A Web-Based User Interface for Machine Learning Analysis

Fatma Nasoz[1](✉) and Chandani Shrestha[2]

[1] University of Nevada-Las Vegas, Las Vegas, NV, USA
fatma.nasoz@unlv.edu
[2] Microsoft, Seattle, WA, USA
chshrest@microsoft.com

Abstract. The purpose of this study is to develop a user-friendly web application that follows human computer interaction design guidelines and principles and is used to recognize patterns in datasets and to predict outputs of instances that it hasn't previously encountered. The application design follows human computer interaction design guidelines and principles and it employs supervised machine learning algorithms linear regression, logistic regression, and backpropagation for prediction. Java is used in the backend to create a model that maps the input and output data based on any of the machine learning algorithms while Play Framework and Bootstrap are used to display content in frontend. The application allows users to upload datasets that will be used to train and test the system. Each column of an uploaded dataset represents an attribute and each row represents an instance. The system is also developer friendly and allows changes be made to the source code for a more customized interaction.

Keywords: Machine learning · Classification · Regression · Web-based graphical user interface

1 Introduction

Machine learning systems are used to solve a variety of learning tasks [11]. For any given application, the main goal of machine learning is to build a model that represents and generalizes the training examples [7] and the performance of the model is measured by how well it generalizes when tested on new data.

In this study, we develop a web-based graphical user interface that allows its users to utilize machine learning algorithms to solve regression and classification problems. The supervised machine learning algorithms used to build the predictive models are linear regression [13], logistic regression [1], backpropagation [14].

This application can be used for classification problems like predicting whether a tumor with certain attributes is benign or malignant. In supervised machine learning techniques, each data instance used for training must have both input and output values, meaning all instances which are being used to train the machine learning model for breast tumor data have a set of features describing the tumor and an attribute stating if the tumor is cancerous or not.

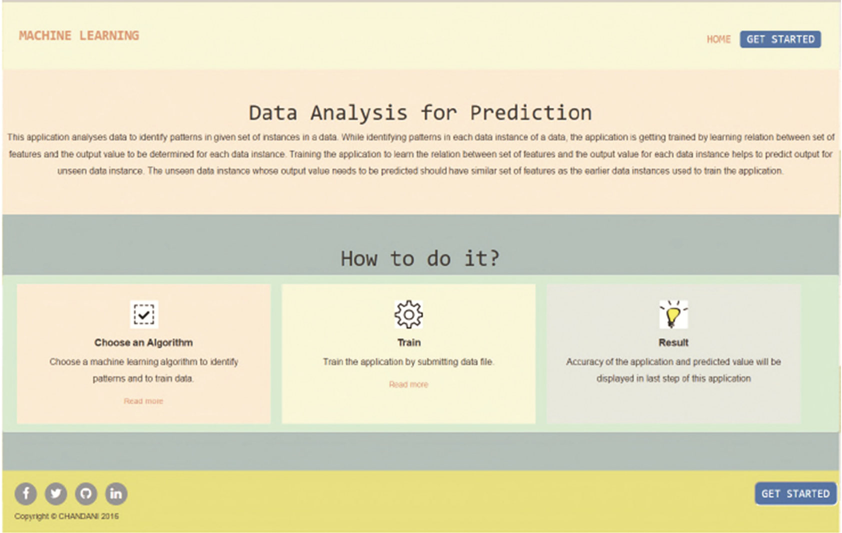
S. Yamamoto (Ed.): HIMI 2017, Part II, LNCS 10274, pp. 444–453, 2017.
DOI: 10.1007/978-3-319-58524-6_35

As for the interface, in the backend, Java is used to create a model that maps the input and output data based on any of the machine learning algorithms. In the frontend, Play Framework and Bootstrap are used to display content. Play Framework is chosen because it is based on web-friendly architecture. As a result, it uses predictable and minimal resources (CPU, memory, threads) for highly scalable applications. It is also developer friendly where changes can be made in the code and hitting the refresh button in browser will update the interface. Bootstrap is used to style the web application and it adds responsiveness to the interface with added feature of cross-browser compatible designs. Thus, the website has a responsive design and suitable for any device.

