

MNIST Fashion Dataset classifier

Abstract

In recent years, image recognition has become an important branch of optical character recognition. This paper uses KNN and decision tree algorithm to recognize MNIST Fashion Dataset. Firstly, this paper introduces the implementation principle and characteristics of KNN and decision tree algorithms. Since image recognition has a total of 10 types of samples, which is a multi-classification problem, this paper divide the samples into training samples and test samples. Both training samples and test samples are pre-processed to get lattice data, and then feature extraction is used to train the recognizer and identify. Then compares the confusion matrices, recall, precision, F1 score and classification accuracy of KNN and decision tree algorithms through experiments. Finally, the above work is summarized. To sum up, image recognition is a cross field of modern pattern recognition and intelligent application, which not only has important theoretical value, but also has broad application value.

1 Introduction

Image recognition is a field of computer science that uses images to study, comprehend, and identify things and objects with distinct patterns. There are three phases in the evolution of image recognition: character recognition, digital image processing and recognition, and object recognition. This type of recognition challenge is often translated into classification problems in the field of machine learning. KNN and decision tree algorithm have many attractive advantages and promising experimental performance, and have very ideal results in face recognition, image recognition, web page classification, etc. Therefore, this article uses KNN and decision tree algorithm to classify the Fashion-MNIST dataset. The Fashion-MNIST data set, in contrast to the MNIST handwritten data set, has ten picture categories: T-shirt, jeans, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot.

2 Related work

2.1 MNIST Fashion

The MNIST Fashion dataset is a well-known image recognition dataset that takes the role of the MNIST handwritten numeral dataset. It is frequently used to assess how well image character image algorithms function. It is offered by Zalando's research division (a German fashion technology company). It has a total of 70000 uplifting images of various items from 10 distinct categories. The MNIST Fashion dataset is identical to the original MNIST in terms of size, format, and training/test set division. 28x28 grayscale picture, 60000/10000 training test data division. The fashion dataset from MNIST is shown in Figure 2-1.

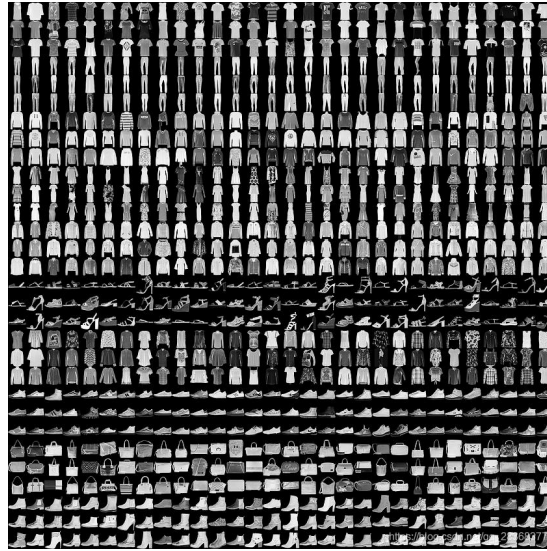


Figure 2-1. MNIST Fashion

2.2 Research status of image recognition based on Machine Learning

In the development of machine learning, traditional classifiers such as K nearest neighbor (KNN), decision tree (DT), Bayesian network (BN) and support vector machine (SVM) have exerted some influence on image recognition.

In 1968, Cover proposed a new algorithm called K-nearest neighbor (KNN) algorithm. The fundamental tenet is that a sample belongs to a category if the majority of the K most comparable samples in the feature space (i.e., the sample's nearest neighbor) do as well. In 1986, the decision tree (DT) algorithm was proposed by Quinlan. The decision tree is a tree with decision criteria, which predicts the categories and values of unknown samples according to the decision criteria in the tree. The commonly used training algorithms of decision tree are greedy algorithm (ID3), improved version of greedy algorithm (C4.5) and classification regression tree algorithm (CART). In 1988, a network proposed by Pearl, which is based on statistical information theory, uncertainty reasoning and data analysis theory, combined with the need to determine the topological relations between random variables and construct conditional probability tables, is called Bayesian networks (BN).

