ME674-Soft Computing Coding Assignment- Detect a Happiness score Project Report

Name : Mukund Jha

Roll No. : 224103421

Dept: Mtech Machine Design

INDEX

INTRODUCTION

Problem Definition

In today's society, happiness has attracted more and more attentions from researchers. It is interesting to study happiness from the perspective of data mining. In psychology domain, the application of data mining gradually becomes widespread and popular, which works from a novel data-driven viewpoint. Where standard economics relies on measurements of income and consumption or other observed behaviors to demonstrate the immeasurable concept of utility, or the satisfaction of material wants and needs, happiness economics uses surveys and related methods to elicit people to directly reveal their level of satisfaction. Happiness economics apples econometric analysis to discover which factors might increase or decrease human well-being and quality of life.

Happiness economics is an attempt to overcome certain shortcomings of traditional approach by trying to measure utility, or happiness, more directly. One major shortcoming of traditional utility theory is that because it relies on observed market prices, quantities, and incomes, it cannot account for the enjoyment that people receive from goods, services, activities, or amenities that occur outside of markets

Problem Formulation

To overcome the above high algebraic complexity problem, a method is proposed in which multilayer perceptron (MLP) is used to predict Happiness score from the following data.The parameters used for Happiness prediction are Social support,healthy life, freedom to make life choice ,generosity , perception of corruption,Gdp per capiata.etc

These are input data for prediction Happiness score.

Research Paper Modelled

Title: Utilization of multilayer perceptron for determining the Happiness score .https://www.researchgate.net/publication/330046774_Application_of_Artificial_Neural_Networks_to_Assess_Student_Happiness

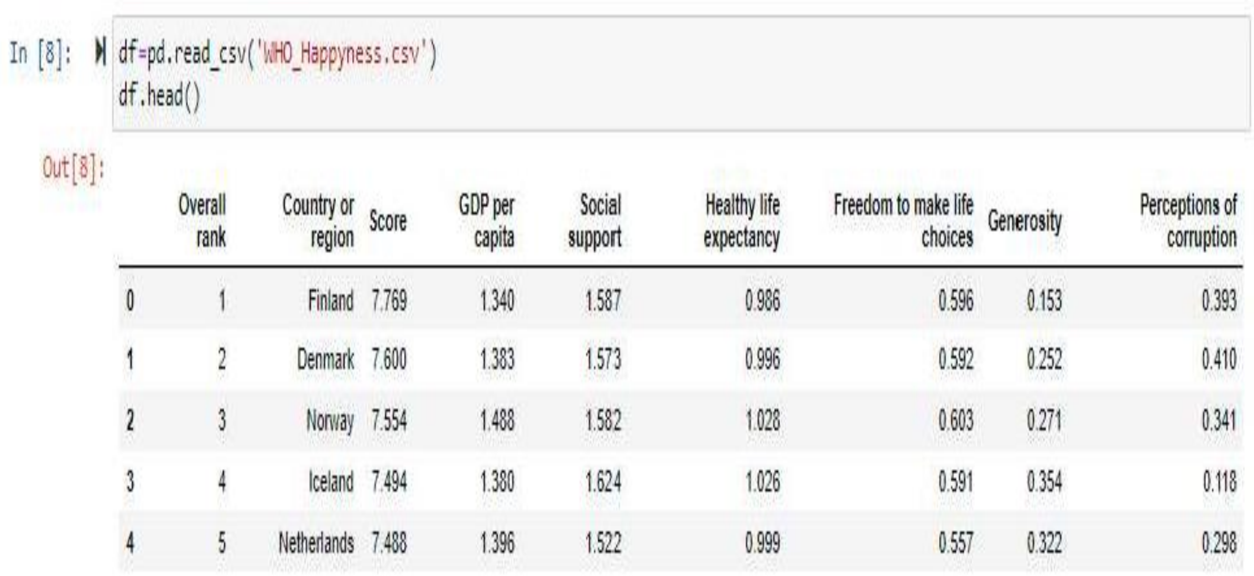
School of Computer Science and Technology, Wuhan University of Technology, "China An Empirical Study of Learning Based Happiness Prediction Approaches". Received 30 April 2021 Accepted 20 June 2021

Dataset Description

The dataset titled ‘**Happiness score** ’ taken from https://www.kaggle.com/ . The data is generated by uniformly randomly generating values of input data there have following input columns.Social support,healthy life, freedom to make life choice ,generosity , perception of corruption,Gdp per capiata.

