USKUDAR UNIVERSITY

FACULTY / INSTITUTE OF ……………….

……………….. DEPARTMENT / PROGRAM

FINAL PROJECT REPORT

STUDENT ID:

NAME LASTNAME:

PROJECT TITLE: Machine Learning

1. INTRODUCTION

Machine Learning is the use of data and models that use algorithms to try and imitate the way humans learn and try to initiate or predict a task, trying to increase accuracy everytime. Machine Learning is a new breakthrough in technology since in the past computers were given a set of tasks and a certain way of learning that may not be optimal or that could skip trends that humans cannot see. Using Machine Learning, there are certain models and pattern recognition theories for the computer to learn from data and try to predict without being programmed to do so initially. This gives the computer more space to find deeper trends. Machine Learning is the basis of Artificial Neural Networks which are computer systems which try and emulate a real human brain. It is made up of a collection of nodes which are meant to represent neurons in the brain and are programmed to simulate the electrical synapses that occur. Just as in the brain, each two nodes has a certain weight that increases after each synapse which then increases the strength at this connection, which tries to show how the brain learns with repitition. To train the ANN, just like training a ML model, the network is given an input with its known output, which is then passed through the network to build the weights and then a test input is passed to check the output to see the difference and amount of error. This is repeated until accuracy of the ANN is acceptable (human-like).

There are four types of learning paradigms that can be programmed for the ANN:

* Supervised Learning where the output needed is known, and is given with its respective input for the network to learn and then evaluate its prediction based on a cost function provided by the programmer.
* Unsupervised Learning where the input and the cost function are given to the network and the computer iterates through the inputs to find the combination that minimizes cost.
* In reinforcement learning, a cost function is changed after each iteration and the goal of the network is to minimize the long term cost.
* Finally in self learning, the computer is only given one situation and one output and the computer iterates through all actions to understand and lead to the output.

In this paper, we will start building a network by starting with a machine learning problem, where we built a model that predicts the price of a house in in Taiwan based on historical data.

1. MATERIALS AND METHODS

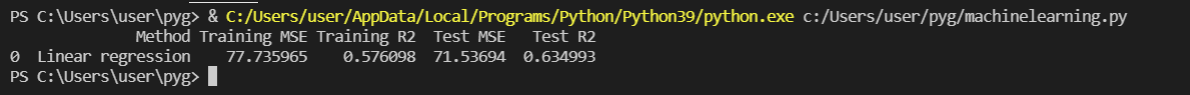
The dataset used (<https://archive.ics.uci.edu/ml/datasets/Real+estate+valuation+data+set>) is a multivariate dataset where all its attributes are real continuous numbers. This data is a market historical data collected and donated from a Civil Engineering Professor I-Cheng Yeh, from the Tamkang University in Taiwan. There are 414 instances of real estate valuations from Sindian Dist., New Taipei City, Taiwan. This dataset is used for a regression problem since the output we will analyze and predict is a continuous integer. There are 6 attributes that are given in each sample to predict the house price and they are: the transaction date, the house age in years, the distance to the nearest MRT station in meters, the number of convenience stores in the living circle on foot, the georgraphic latitude in degrees, and the geographic longitude in degrees. The output is the house price of unit area per ping (where 1 Ping = 3.3 meters squared).

The Machine Learning model will be written on python and will use the Linear Regression model since the data is a regression problem. The code will be written in VS code and will use scikit-learn module to import the models and accuracy scores used. The python file will also need pandas to read the data file and parse it in python for training the model and testing it. The main values to find how accurate the model is are the MSE and R2 values which are the best to use for linear regressions.

1. CODES
2. import pandas as pd
3. import numpy as np
4. df = pd.read\_excel('realestate.xlsx')
5. X = df.drop(['Y house price of unit area'], axis = 1)
6. y = df['Y house price of unit area']
7. from sklearn.model\_selection import train\_test\_split
8. X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.1, random\_state=42)
9. from sklearn.linear\_model import LinearRegression
10. #Regression Method
11. lr = LinearRegression()
12. lr.fit(X\_train,y\_train)
13. y\_lr\_train\_pred = lr.predict(X\_train)
14. y\_lr\_test\_pred = lr.predict(X\_test)
15. from sklearn.metrics import mean\_squared\_error, r2\_score
16. lr\_train\_mse = mean\_squared\_error(y\_train, y\_lr\_train\_pred)
17. lr\_train\_r2 = r2\_score(y\_train, y\_lr\_train\_pred)
18. lr\_test\_mse = mean\_squared\_error(y\_test, y\_lr\_test\_pred)
19. lr\_test\_r2 = r2\_score(y\_test, y\_lr\_test\_pred)
20. lr\_results = pd.DataFrame(['Linear regression',lr\_train\_mse, lr\_train\_r2, lr\_test\_mse, lr\_test\_r2]).transpose()
21. lr\_results.columns = ['Method','Training MSE','Training R2','Test MSE','Test R2']
23. print(lr\_results)
24. date = input("Please enter transaction date: ")
25. age = input("Please enter the age of the house: ")
26. distance = input("Please enter distance to the nearest MRT station: ")
27. number\_stores = input("Please enter number of convenience stores: ")
28. latitude = input("Please enter the latitude of house: ")
29. longitude = input("Please enter the longitude of house: ")