The commonly used algorithms in this network are scoring algorithm (K2 algorithm) and mountain climbing algorithm. SVM, which is based on statistical theory proposed in 1995, utilizes the concept of kernel function to convert the nonlinear classification issue in low-dimensional space into the linear classification problem in high-dimensional space.

Through the combination of several SVM construction algorithms, can be used to achieve multi-classification problems, the main methods are one-to-one algorithm (OVO), one-to-many algorithm (OVA), directed acyclic graph algorithm (DAG). Domestic and foreign scholars have improved the traditional classifier and achieved good results. Bu Fanjun improved the KNN algorithm, by measuring the shortest distance between different eigenvalues, calculating the K value for classification, and finally obtained a recognition rate of 82.3%.

By extracting the structural features of handwritten numbers and using ID3 algorithm to construct a DT classifier for handwritten numbers for classification, Chen Junsheng can achieve a recognition rate of 86.7%. Taking advantage of association rules to mine data, Zhang Yanfang proposed a BN network based on association rules and mountain climbing algorithm, and achieved a recognition rate of 87.1%. LiuK improves the connection weight and learning parameters of BN model, and proposes a new BN classifier to establish image recognition model. The results show that the new BN classifier can achieve a recognition rate of 87.9% in image recognition. Shi Huifang randomly selected 1050 training samples and 1520 test samples from the MNIST dataset, and optimized the c and g parameters of the kernel function by grid analysis algorithm, genetic algorithm and particle swarm optimization algorithm respectively. According to the experimental results, the classification accuracy of these three algorithms is 92.1%, 92.3% and 93.4% respectively. Eva Tuba, using x-axis, y-axis, y-axis and y-axis projection histogram respectively to extract features from MNIST data set, and select SVM for classification. The experimental results show that the recognition effect of y-axis projection histogram is the best, and the recognition rate is 93.28%.

3 Artificial intelligence algorithm

3.1 Brief introduction of KNN algorithm

One of the simplest techniques in data mining classification technology is the KNN (K-Closest Neighbor) nearest neighbor classification algorithm. Its guiding principle is "those who are close to red, those who are close to ink are black," which means that your neighbor will infer your category.

The following describes the implementation technique for the KNN closest neighbor algorithm: To categorize the unknown samples, choose the K known samples that are closest to them. Using the known samples as a baseline, measure the distance between each sample and each other sample. The K closest samples and the unidentified samples are in the same category, according to the majority vote criteria. Because it simply chooses the category of the sample to be classified based on the category of the nearby sample or samples, rather than the method of discriminating the class domain, the KNN closest neighbor classification algorithm is better suited than other approaches for the sample set with more overlapping or overlapping class domains. The KNN algorithm's summary is shown in Figure 3-1.

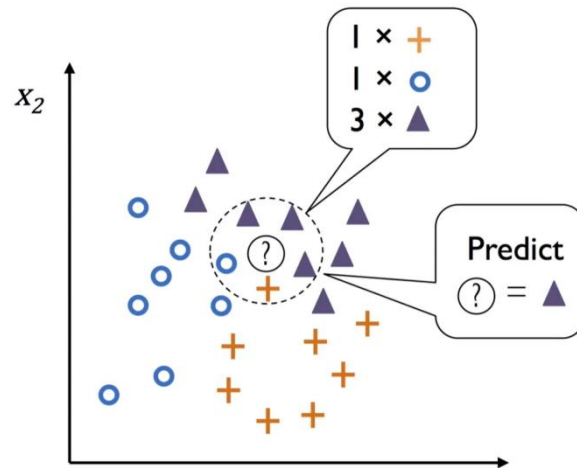


Figure 3-1. Overview of KNN algorithm

- The key to KNN algorithm:

- 1) All sample features should be quantitatively compared; if any sample features are non-numerical kinds, steps must be performed to convert them to numerical values. For instance, you may determine the distance if the sample feature has a color by changing the color to a grayscale value.

- 2) The sample's characteristics should be normalized; the sample contains a number of parameters, each of which has a distinct definition domain and value range as well as a variety of effects on distance estimation, such that the parameters with larger effects will outweigh those with smaller ones. The easiest technique to handle scale processing for the sample parameters is to normalize the values of all features.