And one of the column in data_set is Happiness score value which indicate depends the above 6 columns.and another column is rank which is inverse proportion to score means score is more the rank is low in number and give this rank to all the new column country region ;

The top 5 rows of dataset as shown below:



```
In [8]:  df=pd.read_csv('WHO_Happyness.csv')
          df.head()
```

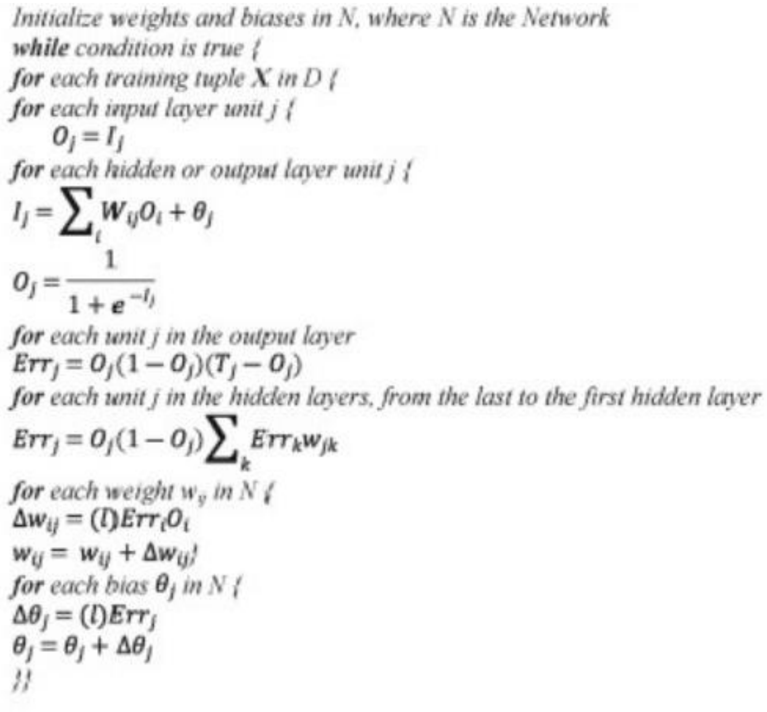
Out[8]:

	Overall rank	Country or region	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption
0	1	Finland	7.769	1.340	1.587	0.986	0.596	0.153	0.393
1	2	Denmark	7.600	1.383	1.573	0.996	0.592	0.252	0.410
2	3	Norway	7.554	1.488	1.582	1.028	0.603	0.271	0.341
3	4	Iceland	7.494	1.380	1.624	1.026	0.591	0.354	0.118
4	5	Netherlands	7.488	1.396	1.522	0.999	0.557	0.322	0.298

METHODOLOGY

Dataset Generation

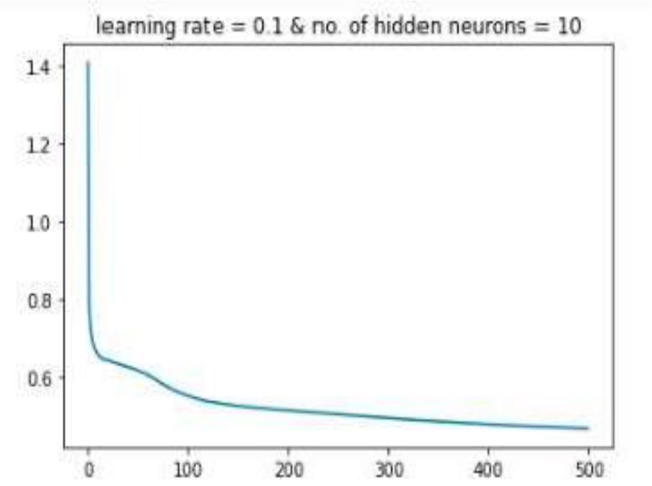
Artificial neurons are used in multiple layers including hidden layers in the multilayer perceptron algorithm. These algorithms are used for binary classification problems. A perceptron uses an activation function for each neuron. Multilayer perceptrons are algorithms evolved from biological neurons. They use artificial neurons that are called perceptrons. The activation function maps the weighted inputs of each neuron and reduces the number of layers to two layers. A perceptron learns by varying the weights assigned to it. The algorithm for a multilayer perceptron is shown in