The interface design is guided by the "Eight Golden Rules of Interface Design" for improved universal usability [16]. The interface is more focused towards novice and intermediate users. Following are the list of implemented design rules:

- ***Consistency:*** Font size, color of text, button, input box size, alignment of content, background color, margin and organization of content is consistent. Similar sequence of actions is used to predict outcome in all machine learning algorithms. The flow of entering inputs and moving from one screen to the next page is also consistent throughout the application. Headers and footers are also consistent throughout.
- ***Simple layout:*** Contents in the application are short and descriptive so that users don't skip or get annoyed with lengthy information. A design is simple more appealing and loads faster as web application should give access to the information not abundance of information. Simple well organized display is used where left section of the interface shows stage of the application and right side shows information related to current stage of the application. Cluttered display of interface elements is also avoided.
- ***Reduce short term memory load:*** Users don't have to memorize steps to reach "Result" screen, which displays the predicted value. The progress towards prediction is displayed on the left side of each interface inside "Steps" section. Steps are listed in ascending order in which they will be executed. It informs users the step they are currently in, the previous step they completed, as well as the upcoming step.
- ***Error prevention and error checking:*** Each input box has a placeholder which gives user an idea of what kind of information needs to be entered there. Additionally, error checking is done to prevent users from submitting invalid information or missing required information.
- ***Compatibility of data entry and data display:*** The format of data entered as input to the application is compatible with the format of data displayed as output. On the "Result" screen the format of predicted value is similar to the format of value in the column representing dependent variable (output column) of data file.

Figure 1 shows the home screen of the application, which provides information on the functionality of the system. To see the available machine learning analysis options the users need to click the "Get Started" button and Fig. 2 displays the steps and available options.



Data Analysis for Prediction

This application analyses data to identify patterns in given set of instances in a data. While identifying patterns in each data instance of a data, the application is getting trained by learning relation between set of features and the output value to be determined for each data instance. Training the application to learn the relation between set of features and the output value for each data instance helps to predict output for unseen data instance. The unseen data instance whose output value needs to be predicted should have similar set of features as the earlier data instances used to train the application.

How to do it?

Choose an Algorithm

Choose a machine learning algorithm to identify patterns and to train data.

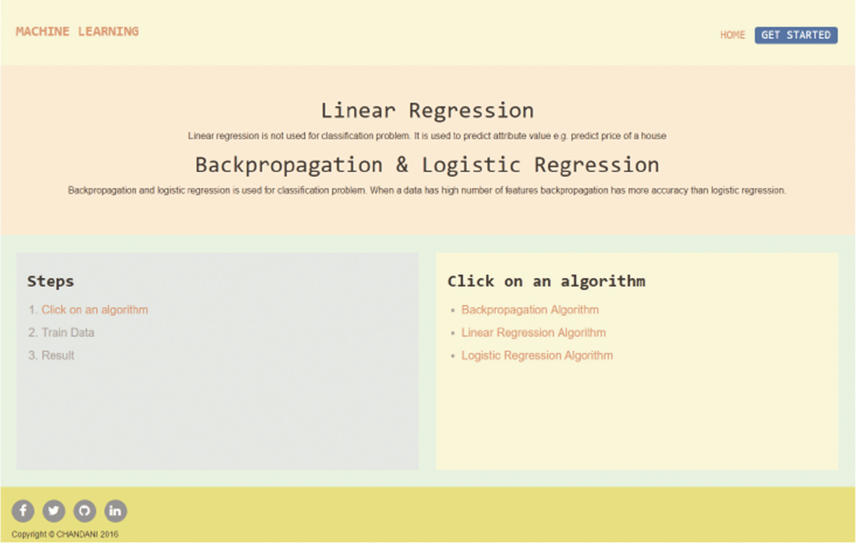
Train

Train the application by submitting data file.

Result

Accuracy of the application and predicted value will be displayed in last step of this application

Fig. 1. Home screen



Linear Regression

Linear regression is not used for classification problem. It is used to predict attribute value e.g. predict price of a house

Backpropagation & Logistic Regression

Backpropagation and logistic regression is used for classification problem. When a data has high number of features backpropagation has more accuracy than logistic regression.

