­­

11/10/2022

eman salah eldin

Zep Analytics

# 

Social Network Ads

[Document subtitle]

Table of Contents:

[1. Introduction: 2](#_Toc118689243)

[2. About data: 2](#_Toc118689244)

[3. Analysis and model the data 2](#_Toc118689245)

[1. Read data: 2](#_Toc118689246)

[2. Prepare and analysis the data: 3](#_Toc118689247)

[3. plot the data: 5](#_Toc118689248)

[4. split the data: 7](#_Toc118689249)

[5. The model: 7](#_Toc118689250)

[4. Conclusion: 11](#_Toc118689251)

# Introduction:

Machine learning is an exciting field, we use it in different fields of (data science, Ai)

It has three kinds of learning, supervised learning (labeled data): describes a class of problem that involves using a model to learn a mapping between input examples and the target variable, unsupervised learning (unlabeled data): describes a class of problems that involves using a model to describe or extract relationships in data. , reinforcement learning : describes a class of problems where an agent operates in an environment and must learn to operate using feedback , predict the next state based on (the current state , the previous action and the reward of this (state-action))

# About data:

This data is called social network ads, I am downloaded from Kaggle dataset <https://www.kaggle.com/datasets/micheldc55/social-network-ads>

Data has information on gender, age and whether the user purchased the product shown to them or not. And I must determine if the user will buy or not

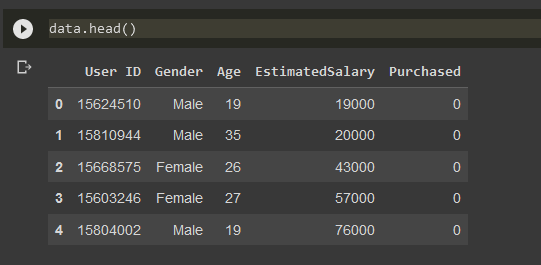
This problem is supervised learning (labeled data), features and class with its values:

1. Gender (Male or Female)
2. Age (18-60)
3. Estimated salary (15000-150000)
4. Purchased (buy or not buy)

# Analysis and model the data

## Read data:

I used pandas library from python to read csv file and print the data



## Prepare and analysis the data

1. normalize the data to make it in the same range: change (Age-Estimated Salary) columns

A screenshot of a computer

Description automatically generated with medium confidence

1. shuffle the data to make the model trained well

A screenshot of a computer

Description automatically generated with medium confidence

1. check the missing values in the data, outliers, duplicated rows in the data
2. First check missing values:

there are no missing values in data

Text

Description automatically generated

1. Second check outliers:

by using boxplot from seaborn library :(Age-Estimated Salary) columns

Chart

Description automatically generated

Chart, histogram

Description automatically generated

1. Check duplicated rows:

There are 20 rows duplicated and solve it by using pandas library with drop duplicates function to remove the 20 rows from the data

Before:

Graphical user interface, application

Description automatically generated

After:

Graphical user interface, text, application

Description automatically generated

## plot the data:

using matplotlib library to plot histogram before and after the normalization for each (Age-Estimated Salary) columns :

Before:

Chart, histogram

Description automatically generated

Chart, histogram

Description automatically generated

After:

Chart, histogram

Description automatically generated

Chart, histogram

Description automatically generated

## split the data:

Usually in machine learning, split data for train validate test, to train the model first and then test the model with unseen data

In this model split the data for 70 % training, 20 % validation, 10% test

## The model:

1. Using (Naive bayes, SVM, Logistic regression) classifiers
2. Using cross-validation with k=5 with (Naive bayes, SVM, Logistic regression)

I used cross-validation because the data is small, and I try to find a solution by cross-validation

1. Using neural network with 3 layers (input -hidden- output)
2. Input layer: the number of neurons is the number of the input of the data is 3
3. Hidden layer: the number of neurons is 16 the best accuracy compared with before it and similar accuracy compared with after it
4. Output layer: the number of neurons is the number of the classes is 2

|  |  |
| --- | --- |
| Classifier | Accuracy |
| Naïve Bayes classifier | Training Accuracy: 0.8834586466165414  Validation Accuracy: 0.9078947368421053  Test Accuracy: 0.868421052631579 |
| SVM | Training Accuracy: 0.8947368421052632  Validation Accuracy: 0.9342105263157895  Test Accuracy: 0.868421052631579 |
| Logistic Regression | Training Accuracy: 0.8383458646616542  Validation Accuracy: 0.8421052631578947  Test Accuracy: 0.868421052631579 |
| Naïve bayes with cross-validation in all the data | Average score: 0.881578947368421 |
| SVM with cross-validation in all the data | Average score: 0.9026315789473683 |
| Logistic Regression with cross-validation in all the data | Average score: 0.8342105263157895 |
| Neural Network | Training Accuracy: 0.9060150375939849  Validation Accuracy: 0.8947368421052632  Test Accuracy: 0.868421052631579 |

# Conclusion:

After building these models and looking at their accuracies, the best model is (SVM and neural network) trained well and predict high accuracy in train validation and test accuracy, and the model with the least accuracy is logistic regression