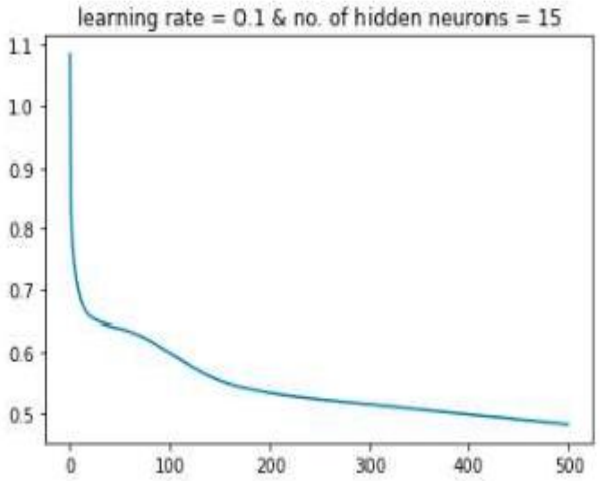


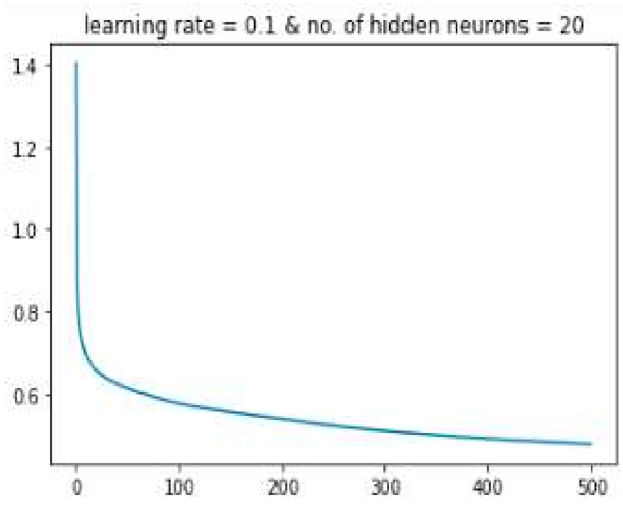
```
Initialize weights and biases in N, where N is the Network
while condition is true {
for each training tuple X in D {
for each input layer unit j {
    $O_j = I_j$
for each hidden or output layer unit j {
$I_j = \sum_i W_{ij} O_i + \theta_j$
$O_j = \frac{1}{1 + e^{-I_j}}$
for each unit j in the output layer
$Err_j = O_j(1 - O_j)(T_j - O_j)$
for each unit j in the hidden layers, from the last to the first hidden layer
$Err_j = O_j(1 - O_j) \sum_k Err_k w_{jk}$
for each weight $w_{ij}$ in N {
$\Delta w_{ij} = (l) Err_i O_i$
$w_{ij} = w_{ij} + \Delta w_{ij}$}
for each bias $\theta_j$ in N {
$\Delta\theta_j = (l) Err_j$
$\theta_j = \theta_j + \Delta\theta_j$
}}
```

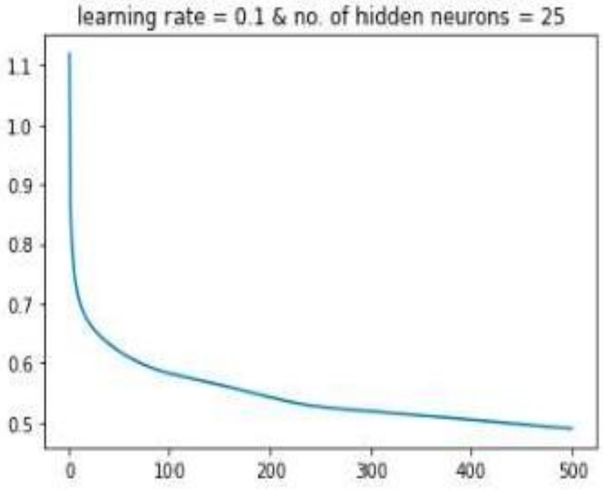
Optimizing Hidden Neurons

RMSE is plotted for different number of hidden neurons i.e. 10,15,20 and 25, with a learningrate of 0.1 and number of iterations = 500.


learning rate = 0.1 & no. of hidden neurons = 10


learning rate = 0.1 & no. of hidden neurons = 15


learning rate = 0.1 & no. of hidden neurons = 20

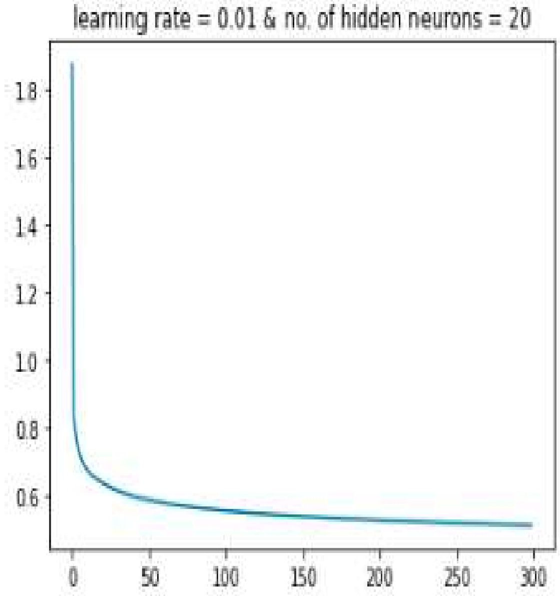

learning rate = 0.1 & no. of hidden neurons = 25

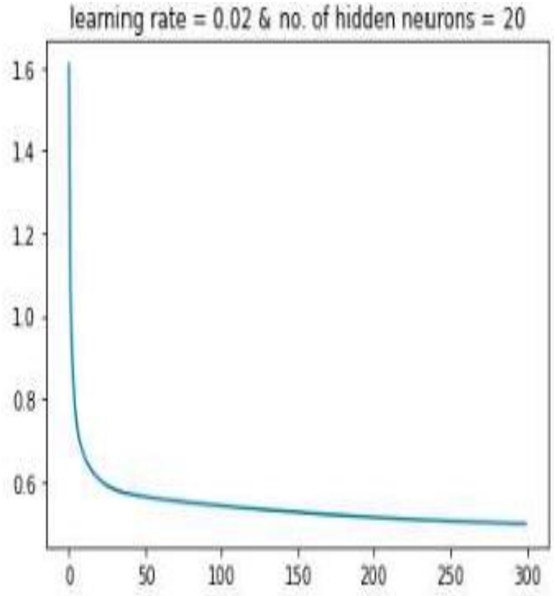
From the above error graphs, number of neurons = 10,15,20 and 25 almost gives the same pattern.

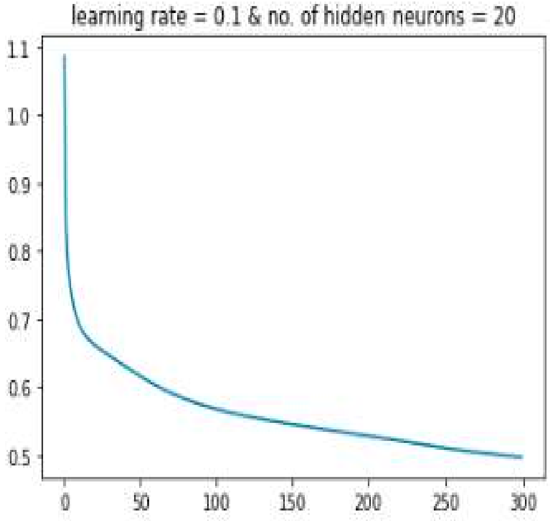
Hence **Optimal Number Hidden Neurons is taken as 20**, as it is more time efficient than 10,15 and 25neurons .