Steps

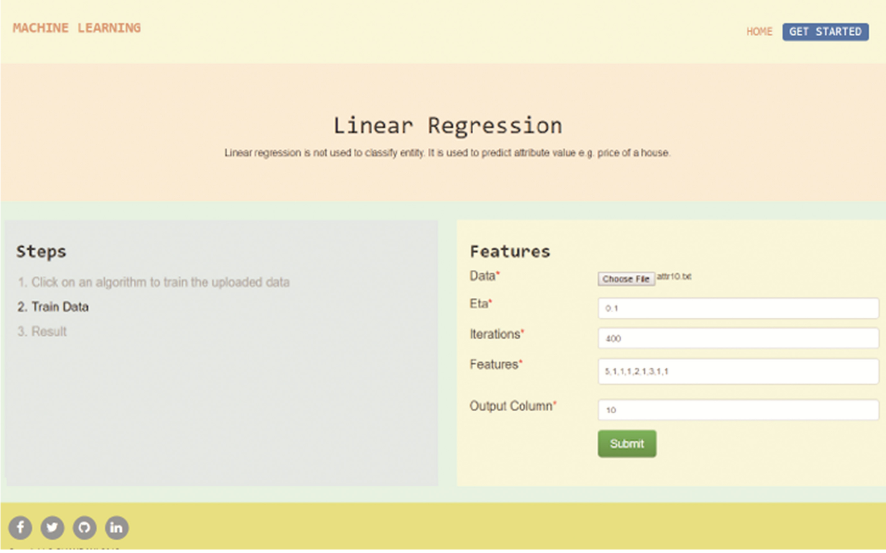
1. Click on an algorithm
2. Train Data
3. Result

Click on an algorithm

- Backpropagation Algorithm
- Linear Regression Algorithm
- Logistic Regression Algorithm

Fig. 2. Content displayed after clicking the "Get Started" button

Figure 3 displays the interface screen during data analysis with one of the algorithms. This screen allows the user to input the parameters for the chosen machine learning algorithm.



Linear Regression

Linear regression is not used to classify entity. It is used to predict attribute value e.g. price of a house.

Steps

1. Click on an algorithm to train the uploaded data
2. Train Data
3. Result

Features

Data*	attr10.txt
Eta*	0.1
Iterations*	400
Features*	5,1,1,1,2,1,3,1,1
Output Column*	10

Fig. 3. Analyzing the data

2 Motivation

There are massive collections of data related to health [2, 10], business [3], education [2], etc. Each of these datasets has some patterns that can reveal cause of certain disease [15], cause of business failure [3], relation between customers and products [2, 3, 17], relationship between education and career opportunities [2], etc.

Machine learning helps to detect patterns and regularities and make good approximations or predictions. It is used in data analysis of retail, finance, credit application, fraud detection, stock market, medical data [2]. It is also used to process large datasets where it is not obvious how information is interrelated [6]. Similarly, neural networks are used to analyze imprecise, fuzzy, imperfect knowledge where there is a lack of mathematical algorithm to perform the analysis [8].

In order to serve the individual needs of individuals and organizations, there is a need to extract information from this huge volume of data sets in digital world [9]. There are many research studies conducted in machine learning and advanced desktop applications like MATLAB and SASS have been developed to detect patterns and predict outcome; however not all users have are trained in using the advanced methods and applications. The main of this study is to design and develop a web-based application that follows human computer interaction design guidelines and principles [16] and can learn patterns in the data and utilize these patterns to make prediction.

3 Machine Learning

Machine learning is a sub-discipline of artificial intelligence and uses theory of statistics to build a mathematical model [2]. The developed model is capable of learning from complex large data sets or past experiences [2, 11]. The model can be used for prediction or to visualize pattern in the data and understand information hidden in complex data.

Algorithms in machine learning focus on optimizing the model, which means adjusting the weight parameters of the model to best fit the data. The optimized model has more accuracy and has low time and space complexity [2, 15].

Solving a specific machine learning task, supervised learning algorithms build a model from label datasets in which each data instance is a pair of input and its corresponding output. The goal of supervised learning is to determine the parameters of the function *f(x)* that best fits those input-output pairs. The dataset that is used to fit the model is called a training set.

Function *f(x)* is then used to predict the output for data instances that were not seen before. In an optimized cost function, the difference between output from *f(x)* and target output is minimum, which increases the accuracy of the system [12]. Supervised learning is used to solve both classification (i.e., output values are categorical) and regression (i.e., output values are continuous) problems. Three supervised learning algorithms (linear regression, logistic regression, and backpropagation) were implemented to be used with the interface. The models were optimized with gradient descent [11].

4 Results

The system application is tested using two different publicly available breast cancer data sets found at University of California, Irvine Machine Learning Repository.

4.1 Dataset with Nine Attributes

The properties of the first data set [4] are:

- Number of Instances: 699
- Number of Attributes: 9 (input features with values ranging 1–10: Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal Nucleoli Mitosis)
- Missing values: 16
- Class distribution: 458 Benign and 241 Malignant

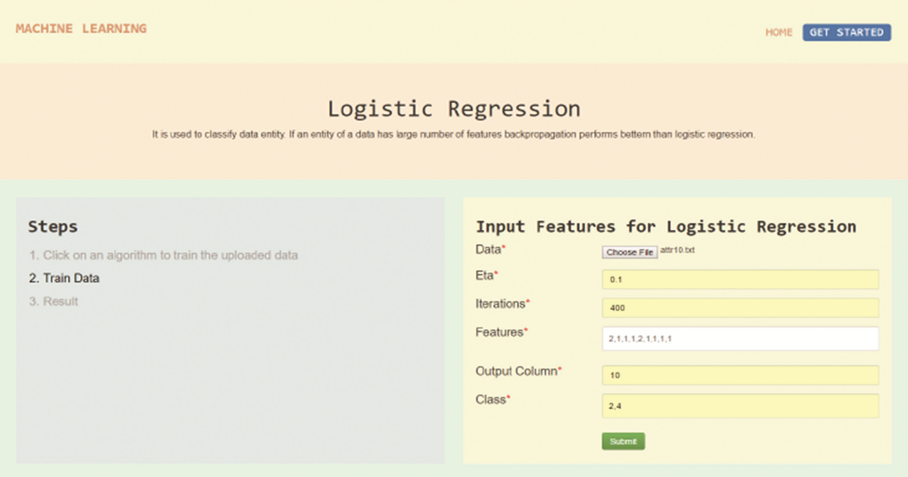
The data set is divided into three groups using k-fold method

- Training data is 80% of total data
- Cross validation is 10% of total data
- Test data is 10% of total data

The models are tested on the test data with 70 instances (10% of the dataset) that have not been used to train the model. Table 1 shows the number of correctly predicted instances in both benign and malignant classes with logistic regression, which corresponds to an overall accuracy of 98.6%. Figures 4 and 5 show the interface screens during logistic regression training and prediction, respectively.

Table 1. Prediction accuracy with logistic regression on breast cancer data with 9 attributes

Class	Correctly predicted instances	Total instances
Benign	55	56
Malignant	14	14



Logistic Regression

It is used to classify data entity If an entity of a data has large number of features backpropagation performs bettern than logistic regression.

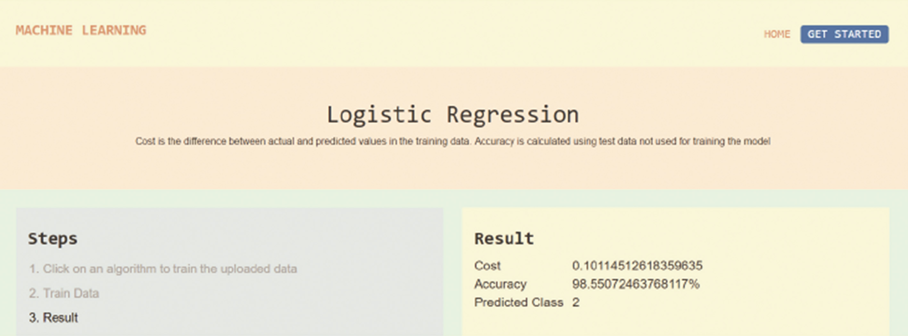
Steps

1. Click on an algorithm to train the uploaded data
2. Train Data
3. Result

Input Features for Logistic Regression

Data* attr10.txt
Eta* 0.1
Iterations* 400
Features* 2,1,1,1,2,1,1,1,1
Output Column* 10
Class* 2,4

