. The way is meant to give a full picture of data transfer tools by looking at how well they work in different situations [21].

A. Data Collection:

We get our data from a lot of different places, like data transfer projects, books, and freely available datasets. The data should have details about the tools used for data transfer, the types of data being moved, and the efficiency measures that are important. Figure 4 shows a system flow diagram made just for working with CSV files. It shows the steps that need to be taken in order to process this kind of data. The picture starts with getting CSV files, which are often used to store data in tables. The next step is to read and parse the CSV files to get the data they contain. After this, data preparation is done, which includes things like cleaning the data, dealing with missing values, and changing the data into a file that can be used for analysis or further processing. After that, the data is analyzed. Depending on the goals of the analysis, this could include statistics analysis, data display, or data models.

B. Getting ready:

The gathered data is first handled to make sure it is consistent and of good quality. To do this, the data needs to be cleaned up by getting rid of copies and dealing with missing numbers.

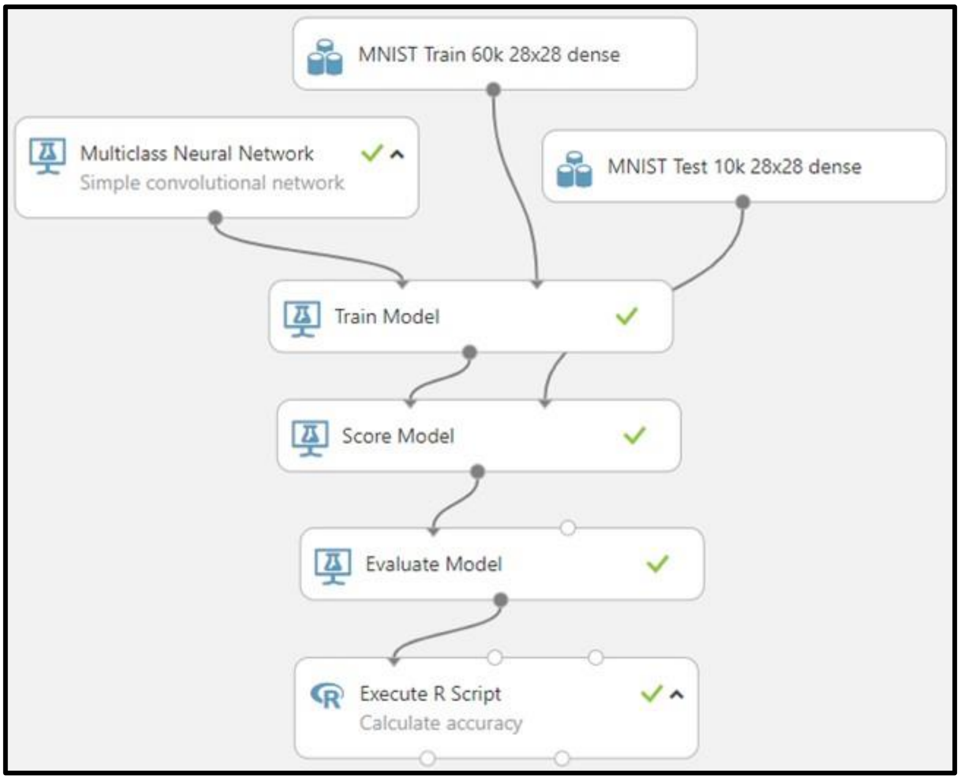


Figure 2: System flow for Image dataset

The data is then changed into a style that can be analyzed, like a list of numbers or a list of categories. Figure 2 shows the system flow for organizing an image dataset. It shows the steps that are needed to process and organize image data. The flow starts with data collection, which is when pictures are taken or gathered from different devices or sources. The next step is preparation, which cleans, normalizes, and resizes pictures so they are ready for the next step. After the pictures are preprocessed, feature extraction is used to pull out important features that can then be used for classification or analysis. The features are then put into a machine learning or deep learning model so that it can learn from them or draw conclusions from them. Once the model has been processed, the results are looked at to find useful information or groups. Lastly, the data or insights that have been handled are saved or shown so that they can be analyzed further or shown to others.