- 3) A distance function is necessary to calculate the distance between two samples. The Cosine distance, Manhattan distance, Hamming distance, Euclidean distance, and others are examples of common distance functions. Euclidean distance is the most widely utilized distance metric, however it is only applicable to continuous variables. In the case of discontinuous variables, including such text classification, the hamming distance can be used as a measurement. By applying specialized methods to build the metric, such as the large edge nearest neighbor methodology or nearest neighbor component analysis, the accuracy of K-closest neighbor classification may frequently be significantly improved.

- 4) Establish the value of K. Because underfitting and overfitting are both easily caused by K values that are too high or too little, the value of K must be established by cross-validation.

3.2 Brief introduction of of decision Tree algorithm

A technique for roughly estimating the value of a discrete function is the decision tree algorithm. It is a common categorization strategy. It examines data first, using induction techniques to produce comprehensible rules and decision trees, and then uses decisions to examine fresh data. Decision trees are essentially a method of categorizing data using a set of rules. The decision tree algorithm's summary is shown in Figure 3-2.

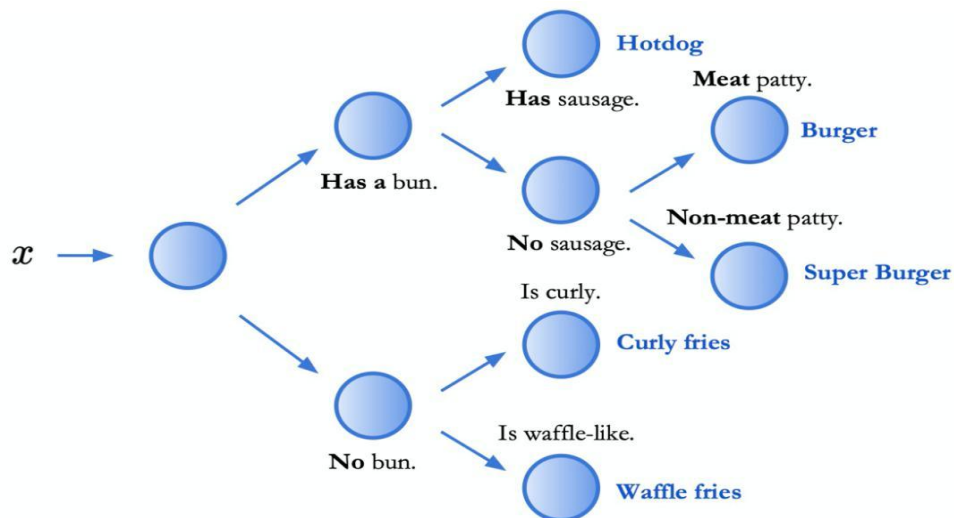


Figure 3-2. Overview of decision tree algorithm

- Construction of decision Tree:
 - 1) Feature selection: select the features with strong classification ability.
 - 2) Decision tree generation: the typical algorithms are ID3 and C4.5, and their decision tree generation processes are similar. ID3 uses information gain as a feature selection metric, while C4.5 uses information gain ratio.
 - 3) Decision tree pruning: the reason for pruning is that the tree generated by the decision tree generation algorithm predicts the training data accurately, but the classification of unknown data is very poor, which leads to the phenomenon of over-fitting. The algorithm involved is CART algorithm.
- Partition and selection of decision Tree:
 - 1) Entropy: physical meaning is a measure of the degree of chaos in the system.
 - 2) Information entropy: a measure of the uncertainty of things, which can be calculated according to the probability in mathematics. The probability is large, there are more opportunities, and the uncertainty is small (information entropy is small).
- Pruning of decision trees:

Pruning: as the name implies, it "removes" some judgment branches from the decision tree, while still getting good results under the rest of the tree structure. The reason why pruning is to prevent or reduce the occurrence of "over-fitting phenomenon", is that the decision tree has better generalization ability.

Approach in detail: Discard the excessively subdivided leaf node, cause it to revert to the parent or even higher leaf node, and then convert the parent or higher leaf node to a new leaf node.