Fig. 4. Training with logistic regression



Logistic Regression

Cost is the difference between actual and predicted values in the training data. Accuracy is calculated using test data not used for training the model

Steps

1. Click on an algorithm to train the uploaded data
2. Train Data
3. Result

Result

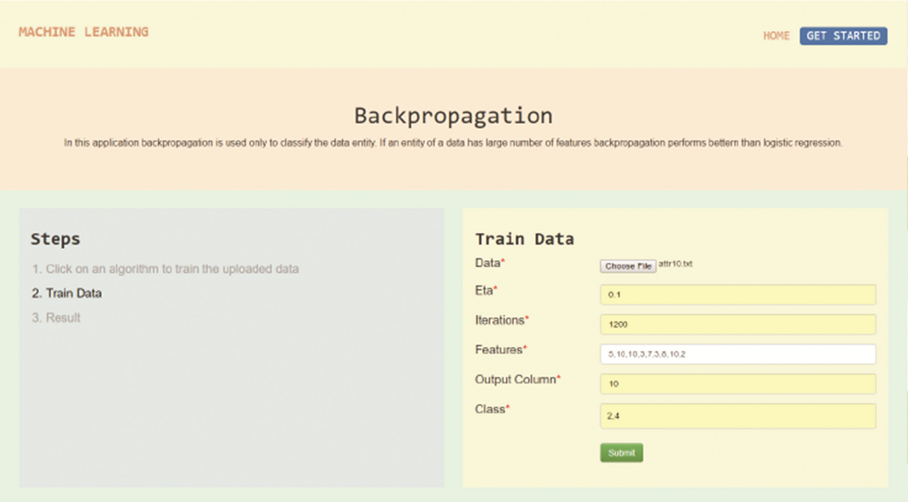
Cost 0.10114512618359635
Accuracy 98.55072463768117%
Predicted Class 2

Fig. 5. Prediction with logistic regression

Table 2 shows the number of correctly predicted instances in both benign and malignant classes with backpropagation, which corresponds to an overall accuracy of 98.6%. Figures 6 and 7 show the interface screens during backpropagation training and prediction, respectively.

Table 2. Prediction accuracy with backpropagation on breast cancer data with 9 attributes

Class	Correctly predicted instances	Total instances
Benign	55	56
Malignant	14	14



Backpropagation

In this application backpropagation is used only to classify the data entity. If an entity of a data has large number of features backpropagation performs bettern than logistic regression.

Steps

1. Click on an algorithm to train the uploaded data
2. Train Data
3. Result

Train Data

Data* attr10.txt

Eta* 0.1

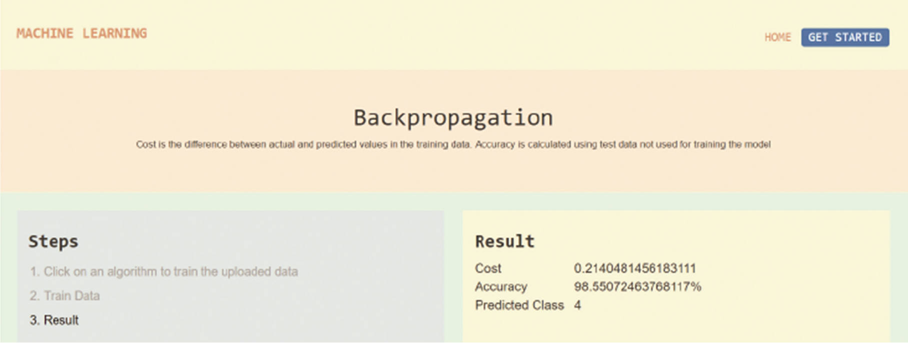
Iterations* 1200

Features* 5,10,10,3,7,3,8,10,2

Output Column* 10

Class* 2,4

Fig. 6. Training with backpropagation



Backpropagation

Cost is the difference between actual and predicted values in the training data. Accuracy is calculated using test data not used for training the model