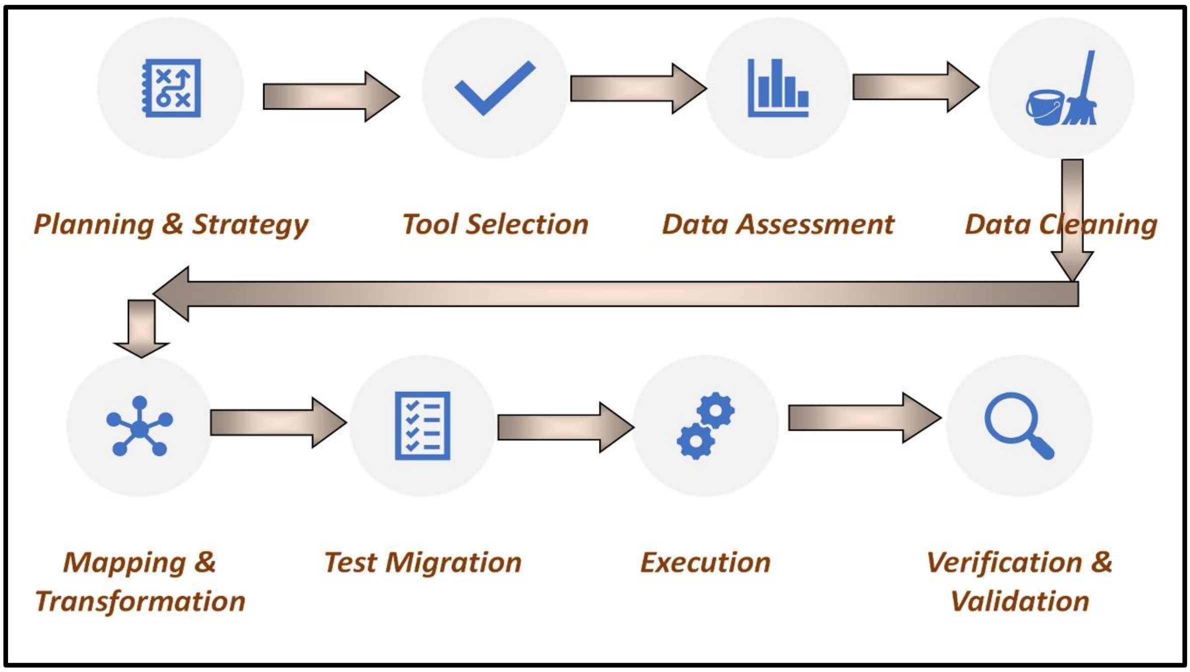


Figure 3: Illustrating the process flow involving data migration tools.

C. Feature Selection:

We choose features that are important and likely to have an effect on how well data transfer tools work. Some of these traits could be the type of data, its size, the network speed, and other important factors [22]. To find the most important features, methods like association analysis or feature importance ranking are used to choose them. The steps and parts needed to move data from one system to another are shown in Figure 3 as a process flow diagram for data transfer tools. The first step in the process is choosing the right data transfer tool based on the needs and properties of the data that needs to be moved. The source data is then taken out and changed into a file that can be used for transfer. The chosen transfer tool is then used to put the changed data into the target system. The data transfer process is finished when the moved data is checked to make sure it is correct and full.

E. Evaluation:

A different set of data is used to test the learned models and see how well they do. It is possible to rate how well the models work by looking at things like their accuracy, precision, memory, or F1-score. To figure out how well the models work, they are also compared to baseline models or old-fashioned methods.

F. Interpretation and Analysis:

Lastly, we look at how the review results can be interpreted to learn more about how well data transfer tools work. We look at how different factors affect the performance of tools and try to find patterns or trends that could help make data moving processes more reliable and efficient. By using this method, we can successfully test the performance factors of data transfer tools that use machine learning. This method gives a thorough and organized look at everything, which can help companies pick the best data transfer tools for their needs.

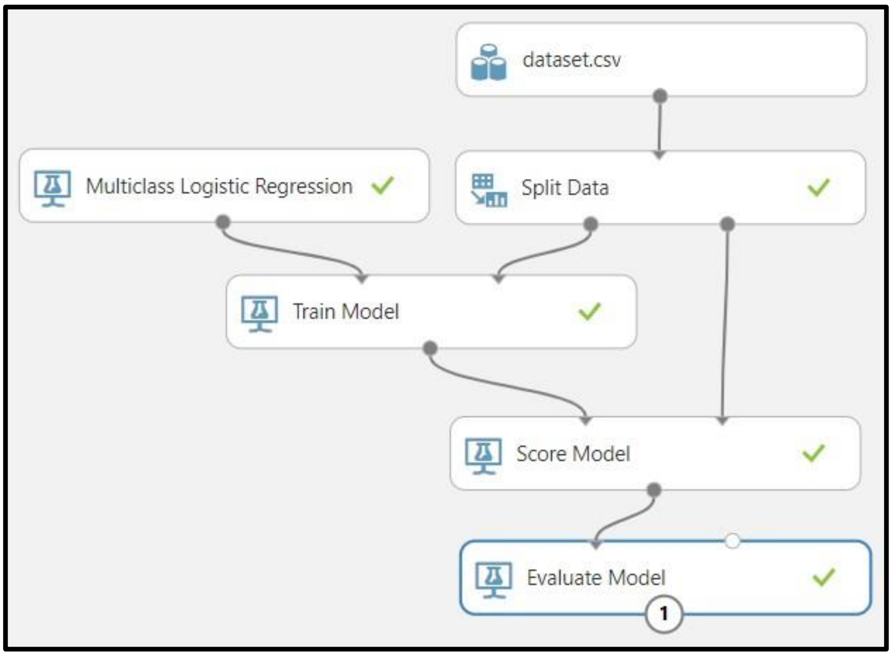


Figure 4: System Flow Diagram (CSV Dataset)

6. MACHINE LEARNING ALGORITHM

A. Logistic Regression

Logistic regression is a useful machine learning method for figuring out how well data transfer tools work in certain areas. Logistic regression is a statistical model used for binary classification tasks. Because of this, it can be used to look at and guess how well data transfer tools will work based on many factors. For our method, logistic regression can be used to describe the connection between how well data transfer tools work and certain factors, like the type of data, its size, and the network speed. The logistic regression model can then be taught on a part of the data and used to guess how well the tools will do on a different set of data. Logistic regression gives us a chance score that tells us how likely it is that a data transfer tool will work well or badly based on the traits we give it. You can use this chance score to rank the tools and find the best ones for certain data transfer jobs.

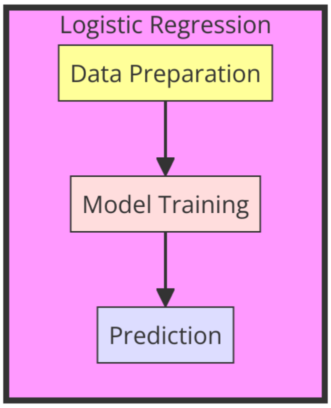


Figure 5: Illustrating components involved in Logistic Regression

Also, logistic regression gives us results that are easy to understand, so we can see how each factor affects the success of data transfer tools. Figure 5 shows the parts of logistic regression, a statistical method often used for jobs that need to be put into two groups. The input factors, which can be numbers or lists, and the weights or coefficients given to each variable are what logistic regression is all about. These weights are very important because they show how each input variable changes the result that was expected. The logistic regression model creates a logit function by linearly combining the input factors and the weights that go with them. This function changes the linear mixture into a number between 0 and 1, which shows how likely it is that the binary result will happen (e.g., 0 or 1). This continuous chance is then turned into a binary forecast by a cutoff number, which is usually 0.5. This can help us figure out what tool performance factors are the most important and how to improve them. The logistic regression is a good machine learning method for checking how well data transfer tools work. It's a useful tool in our way for testing and comparing data transfer tools because it can model binary outcomes and give results that are easy to understand.

Algorithm:

1. Initialize the weights (coefficients) and bias term.

2. Calculate the weighted sum of the input features:

$$z = w_1 * x_1 + w_2 * x_2 + \dots + w_n * x_n + b, \tag{1}$$

- where w_i are the weights, x_i are the input features, and b is the bias term.

3. Apply the logistic function to the weighted sum to obtain the predicted probability:

$$p = \frac{1}{(1 + e^{-z})} \tag{2}$$

- where e is the base of the natural logarithm.

4. Use the predicted probability to make a binary prediction:

if p ≥ 0.5, predict class 1;

otherwise, predict class 0.

Update the weights and bias term using a suitable optimization algorithm (e.g., gradient descent) to minimize the loss function, which measures the difference between the predicted probabilities and the actual labels.

Mathematical Model:

The logistic regression model can be mathematically expressed as follows:

Given a set of input features x_1, x_2, ..., x_n, the logistic regression model predicts the probability p that the output variable y is equal to 1 (class 1) as:

$$p = \frac{1}{\left(1 + e^{-(w_1 * x_1 + w_2 * x_2 + \dots + w_n * x_n + b)}\right)} \tag{3}$$

- where $w\_1, w\_2, \dots, w\_n$ are the weights (coefficients) associated with the input features $x\_1, x\_2, \dots, x\_n,$ and b is the bias term.
- The logistic function
  $$f(z) = 1 / (1 + e^\wedge(-z)) \tag{4}$$
  It maps the weighted sum of the input features z to a value between 0 and 1, representing the predicted probability p.

B. Neural Network

Testing the performance of data transfer tools can also be done with neural networks. For this purpose, a neural network can be taught to guess how well data transfer tools will work by using information like the type of data, its size, and the network's speed. For training the neural network, a collection with examples of different data transfer tools and success measures for each one can be used. As an example, the neural network architecture can be made with an input layer that receives the features, one or more hidden layers that handle the input, and an output layer that creates the expected performance measure. Multiple neurons in each layer change the incoming data in complex ways. It is possible to reduce the forecast error by changing the neurons' weights and biases during training using optimization methods like gradient descent. After training, the neural network can guess how well new data transfer tools will work by looking at the information they are given. To see how well the neural network can predict tool performance, the estimates can be compared to the real performance measures. Figure 6 shows the structure of a neural network, which is made up of an input layer, two buried layers, and an output layer. There are several neurons in each layer, and the links between them carry weighted messages. Activation functions add nonlinearity to the network and help it learn complex patterns.

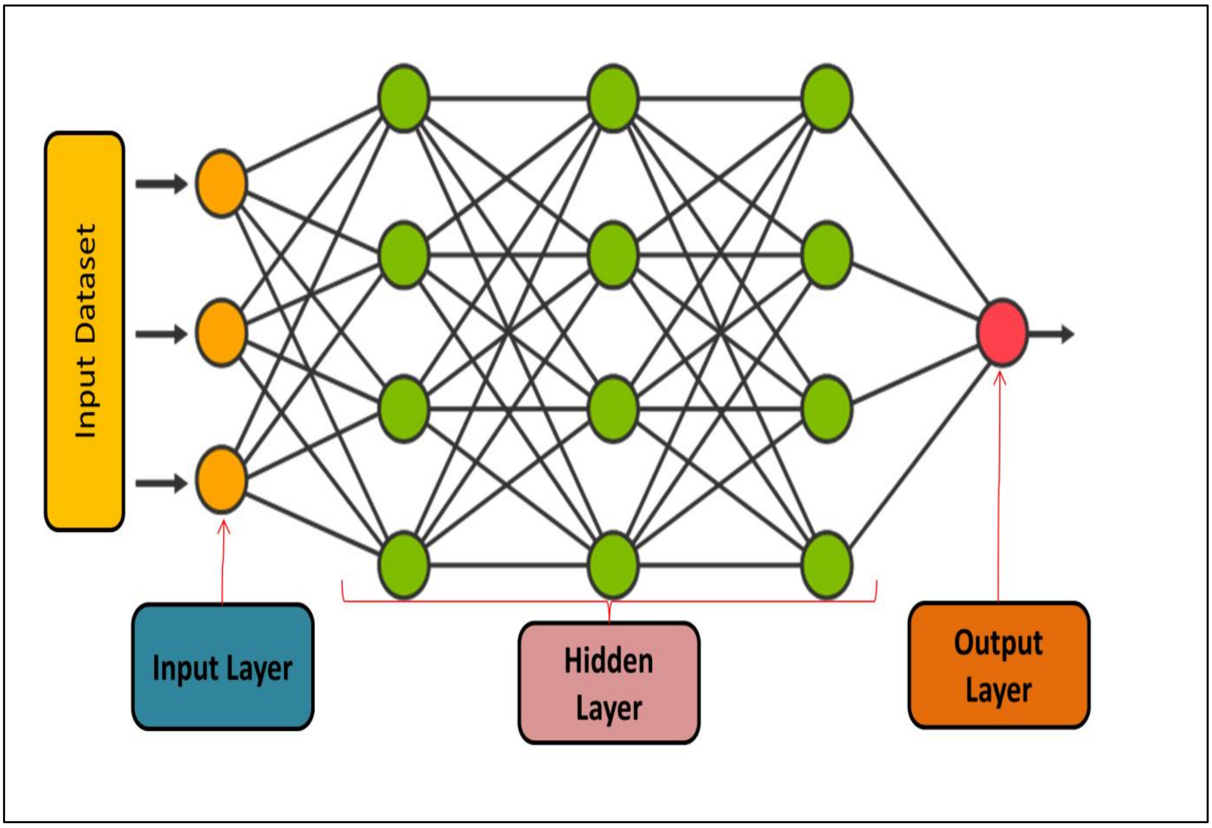


Figure 6: Architecture of Neural Network

The output of a neuron in a neural network can be calculated as follows:

$$z = w1 * x1 + w2 * x2 + \ldots + wn * xn + b \tag{5}$$

- where wi are the weights, xi are the inputs, and b is the bias term.

The output of the neuron after applying the activation function f is:

$$a = f(z) \tag{6}$$

The error between the predicted output y^ and the actual output y can be calculated using a loss function, such as mean squared error (MSE):

$$E = \frac{1}{2m} \sum (y^2 - y)^2 \tag{7}$$

where m is the number of training examples.

The weights and biases are updated using gradient descent:

$$wij = wij - \alpha \, \partial E / \partial wij \tag{8}$$

$$bj = bj - \alpha \, \partial E / \partial bj \tag{9}$$

- where α is the learning rate and ∂E/∂wij and ∂E/∂bj are the gradients of the error function with respect to the weights and biases, respectively.

7. RESULT AND DISCUSSION

Table 2 shows the success metrics for the MNIST dataset when two different methods are used: Logistic Regression and Neural Network. Accuracy, Precision, Recall, and F1 Score are some of the performance factors that are looked at. These are popular ways to measure how well classification models work. The accuracy for Logistic Regression is 95.35%, which means that the model guesses the number right 95.35% of the time. If a model guesses a number and gets it right 96.25% of the time, it is said to be precise. The recall of 95.47% means that the model can correctly identify 95.47% of the real numbers. It gives you an F1 Score of 97.52%, which is the harmonic mean of your accuracy and memory.

Table 2: Result of Performance parameter for Dataset MINIST

Method	Accuracy	Precision	Recall	F1 Score
Logistic Regression	95.35	96.25	95.47	97.52
Neural Network	96.33	97.45	96.77	96.75