Two methods of pruning:

 - 1) Pre-pruning: pruning is performed when the decision tree is constructed. In the process of constructing the decision tree, the node is evaluated, and if the division of the node can no longer improve the accuracy of the verification set, then the node should not continue to be divided under the king. At this point, the current node is used as the leaf node.
 - 2) Post-pruning: pruning after the decision tree is generated. Each node is usually

evaluated layer by layer, starting from the leaf nodes of the decision tree. If the node is cut, the accuracy of the verification set is not different or significantly improved, it can be pruned and the leaf node can be used to fill the node.

3) Note that while the global optimization is taken into account during decision tree pruning, only local optimization is taken into account during decision tree formation.

4 Experimental results

4.1 KNN algorithm

- Confusion matrices of KNN can be shown in Figure 4-1:

```
Confusion_matrix of KNN:
[[863    0   20   19    5    0   84    1    8    0]
 [  8 964    6   13    4    0    4    0    1    0]
 [ 21    2 813    7  82    0   74    0    1    0]
 [ 37    4   14 871   37    0   35    0    2    0]
 [  1    0 128   25 755    0   90    0    1    0]
 [  1    0    0    1    0 807    5 100    2   84]
 [195    0 135   22   76    0 560    0   12    0]
 [  0    0    0    0    0    2    0 969    0   29]
 [  1    1   14    3    8    0   16    6 949    2]
 [  0    0    0    0    0    1    1   34    0 964]]
```

Figure 4-1. Confusion matrices of KNN

- Other evaluation indicators of KNN can be shown in Figure 4-2:

```
Other evaluation indicators of KNN:
           precision    recall  f1-score   support

T-shirt      0.7657      0.8630      0.8115      1000
Trouser      0.9928      0.9640      0.9782      1000
Pullover     0.7195      0.8130      0.7634      1000
Dress        0.9063      0.8710      0.8883      1000
Coat         0.7808      0.7550      0.7677      1000
Sandal       0.9963      0.8070      0.8917      1000
Shirt        0.6444      0.5600      0.5993      1000
Sneaker      0.8730      0.9690      0.9185      1000
Bag          0.9723      0.9490      0.9605      1000
Ankle boot   0.8934      0.9640      0.9274      1000

accuracy                    0.8515      10000
macro avg      0.8545      0.8515      0.8506      10000
weighted avg   0.8545      0.8515      0.8506      10000
```

Figure 4-2. Other evaluation indicators of KNN

4.2 Decision tree algorithm

- Confusion matrices of Decision tree can be shown in Figure 4-3:

```
Confusion_matrix of DecisionTree:
[[743   3   26   36   13    1 166    2    7    3]
 [   3 945    4   22    9    0   13    0    3    1]
 [  24    2 651   11 154    2 136    0   18    2]
 [  54   29   24 776   54    1   49    0   13    0]
 [  12    5 171   53 634    0 113    0   12    0]
 [   2    2    1    1    1 886    2   58   17   30]
 [142   12 131   43   99    4 546    1   20    2]
 [   0    0    0    0    0   44    0 892    9   55]
 [   8    0   18    8   13    8   22   12 906    5]
 [   0    0    0    0    1   23    1   60    4 911]]
```

Figure 4-3. Confusion matrices of Decision tree

- Other evaluation indicators of Decision tree can be shown in Figure 4-4:

```
Other evaluation indicators of DecisionTree:
              precision    recall  f1-score   support

T-shirt      0.7520      0.7430      0.7475      1000
Trouser      0.9469      0.9450      0.9459      1000
Pullover     0.6345      0.6510      0.6426      1000
Dress        0.8168      0.7760      0.7959      1000
Coat         0.6483      0.6340      0.6411      1000
Sandal       0.9143      0.8860      0.8999      1000
Shirt        0.5210      0.5460      0.5332      1000
Sneaker      0.8702      0.8920      0.8810      1000
Bag          0.8979      0.9060      0.9019      1000
Ankle boot   0.9029      0.9110      0.9069      1000

accuracy                    0.7890      10000
macro avg      0.7905      0.7890      0.7896      10000
weighted avg   0.7905      0.7890      0.7896      10000
```

Figure 4-4. Other evaluation indicators of Decision tree

5 Reason discussion

In this experiment, I use KNN and decision tree algorithm to classify MNIST Fashion data, and the experimental results show that the performance of KNN algorithm is better than that of decision tree algorithm on MNIST Fashion dataset.