Steps

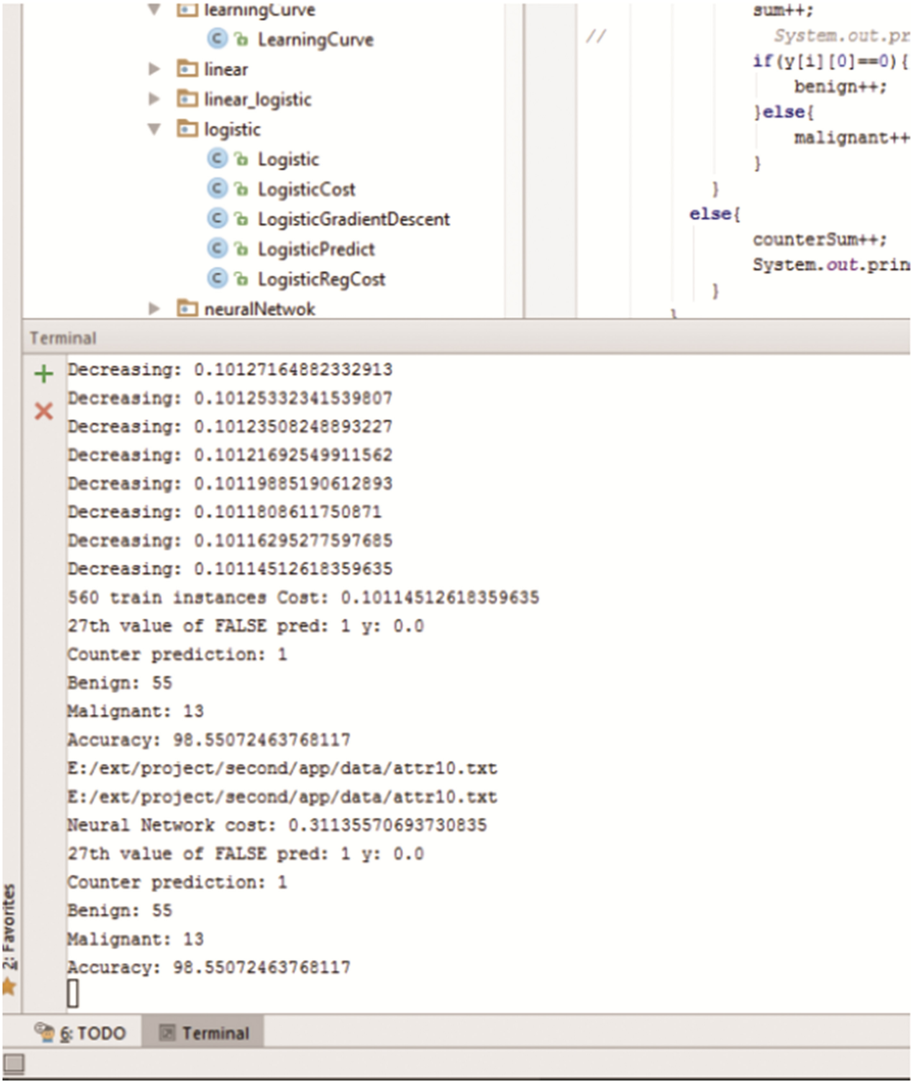
1. Click on an algorithm to train the uploaded data
2. Train Data
3. Result