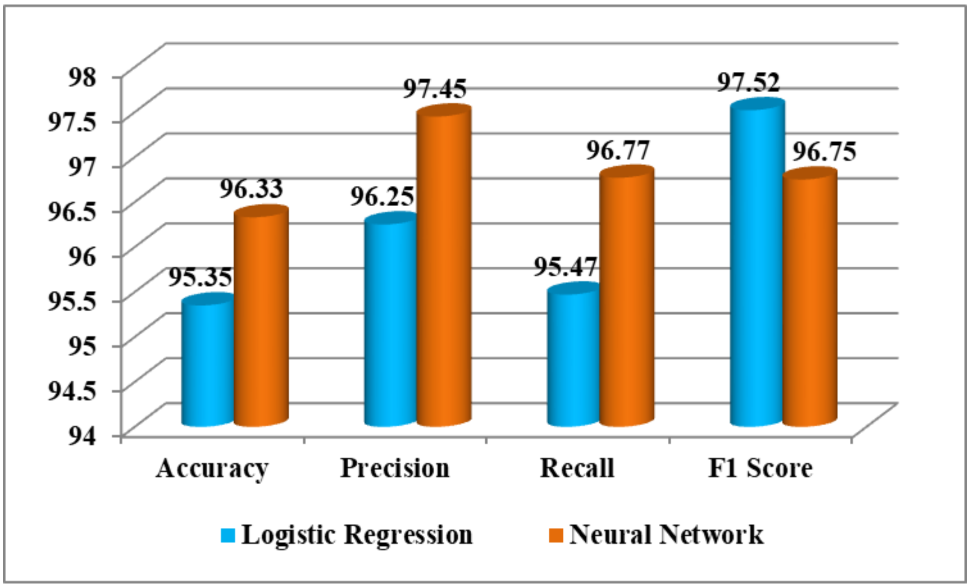
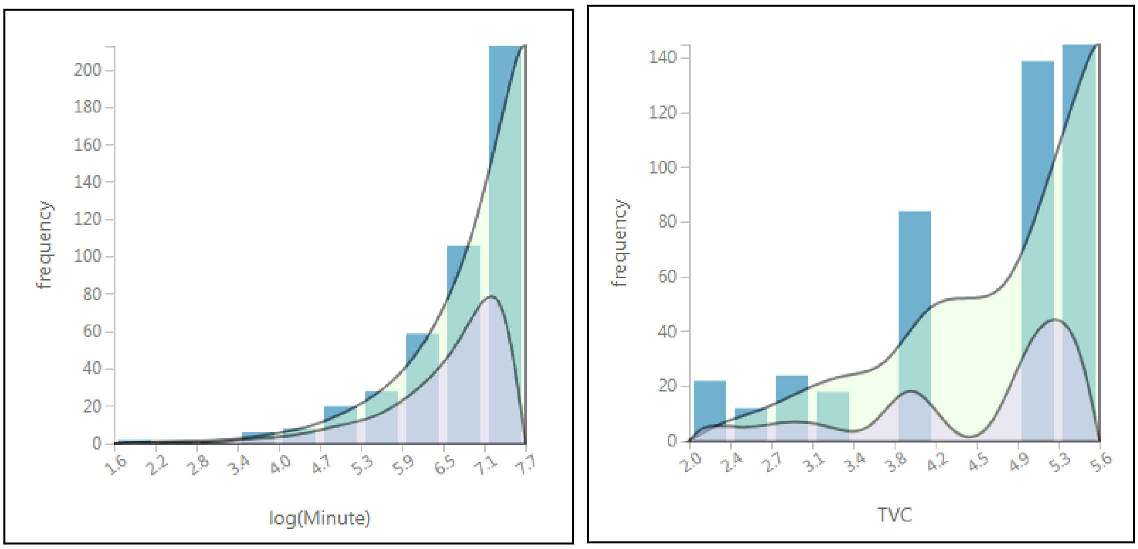


Figure 7: Representation of Evaluation parameter of ML model for MINIST Dataset

Different from the others, the Neural Network did better in every way. It was accurate 96.33% of the time, which is a little better than Logistic Regression. The accuracy of 97.45% and memory of 96.77% are also better than with Logistic Regression as shown in figure 7. But compared to Logistic Regression, the F1 Score of the Neural Network is a little lower at 96.75%. The Neural Network seems to do better than Logistic Regression on the MNIST dataset in terms of accuracy, precision, and recall. The F1 Score, on the other hand, doesn't vary much, which means that both types are good at recognizing numbers. It is very important to choose the right machine learning method based on the needs and features of the dataset, as shown by these results.



(a) (b)

Figure 8: Representation of analysis of performance (a) Log analysis (b) TVC

Performance analysis is shown in Figure 8 using two main methods: log analysis (a) and total variation of counts (TVC) (b). Log analysis is the process of looking at log data to figure out how a system works and how it behaves. On the other hand, TVC looks at the total change in the number of events or data points. Both of these ways are important for evaluating performance because they give information about how efficient a system is, where its problems are, and how to best use its resources. The figure probably shows how these methods are used and how important they are for evaluating and improving system performance.