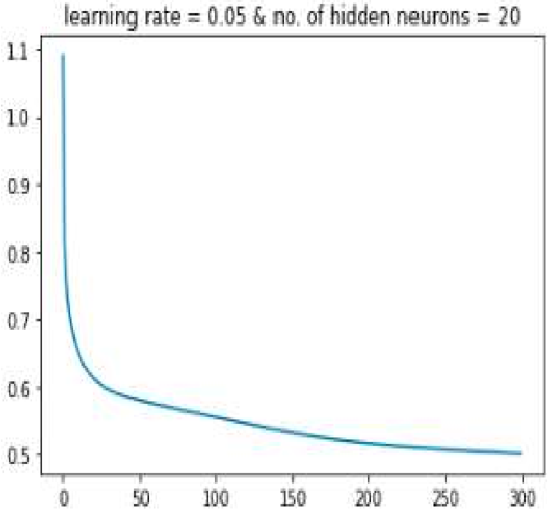
Optimizing Learning Rate & Number of Iterations

RMSE error is plotted for different learning rates i.e. 0.01, 0.02, 0.05, 0.1 with 20hidden layer neurons and number of iterations = 500.







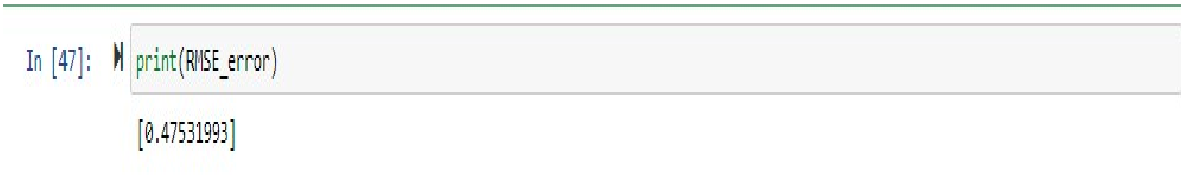


From the above error graphs, a learning rate between 0.02 and 0.01 is feasible with proper learning i.e. proper updation of weights and biases.

Hence **Optimal learning rate is taken as 0.01 with 500 iterations.**

RESULTS AND DICUSSIONS

The results are presented in below Data Frame. RMSE for of the outputs is given.



```
In [47]:  print(RMSE_error)

[0.47531993]
```

Final Hyperparameters for training model:

- Learning Rate (Ƞ): 0.01
- # of Hidden Layer Neurons: 20
- # of Iterations: 500

From the above table it can be seen that average RMSE is around 0.47, which indicates a successful regression model. The RMSE can be further decreased by increasing the number of hidden layers in the model and as well further hyperparameter tuning.

CONCLUSION AND REFERENCES

It can be concluded that a Single Hidden Layer Perceptron can be used to successfully regress the models for detect a heart dieses , using an artificially generated set of data. The results achieved show that errors below 1.00 can be reached and my model error is less than 7 %, with further improvements being possible through the finer tuning of hyperparameter values.

REFERENCES

□ https://www.kaggle.com/datasets/unsdsn/world-happiness

□ School of Computer Science and Technology, Wuhan University of Technology, "China An Empirical Study of Learning Based Happiness Prediction Approaches". Received 30 April 2021 Accepted 20 June 2021

□ L. Gao, F. Chen and Y. Lei, "Research on the Integration of Happiness Economics and Happiness Management", *2009 International Conference on Management and Service Science*, pp. 1-4, 2009

□ https://machinelearningmastery.com/implement-backpropagation-algorithm-scratch-python/ https://