Result

Cost 0.2140481456183111
Accuracy 98.55072463768117%
Predicted Class 4

Fig. 7. Prediction with backpropagation

Users can also view the console section of the backend of the system when it is being trained. The console displays the decreasing value of cost in each iteration of gradient descent, as well as the number of correctly predicted benign and malignant data instances, and overall accuracy. Figure 8 shows the console during back analysis of breast cancer data with 9 attributes.



```
//              sum++;
                  System.out.pr
                if(y[i][0]==0){
                    benign++;
                }else{
                    malignant++
                }
            }
        else{
                counterSum++;
                System.out.prin
            }
```

```
Decreasing: 0.10127164882332913
Decreasing: 0.10125332341539807
Decreasing: 0.10123508248893227
Decreasing: 0.10121692549911562
Decreasing: 0.10119885190612893
Decreasing: 0.1011808611750871
Decreasing: 0.10116295277597685
Decreasing: 0.10114512618359635
560 train instances Cost: 0.10114512618359635
27th value of FALSE pred: 1 y: 0.0
Counter prediction: 1
Benign: 55
Malignant: 13
Accuracy: 98.55072463768117
E:/ext/project/second/app/data/attr10.txt
E:/ext/project/second/app/data/attr10.txt
Neural Network cost: 0.31135570693730835
27th value of FALSE pred: 1 y: 0.0
Counter prediction: 1
Benign: 55
Malignant: 13
Accuracy: 98.55072463768117
```

Fig. 8. Console view during training with backpropagation on breast cancer data with 9 attributes

4.2 Dataset with Thirty-One Attributes

The properties of the second data set [5] are:

- Number of instances: 569
- Number of attributes: 31
- 30 real value-valued input features: radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, fractal dimension, etc.
- Missing values: none

- Class distribution: 357 benign, 212 malignant

The models are tested on the test data with 57 instances (10% of the dataset) that have not been used to train the model. Tables 3 and 4 show the number of correctly predicted instances in both benign and malignant classes with logistic regression and backpropagation, which corresponds to overall accuracies of 91.2% and 75.4%, respectively.

Table 3. Prediction accuracy with logistic regression on breast cancer data with 31 attributes

Class	Correctly predicted instances	Total instances
Benign	39	44
Malignant	13	13

Table 4. Prediction accuracy with backpropagation on breast cancer data with 31 attributes

Class	Correctly predicted instances	Total instances
Benign	30	44
Malignant	13	13

References

1. Agresti, A.: Categorical Data Analysis, 3rd edn. John Wiley & Sons, Hoboken (2012)
2. Alpaydin, A.: Introduction to Machine Learning, 3rd edn. The MIT Press, Cambridge (2014)
3. Bose, I., Mahapatra, R.K.: Business data mining: a machine learning perspective. Inf. Manag. **39**(3), 211–225 (2001)
4. Breast Cancer Wisconsin (Diagnostic) Data Set. https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Original)
5. Breast Cancer Wisconsin (Original) Data Set. https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
6. Chester, M.: Neural Networks: A Tutorial. Prentice-Hall, Inc., Upper Saddle River (1993)
7. Domingos, P.: A few useful things to know about machine learning. Commun. ACM **55**(10), 78–87 (2012)
8. Eberhart, R.C., Robbins, R.W. (eds.): Neural Network PC Tools: A Practical Guide. Academic Press, Cambridge (1990)
9. Kausnal, C., Arya, A., Pathania, S.: Integration of data mining in cloud computing. Adv. Comput. Sci. Inf. Technol. (ACSIT) **2**(7), 48–52 (2015)
10. Magoulas, G.D., Prentza, A.: Machine learning in medical applications. In: Paliouras, G., Karkaletsis, V., Spyropoulos, C.D. (eds.) ACAI 1999. LNCS (LNAI), vol. 2049, pp. 300–307. Springer, Heidelberg (2001). doi:10.1007/3-540-44673-7_19
11. Mitchell, T.M.: Machine Learning. McGraw-Hill, New York (1997)
12. Reed, R.D., Marks, R.J.: Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks. The MIT Press, Cambridge (1998)
13. Rencher, A.C., Christensen, W.F.: Methods of Multivariate Analysis, 3rd edn. John Wiley & Sons, Hoboken (2012)
14. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**(6088), 533–536 (1986)

15. Sajda, P.: Machine learning for detection and diagnosis of disease. Annu. Rev. Biomed. Eng. **8**, 537–565 (2006)
16. Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., Diakopoulos, N.: Designing the User Interface: Strategies for Effective Human-Computer Interaction, 6th edn. Pearson, Hoboken (2016)
17. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 3rd edn. Morgan Kaufmann, Cambridge (2016)