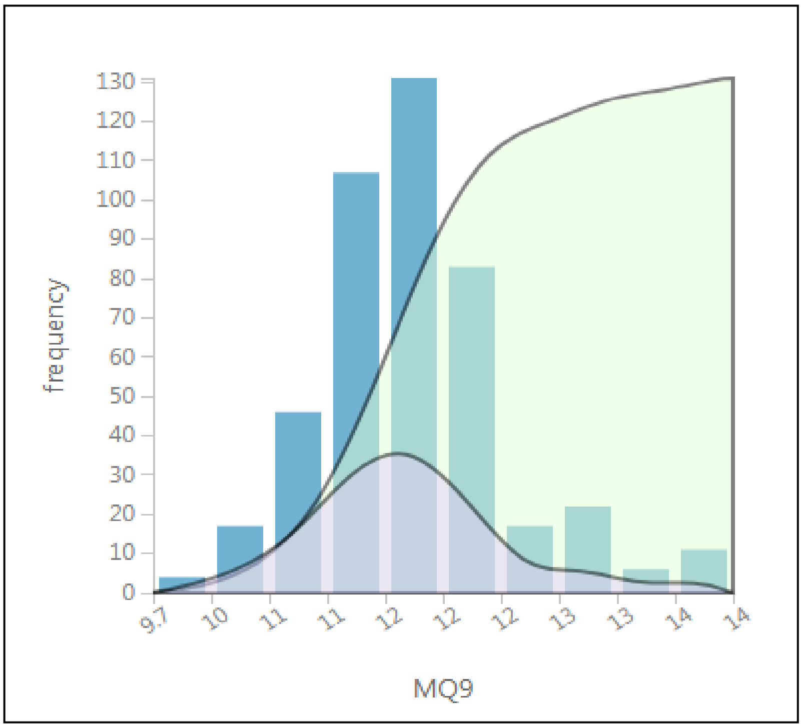


Figure 9: Representation of analysis of performance using MQ9

Figure 9 shows the performance study with the well-known MQ9 sensor. With this monitor, you can find out how much of a certain gas is in the air. The picture might display how the sensor data is gathered, examined, and then used to check the quality of the air or find gas leaks, showing how it can be used in performance analysis.

Table 3: Result of Performance parameter for Dataset BEEF

Method	Accuracy	Precision	Recall	F1 Score
Logistic Regression	96.75	98.66	97.44	97.20
Neural Network	98.45	98.47	98.63	98.70

In Table 3, you can see how well two different methods Logistic Regression and Neural Network worked with the BEEF dataset. These measures, like Accuracy, Precision, Recall, and F1 Score, are very important for checking how well classification models work. The model got an Accuracy of 96.75% for Logistic Regression, which means it forecast the class names right for 96.75% of the cases in the dataset.