* In lines 1 and 2, we import the needed modules to be able to create a dataframe from the data document provided.
* In Line 3, we use pandas function read\_excel to be able to read the excel data file and parse it into pandas table (dataframe) called df here.
* In lines 5 and 6, we set the attributes in X and the output variable in y to be able to train the models later on. In X we drop the output variable so it holds all the samples with all 6 attributes while we assign the output to y in line 6.
* In Lines 7 and 9, we import the module scikit-learn and its function train\_test\_split which is used to split the data in table for learning data which is used to train the model and then testing data where the model predicts values to calculate the accuracy. Random\_state = 42 is set a random seed generator to shuffle the data and test\_size of 0.1 means that 90% of the data will be used for training and 10% will be used for testing.
* In lines 11 and 14, we import the Linear Regression model from the scikit-learn module and we create a linear regression in the variable lr.
* In line 15, the function fit is a function in the module that is used to train the model and is passed 2 arguments X\_train and y\_train which are the attributes of each sample and the output for each set of attributes that were set for training.
* In line 17, after the model is trained, we use the function predict and pass an argument of the attributes from the training set to see how the model does on familiar data and is saved in the variable y\_lr\_train\_pred.
* In line 18, the model tries to predict on the testing data which is not familiar to see how accurate it is and the predictions are stored in the variable y\_lr\_test\_pred.
* In lines 20-24, we import the metrics from the scikit-learn module that are used to find how accurate our linear Regression model is, which are the Mean Squared Error (MSE) and the R2 score. We then find the MSE and R2 for each of the two prediction sets y\_lr\_train\_pred and y\_lr\_test\_pred.
* In lines 26 and 27, we create a pandas dataframe which is a table to show the results for the MSE and R2 of the linear regression. In line 27, the columns of the dataframe are given names. The transpose function is used to show the table horizontally and not vertically.
* In line 29, we print the dataframe stored in lr\_results which shows a table in the terminal.

1. RESULTS



As you can see from the output below the Mean Squared Error of the testing that it is lower than the MSE for training and that the R2 also increased therefor the model was accurate and around 64% of testing data which is pretty good for real estate valuations and enough to beat the real estate in market in purchasing and selling leading to a 14% profit advantage over the market.

1. DISCUSSION AND CONCLUSIONS

In conclusion, the model building was a success since we were able to build a real estate linear regression price prediction model that is able to beat the market and predict the price of the house based on historical data and different attributes, although it was not perfectely accurate in prediction, further training and testing data will increase accuracy of the model and can lead to develop AANs that are able to find the house price in any location in the world, which can be a breakthrough in real estate investment and change how people percieve house prices in different locations.

REFERENCES

Yeh, I. C., & Hsu, T. K. (2018). Building real estate valuation models with comparative approach through case-based reasoning. Applied Soft Computing, 65, 260-271.

<https://en.wikipedia.org/wiki/Artificial_neural_network#:~:text=Artificial%20neural%20networks%20(ANNs)%2C,neurons%20in%20a%20biological%20brain>.

<https://www.sciencedirect.com/topics/engineering/artificial-neural-network>

<https://archive.ics.uci.edu/ml/datasets/Real+estate+valuation+data+set>

https://www.ibm.com/cloud/learn/machine-learning#:~:text=Machine%20learning%20is%20a%20branch,learn%2C%20gradually%20improving%20its%20accuracy.&text=Machine%20learning%20is%20an%20important,growing%20field%20of%20data%20science.

CHECKLIST

|  |  |
| --- | --- |
|  | YES / NO |
| Did you study on the method assigned to you? | YES |
| Did you prepare your report as mentioned in the template? | YES |
| Did you add the results (print screen) of your study to the report? | YES |
| Did you rename your report file as asked in the template? | NO |
| Are you uploading the report to the system? | NO |
| Are you uploading the codes to the system? | NO |