Therefore, the characteristics of KNN algorithm and decision tree algorithm are

summarized as follows:

5.1 Characteristics of KNN algorithm

KNN is a model with a slow, nonparametric algorithm. Contrary to linear regression, non-parametric modeling indicates that the model does not make any assumptions about the data. This is not to say that the technique does not need parameters (we always assume that linear regression is a straight line). To put it another way, the KNN model structure is formed based on the data, which is better in accordance with reality as the situation in reality frequently contradicts the theoretical concept. Inertia denotes that the KNN algorithm may be trained quickly or without a defined procedure for gathering large amounts of training data.

- Principal benefits of KNN:

- 1) The theory is developed, and the idea is straightforward. Both classification and regression may be done with it.

- 2) Nonlinear classification is a possible application.

- 3) The complexity of the training process is less than that of support vector machine-type methods.

- 3) It has excellent accuracy, is immune to outliers, and makes no assumptions about the data, in contrast to algorithms like naive Bayes.

- 4) The KNN approach is more appropriate than other methods for the sample set to be categorized with more or more overlapping class domains because it relies mostly on the limited neighboring samples nearby rather than on the way of discriminating the class domain.

- 5) This approach works best for automatically classifying class domains with a big sample size, but it can easily misclassify class domains with a short sample size.

- Principal drawbacks of of KNN:

- 1) A significant quantity of calculation, particularly when there are a lot of characteristics.

- 2) The accuracy of uncommon category predictions is low when the samples are imbalanced.

- Ball tree modeling, for example, calls for a lot of memory.

- 4) It uses a lazy learning approach that, in essence, does not learn, making predictions take longer to make than they would with an algorithm like logical regression.

- 5) Compared to the decision tree model, the KNN model is less interpretable.

5.2 Characteristics of decision Tree algorithm

A supervised learning technique called the decision tree employs categorization to create a mathematical model based on the properties of the data in order to accomplish the tasks of data screening and decision-making.

- Principal benefits of decision tree:

- 1) The decision tree model may be envisioned, and it is simple to grasp and explain.

- 2) The quantity of data that has to be prepared is little, as opposed to other technologies that frequently require big data sets and the creation of virtual variables in order to eliminate missing data, which the algorithm is unable to reliably forecast.

3) The logarithm of the data points required to train the decision tree represents the temporal complexity of the decision tree method (i.e., the prediction data).

4) Datasets evaluated by other algorithms often only include one sort of variable, as opposed to categories that can handle numbers and data (and need to be converted appropriately).

5) Be able to handle issues with many outputs.

6) With a white box model, if a certain circumstance is detected in the model, it is simple to interpret this condition using Boolean logic; but, with a black box model (such as an artificial neural network), the outcomes may be more challenging to comprehend.

7) Statistical tests may be employed to validate the model and ascertain its dependability.

8) Based on the data results, it appears to perform well, despite the fact that its theory is somewhat at odds with the actual model.

- Principal drawbacks of decision tree:

1) Decision tree algorithm learners are able to produce complicated trees, however, overfitting, which lacks a basis for generalization, occurs. The idea of pruning—which entails determining the minimum amount of leaf nodes or the maximum depth of the tree—appears as a solution to this issue.

2) Because a slight change in the data might result in a totally new tree, the decision tree's output may be unstable; this problem can be resolved by utilizing an integrated decision tree.

3) It is commonly known that even for simple concepts, learning an optimal decision tree is an NP-complete problem. Heuristic algorithms, such as the greedy algorithm, form the basis of useful decision-tree learning algorithms since each node makes judgements that are locally optimal. Such algorithms are unable to guarantee the global optimum decision tree. This can be decreased with the aid of multi-tree training in an ensemble learner using replacement sampling for the features and samples.

4) Some ideas, like XOR checking or reuse, are challenging to comprehend despite the decision tree's ability to describe them clearly.

5) It is advised to train decision trees using balanced data because when some classes are dominating, decision tree learners are likely to produce biased trees.