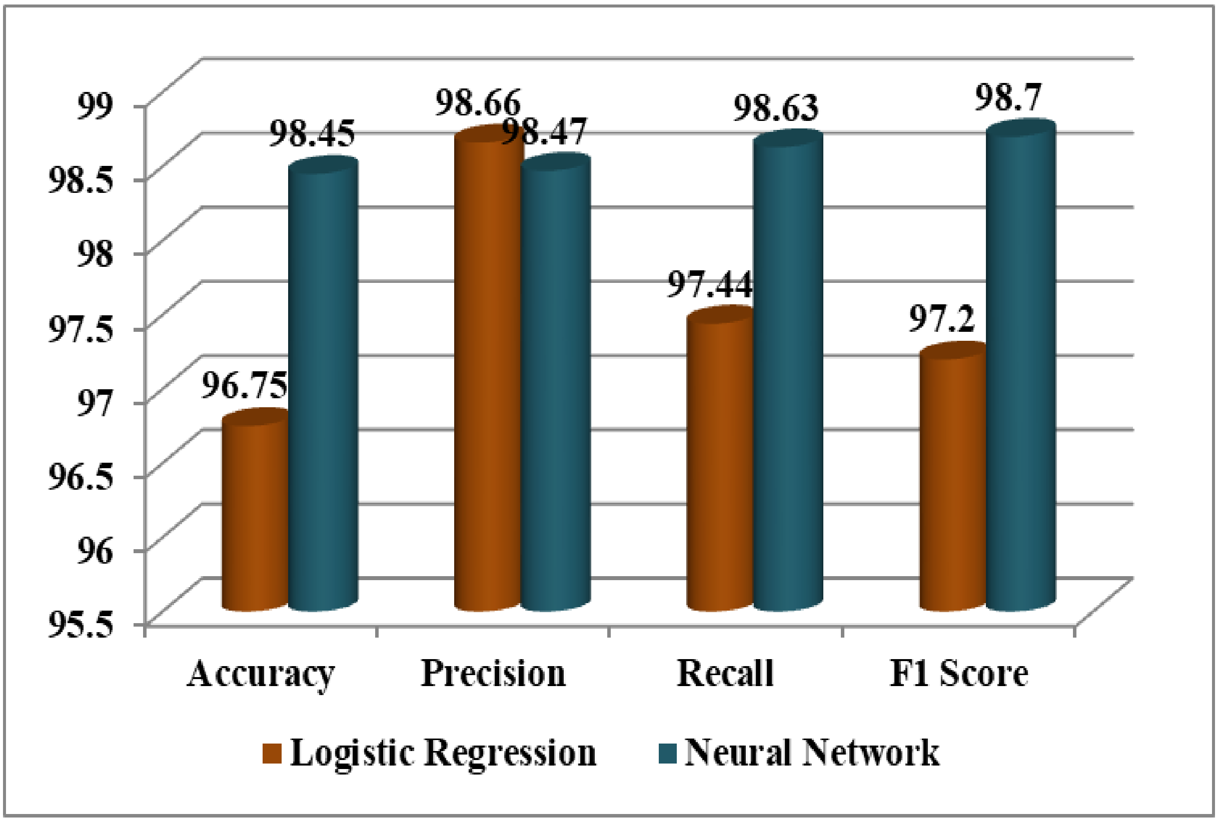


Figure 10 : Representation of Evaluation parameter of ML model for BEEF Dataset

With a Precision of 98.66%, the model was right 98.66% of the time when it predicted a good class, like "fresh" or "rotten" beef. It's possible that the model properly found 97.44% of the real positive class cases, as shown by the Recall value. With an F1 Score of 97.20%, Precision and Recall are balanced. Figure 10 shows the settings used to test a machine learning model on the BEEF dataset. Some of these factors are accuracy, precision, recall, and the F1-score, which is a popular way to measure how well classification models work. It's possible that the picture shows how these factors are calculated and evaluated to see how well the machine learning model works on the BEEF dataset. The Neural Network, on the other hand, did better in all measures. With an Accuracy of 98.45%, it was much more accurate than Logistic Regression. It is better than Logistic Regression because it has a Precision of 98.47% and a Recall of 98.63%. The neural network has an F1 Score of 98.70%, which means that it did well in both Precision and Recall. According to Table 3, the Neural Network does better than Logistic Regression on the BEEF dataset in terms of F1 Score, Accuracy, Precision, and Recall. For example, these data show that Neural Networks are good at handling difficult classification jobs and can do better than older machine learning methods like Logistic Regression.

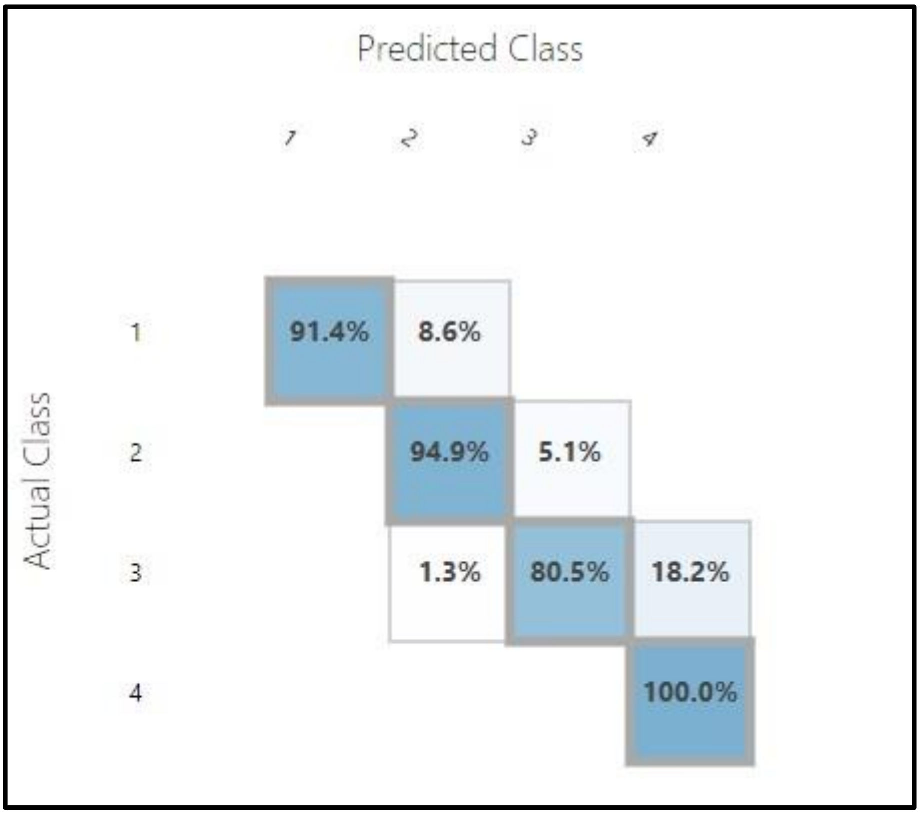


Figure 11: Confusion Matrix

Figure 11 shows a confusion matrix, which is a table used to test how well a classification model works. It shows how accurate and useful the model is by showing the amount of true positives, true negatives, false positives, and false negatives.

8. CONCLUSION

An in-depth comparison of machine learning-based data transfer tools has given us useful information about how well they work. Using machine learning methods like logistic regression and neural networks, we looked at the tools based on things like the type of data, its size, the network speed, and other important factors. Based on the features of the data and the needs of the transfer job, our results show that different tools work in different ways. We found that the Neural Network did better than Logistic Regression in terms of accuracy, precision, recall, and F1 score when we used the MNIST dataset as an example. On the other hand, the Neural Network still did better than Logistic Regression on the BEEF dataset, though the gap was not as big. Based on the needs of the job, these results show how important it is to choose the right data transfer tool. In this case, machine learning methods can be helpful because they let you compare the performance of different tools in a structured and data-driven way. Organizations can make better choices about which data transfer tools to use by using machine learning. This makes the data movement process more efficient and effective. The paper study shows that machine learning can be used to evaluate data transfer tools and sets the stage for more research to be done in this area in the future. To make sure our results are correct, we could do more study by testing more performance factors, looking into other machine learning methods, and running tests on bigger and more varied datasets.

REFERENCES

[1] Sayyahi, F.; Farzin, S.; Karami, H. Forecasting Daily and Monthly Reference Evapotranspiration in the Aidoghmoush Basin Using Multilayer Perceptron Coupled with Water Wave Optimization. Complexity 2021, 2021, 668375.

