Dubai Campus

F21DM

Data Mining and Machine Learning

Coursework 1

Implementing Nearest Neighbor Algorithm

Authors:
Heba El-Shimy

Student Number:
H00280277

Contents

1 Introduction 2

2 Loading and Manipulating the Data 2

3 Deeper Analysis and Attribute selection 2

4 Experiments 3

5 Conclusions 5

1 Introduction

This report briefly discusses the implementation of Nearest Neighbour Algorithm on image dataset in order to learn different emotions and classify them correctly when introduced to new data.

2 Loading and Manipulating the Data

We chose working with Python and the available libraries for it to complete this coursework. The editor we used for this purpose was Jupyter Notebook to run and test our code.

For loading the dataset, we used the Pandas library, to read directly from the provided csv file for the dataset. We used pandas to manipulate the data and transforming the image pixels string into separate columns for each pixel, that will later represent a feature or attribute. We started with 2 columns and ended up with 2305 columns, 2304 of them are a column for each pixel in our 48x48 pixels image data.

We repeated the previous process for all emotion datasets and we wrote the resulting data into a new csv as well as serializing it and writing it to a pickle file using pandas.

The next step we did was randomizing our data, we used Scikit-Learn for that purpose, particularly the shuffle method, the randomly changes the order of the instances in the dataset. We used a random state to be able to regenerate the random pattern later on.

Lastly, we normalized our data, by using Scikit-Learn's normalize method, which converts all inputs into unit norm, making all input data ranging between the values 0 and 1.

We decided on using the full dataset with python, as pandas performance is relatively good on large datasets and can handle them. We only used a reduced version of the dataset for Weka and that version was generated after performing the attribute selection methods using python.

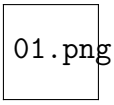
3 Deeper Analysis and Attribute selection

The process of

4 Experiments

Experiment Number	Initial weights	Initial learning Rate	LR decay*	Number of Epochs	Cost at last epoch	Accuracy
01	$W = [-0.5, 0.5]$ [3]	1	lr /=2, epochs %5	100	nan	64%
02	$W = [-0.5, 0.5]$ [3]	0.1	lr /=2, epochs %5	100	2.04	56%
03	$W = [-0.5, 0.5]$ [3]	0.01	lr /=2, epochs %5	1000	2.80	40%
04	$W = [-0.05, 0.05]$ [2]	0.1	lr /=2, epochs %5	100	0.58	64%
05	$W = [-0.05, 0.05]$ [2]	0.1	lr /=2, epochs %5	1000	0.58	64%
06	$W = [-0.05, 0.05]/sqrt(n)$ [2][1]	0.1	lr /=2, epochs %5	100	0.56	58%
07	$W = [-0.5, 0.5]/sqrt(n)$ [3][1]	0.1	lr /=2, epochs %5	100	0.50	60%
08	$W = [-0.05, 0.05]/sqrt(n)$ [2][1]	0.01	lr /=2, epochs %50**	1000	0.41	64%
09	$W = [0, 1]/sqrt(n)[1]$	0.1	lr /=2, epochs %100***	1000	nan	70%
10	$W = [0, 1]/sqrt(n)[1]$	0.01	lr /=2, epochs %100***	1000	0.12	72%
11	$W = 0.1 * [0, 1][1]$	0.01	lr /=2, epochs %50****	1000	0.14	76%
12	$W = [-0.5, 0.5]/sqrt(n)$ [3][1]	0.1	lr /=2, epochs %100	1000	0.14	78%

Below are some screenshots of the output of testing on new images.


01.png

5 Conclusions

Using the biological concept of a neuron and training it using backpropagation allowed for solving complex problems that were not easy to solve using rule-based systems, nor simple machine learning statistical functions. Artificial neural networks perform well in problems that relate to computer vision where a huge input is expected in the form of pixel data. I discussed in this report the building block of ANNs which is a single neuron and how to train it to classify cat images.

References

[1] Cs231n convolutional neural networks for visual recognition, 2018.

[2] Mercedes Fernández-Redondo and Carlos Hernandez Espinosa. Weight initialization methods for multilayer feedforward, 01 2001.

[3] Marta Vallejo. Biologically inspired computation, 2018.