6 Conclusion

At first glance, image recognition seems to be a very simple task, but in fact, due to the similarity of image features, computers often mistake two different types of pictures for the same type. In the image recognition system, it is difficult to distinguish accurately. And because the application fields of image recognition, such as accounting and finance, require high accuracy and high recognition speed, it is quite difficult to realize. At the same time, with the development of unmanned driving, if the computer makes mistakes in image recognition during driving, it may even cause life danger. However, with the development and popularization of artificial intelligence and information technology, the application field of image recognition

will be wider and wider, and this direction will become more and more popular. Therefore, it is urgent to develop a high-performance image recognition system.

At present, the main methods applied to image recognition are decision tree, neural network and KNN. As a new machine learning method, KNN has the advantages of simple, easy to understand, easy to implement, no parameter estimation, no training, suitable for rare events classification, especially suitable for multi-classification problems and so on. At the same time, the decision tree algorithm also has high efficiency, the decision tree only needs to be built once, used repeatedly, they have become a new hot spot of machine learning with good development prospects and a wide range of applications. Of course, it is also well applied to the image recognition system.

This paper mainly discusses the application of KNN and decision tree algorithm in image recognition. image recognition and artificial intelligence are new hotspots recently, both of which have their eye-catching advantages. If we combine them perfectly, I believe they can solve some problems very well. Their combination does have many advantages, and of course there are many directions that need to be further studied. For example, which algorithm is able to better address the problem of image recognition and so on.

References

- [1] Han Xiao, Kashif Rasul, and Roland Vollgraf, "*Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms.*" arXiv preprint arXiv:1708.07747, pp. 1–6, 2017.
- [2] Iosiadis A, Gabbouj M . On the kernel Extreme Learning Machine classifier[J]. Pattern Recognition Letters, 2015, 68(DEC.15):205-210.
- [3] Huang G B, Zhou H, Ding X. Extreme learning machine for regression and multiclass classification[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2012, 42(2): 513-529.
- [4] Ghosh M M A, Maghari A Y. A Comparative Study on Handwriting Digit Recognition Using Neural Network[C] // 2017 International Conference on Promising Electronic Technologies(ICPET). IEEE, 2017: 121-126.
- [5] Mccann M T, Jin K H. A Review of Convolutional Neural Networks for Inverse Problems in Imaging[J]. IEEE Signal Processing Magazine, 2017, 34(6) : 85-95.
- [6] Beohar D, Rasool A. image Recognition of MNIST dataset using Deep Learning state of the art Artificial Neural Network and Convolutional Neural Network[C] // 2021 International Conference on Emerging Smart Computing and Informatics (ESCI) AISSMS Institute of Information Technology. IEEE, 2021:1337-1341.
- [7] Lecun, Y, Bottou, L, Bengio, Y, & Haffner, P. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
- [8] Saha C, Faisal R H, Rahman M M. Bangla image Recognition Using an Improved Deep Convolutional Neural Network Architecture[C] // 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE). 2019: 63-68.
- [9] Chandra B, Varghese P P. Fuzzy SLIQ decision tree algorithm[J]. IEEE trans Syst Man Cybern, 2008, 38(5): 1294-1301.
- [10] Campos L M D. A Scoring Function for Learning Bayesian Networks based on Mutual Information and Conditional Independence Tests[J]. Journal of Machine Learning Research, 2006, 7(7):2149-2187.
- [11] CORTES C, VAPNIK V. Support Vector Networks[J]. Machine Learning, 1995, 20(3):273-297.
- [12] Guyon L, Weston J, Barnhill S, et al. Gene Selection for Cancer Classification using Support Vector Machine[J]. Machine Learning, 2002, 46(7):389-422.
- [13] Du, ZG., Pan, TS., Pan, JS., Chu, SC. (2022). QUasi-Affine Transformation Evolutionary Algorithm for Feature Selection. In: Wu, TY., Ni, S., Chu, SC., Chen, CH., Favorskaya, M. (eds) Advances in Smart Vehicular Technology, Transportation, Communication and Applications. Smart Innovation, Systems and Technologies, vol 250. Springer, Singapore. https://doi.org/10.1007/978-981-16-4039-1_14