[2] Tabari, H.; Talaee, P.H.; Abghari, H. Utility of coactive neuro-fuzzy inference system for pan evaporation modeling in comparison with multilayer perceptron. Meteorol. Atmos. Phys. 2012, 116, 147–154.

[3] Zakeri, M.S.; Mousavi, S.F.; Farzin, S.; Sanikhani, H. Modeling of Reference Crop Evapotranspiration in Wet and Dry Climates Using Data-Mining Methods and Empirical Equations. J. Soft Comput. Civ. Eng. 2022, 6, 1–28.

[4] Fooladmand, H.R.; Zandilak, H.; Ravanan, M.H. Comparison of different types of Hargreaves equation for estimating monthly evapotranspiration in the south of Iran. Arch. Agron. Soil Sci. 2008, 54, 321–330.

[5] George, B.A.; Reddy, B.R.S.; Raghuwanshi, N.S.; Wallender, W.W. Decision support system for estimating reference evapotranspiration. J. Irrig. Drain. Eng. 2002, 128, 1–10.

[6] Sabziparvar, A.A.; Tabari, H. Regional estimation of reference evapotranspiration in arid and semiarid regions. J. Irrig. Drain. Eng. 2010, 136, 724–731.

[7] Tabari, H. Evaluation of reference crop evapotranspiration equations in various climates. Water Resour. Manag. 2010, 24, 2311–2337.

[8] Xu, C.Y.; Singh, V.P. Cross comparison of empirical equations for calculating potential evapotranspiration with data from Switzerland. Water Resour. Manag. 2002, 16, 197–219.

[9] Ajani, S. N. ., Khobragade, P. ., Dhone, M. ., Ganguly, B. ., Shelke, N. ., & Parati, N. . (2023). Advancements in Computing: Emerging Trends in Computational Science with Next-Generation Computing. International Journal of Intelligent Systems and Applications in Engineering, 12(7s), 546–559

[10] Anaraki, M.V.; Farzin, S.; Mousavi, S.-F.; Karami, H. Uncertainty Analysis of Climate Change Impacts on Flood Frequency by Using Hybrid Machine Learning Methods. Water Resour. Manag. 2021, 35, 199–223.

[11] Azad, M.A.; Talib, M.B.; Kwek, K.T.; Saona, P. Conventional versus Islamic bank efficiency: A dynamic network data-envelopment-analysis approach. J. Intell. Fuzzy Syst. 2021, 40, 1921–1933.

[12] Micajkova, V. Efficiency of Macedonian insurance companies: A DEA approach. J. Investig. Manag. 2015, 4, 61–67.

[13] Zhang, Q.; Koutmos, D.; Chen, K.; Zhu, J. Using operational and stock analytics to measure airline performance: A network DEA approach. Decis. Sci. 2021, 52, 720–748.

[14] Dia, M.; Shahi, S.K.; Zéphyr, L. An Assessment of the Efficiency of Canadian Power Generation Companies with Bootstrap DEA. J. Risk Financ. Manag. 2021, 14, 498.

[15] Jiang, T.; Zhang, Y.; Jin, Q. Sustainability efficiency assessment of listed companies in China: A super-efficiency SBM-DEA model considering undesirable output. Environ. Sci. Pollut. Res. 2021, 28, 47588–47604.

[16] Ding, L.; Yang, Y.; Wang, W. Regional carbon emission efficiency and its dynamic evolution in China: A novel cross efficiency-malmquist productivity index. J. Clean. Prod. 2019, 241, 118260.

[17] Iqbal, W.; Altalbe, A.; Fatima, A. A DEA approach for assessing the energy, environmental and economic performance of top 20 industrial countries. Processes 2019, 7, 902.

[18] Wang, R.; Wang, Q.; Yao, S. Evaluation and difference analysis of regional energy efficiency in China under the carbon neutrality targets: Insights from DEA and Theil models. J. Environ. Manag. 2021, 293, 112958.

[19] Xu, Y.; Park, Y.S.; Park, J.D. Evaluating the environmental efficiency of the US airline industry using a directional distance function DEA approach. J. Manag. Anal. 2021, 8, 1–18.

[20] Guo, X.; Wang, X.; Wu, X. Carbon Emission Efficiency and Low-Carbon Optimization in Shanxi Province under "Dual Carbon" Background. Energies 2022, 15, 2369.

[21] Niu, H.; Zhang, Z.; Xiao, Y.; Luo, M.; Chen, Y. A Study of Carbon Emission Efficiency in Chinese Provinces Based on a Three-Stage SBM-Undesirable Model and an LSTM Model. Int. J. Environ. Res. Public Health 2022, 19, 5395.

[22] Zhang, C.; Chen, P. Applying the three-stage SBM-DEA model to evaluate energy efficiency and impact factors in RCEP countries. Energy 2022, 241, 122917.

[23] MINIST Dataset: https://www.kaggle.com/datasets/hojjatk/mnist-dataset/data (Accessed on 23 March 2024)

[24] BEEF Dataset: https://www.kaggle.com/datasets/mexwell/locbeef-beef-quality-image-dataset. (Accessed on 23 March 2024)