INFO 7390

Advances in Data Sciences and Architecture

Assignment 3

Professor: Nik Bear Brown

Due: July 29, 2019

In this assignment you will be explaining how a machine learning algorithm works in detail.

Q1 (5 Points)

Pick an algorithm that you will explain. You MUST post on piazza and the TA MUST approve your choice. No two students can choose the same algorithm.

Explain what the algorithm is used for.

**Answer:**

**Algorithm selected :** K-Nearest neighbors’ algorithm.

**Kind of algorithm : Supervised machine learning algorithm**

**Used for: Both classification and regression problems.**

**Description:** K-nearest neighbor algorithm assumes that similar kind of things are close to each other. It stores all available use cases and classifies new cases based on a similarity measure(e.g. distance functions). It is used in domains of healthcare, finance, political science, image recognition, video recognition etc.

Some examples where this algorithm can be used :

1. In banking system, it can predict the credit ratings of customers.
2. In healthcare, it can predict whether patient have diabetes or not.
3. Classifying potential voters whether they will vote or not vote
4. In loan dismemberment, whether loan is safe or risky.

Q2 (45 Points)

Explain the mathematics behind the algorithm in detail. This should NOT be a hand wavy vague explanation.

How fast does it run?

Will scaling effect it.

Explain the equations.

**Answer:**

K in KNN is the number of neighbors and it is the essential factor in this algorithm. K is usually considered to be an odd number if there are only 2 classes.

KNN is a lazy learning algorithm which means that it does not need any training data points for model generation and all the training data used in the testing phases.

Let’s consider a point P which needs to be predicted that under which class that point will be labeled. First step is to consider k neighboring points around point P and then classify them by taking a majority vote of its neighbors. Each object votes for their class and class with most votes is taken as the prediction. In order to find similar distances , we need to find distance between points using distance measures such as Euclidean distance, Hamming distance etc.

KNN has following steps:

1. Load the data
2. Initialize the value of k
3. For getting the predicted class, iterate from 1 to total number of training data points.
4. Calculate the distance between the test data and each row of training data.
5. Sort the calculated distances in ascending order based on distance values.
6. Get the top k rows from the sorted array
7. Fetch the most frequent class of these rows
8. Return the predicted class.

A picture containing text, whiteboard

Description automatically generated

**Euclidean distance is calculated using the formula:**

If p = (p1, p2 , …pn) and q = (q1, q2, ….qn) are two points in Euclidean space then distance from p to q or from q to p is given by the Pythogorean theorem.

**D(p,q) = d(q,p) = √(q1-p1)2 + (q2-p2)2 + ….(qn-pn)2**

Other distances are also used.

A close up of text on a white background

Description automatically generated

KNN performs better with a smaller number of features as compared to a greater number of features. If the count of features increases then there are chances of overfitting the data. It is essential to decide the appropriate value of k. When there are larger number of features then cosine distances or other distance measures except Euclidean are more useful.

Q3 (35 Points)

Show an example in code of the algorithm working on a real data set. This must be in a notebook and must run.

Answer: Please refer Jupyter notebook

Q4 (10 Points)

Did I explain my idea clearly overall? (How effective are you at explaining what you are doing?

Answer:

I have taken up Diabetes dataset and trying to predict whether patient has diabetes or not. I have used KNN algorithm.

First I have loaded the dataset, analyzed the dataset by checking features , shape etc. Then we have set up 2 NumPy arrays for features and target variables.

I have then split the data into two sets – training dataset and testing dataset. I have chosen 30% to be the testing dataset.

Imported knnClassifier and setup 2 arrays to store data for training and testing data accuracy. We have then run the loop which will start from 1 to length of all training points. In order to identify suitable value of k, we plotted a graph with testing accuracy data and training accuracy data. As per the graph, we identified that 7 can be the suitable value for k.

Then using value of k as 7 we have tried fitting data into the model and generated accuracy of 74%.

I tried also checking the effect if k increases to 9 then accuracy score decreases.

Inorder to understand the effect, we have used GridSearch CV for hyperparameter tuning and found out that we can achieve accuracy of 76% when value of k is 14.

Q5 (5 Points)

Citations. ALL sources MUST be cited.

Answer: References:

<https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn>

<https://www.kaggle.com/amolbhivarkar/knn-for-classification-using-scikit-learn>

<https://www.kaggle.com/uciml/pima-indians-diabetes-database/downloads/pima-indians-diabetes-database.zip/1>

<https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/>

<https://medium.com/machine-learning-101/k-nearest-neighbors-classifier-1c1ff404d265>

<https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761>

<https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn>