## SVM

1. explanation and justification of methods developed

Support vector machines (SVM) is a binary classification model. Its basic model is a linear classifier with the largest interval as the feature. SVM also includes kernel techniques, which makes it a substantially nonlinear classifier. The learning strategy of SVM is to maximize the interval , and is also equivalent to the problem of minimizing the loss function. When the SVM solves problems, it has no limited in dimensions of the sample，even if the sample is tens of thousands of dimensions, which makes the SVM dramatically suitable for solving text classification problems. this is due to the kernel functions in it.

图片包含 游戏机, 钟表

描述已自动生成

C1 and C2 are the two categories, and their samples in the two-dimensional plane are shown above. The straight line in the middle is a classification function, which can completely separate the two types of samples. In general, if a linear function can completely separate the samples, the data is said to be linearly separable, otherwise it is called nonlinearly separable.in the graph, infinite lines can choose to separate the two part. the SVM will choose the best one which has the largest margin between the two types of samples.

图片包含 游戏机, 钟表

描述已自动生成

when the two part is nonlinearly separable, those part can transfer to high dimension and be linearly separable, and this can be done by the kernel functions in SVM.

Apply SVM in python:

from sklearn import svm

clf = svm.SVC()

clf.fit( trainX,trainY)

result = clf.predict(testX)

1. feature selection

this model is use text classification to predict the most relevant news articles for each of the 10 topics. Total use two different sklearn.feature\_extraction.text tool , CountVectorizer and TfidfVectorizer.

CountVectorizer is to count the frequency of each word in training text, then form a feature matrix, and each line represents the word frequency statistics of a training text. The idea is that according to all training texts, regardless of the order in which they appear, only count each vocabulary appears in the training text as a feature separately, then forming a vocabulary list. This method is also called the bag of words method.

TfidfVectorizer is a statistical method used to evaluate the importance of a word in a document set. The importance of a word increases proportionally with the number of times it appears in the document, but at the same time decreases inversely with the frequency of its appearance in the corpus. If a word or phrase appears frequently in an article with a high TF and rarely appears in other articles, it is considered that the word or phrase has a good class distinguishing ability and is suitable for classification. TF-IDF is actually: TF \* IDF.

Term Frequency (TF) refers to the frequency which a given word appears in the file. It is the number of occurrences of word w in document d count (w, d) divided the total number of words in document d size (d).

Inverse Document Frequency (IDF) is a measure of the general importance of words. The IDF of a specific word can be obtained by dividing the total number of files and the number of files containing the word, and then taking the logarithm of the obtained number. That is the logarithm of the divided of the total number of documents n and the number of files docs (w, D) appearing in the word w.

When all other parameter is same, the result of CountVectorizer is

图片包含 游戏机, 文字

描述已自动生成

When all other parameter is same, the result of TfidfVectorizer is

图片包含 监控, 游戏机, 黑色, 桌子

描述已自动生成

By comparing the two results, when use the TfidfVectorizer, the model has better performance.

1. evaluation metrics

in a binary problem, assume that yi = 1 corresponds to positive samples and yi = 0 corresponds to negative samples. Suppose established a classification model H, and a predicted value H (xi) will be output for each input sample xi, then comparing the predicted value H (xi) with the actual value yi, we will get the following four cases : H(xi)=1,yi=1, H(xi)=1,yi=0, H(xi)=0,yi=1, H(xi)=0,yi=0.

In the first case, the prediction is positive, and the output is also positive, so it is true positive (TP), in the second case, the prediction is positive, the output is negative, it is false positive (FP), in the third case, the prediction is negative, the output is positive, called false negative (FN),in the last case, the prediction is negative, the output is also negative, called true negative (TN).

In a test set, can get:

Npre=TP+TN

Ntotal=TP+TN+FP+FN

If define a test set where the number of positive samples is P and the number of negative samples is N, then : P = TP + FN, N = TN + FP

Therefore, the acc is actually equal to

Acc=TP+TN/TP+TN+FP+FN=TP+TN/P+N

The recall is equal to :

Recall=TP/TP+FN=TP/P

The precision is equal to :

Precision=TP/TP+FP

F1-score is equal to :

F1=2TP/2TP+FN+FP=2⋅Precision⋅Recall/Precision+Recall

The formula shows that recall reflects the classification model's ability to recognize positive samples. The higher the recall, the stronger the model's ability to recognize positive samples. Precision reflects the model's ability to distinguish negative samples. F1-score is a combination of the two. The higher the F1-score, the classification model is more robust.

The micro method refers to counting all the classes together. Specifically, to precision, it is to add the TP of all the classes and divide by the sum of the TP and FN of all the classes.

The macro method is to first calculate the precision of each class separately and then count average.

## BernoulliNB

1. explanation and justification of methods developed
2. conditional probability ：

The probability of another event happen when an event happen, such as the probability of event A occurring under the condition of event B:

P(A|B)=P(A∩B)/P(B)

multiplication rule of probability :

P(A∩B)=P(A)P(B|A)orP(A∩B)=P(B)P(A|B)

1. Bayes’s Rule :

If there are k mutually exclusive events, B1, B2 ···, Bk, and P (B1) + P (B2) + ·· + P (Bk) = 1 and an observable event A.then:

P(Bi|A)=P(Bi∩A)/P(A)=P(Bi)P(A|Bi)/P(B1)P(A|B1)+P(B2)P(A|B2)+···+P(Bk)P(A|Bk)

Base on above formula.

When only have two class:

If p1 (x, y)> p2 (x, y), then it is classified into class 1

If p1 (x, y)< p2 (x, y), then it is classified into class 2

Combine with Bayes’s Rule :

x, y represents the characteristic variable, ci represents the classification, p (ci | x, y) represents the probability of being classified into the class ci under the x, y. Therefore, combining conditional probability and Bayes' theorem,then:

If p (c1 | x, y)> p (c2 | x, y), then the class belongs to c1

If p (c1 | x, y)< p (c2 | x, y), then the class belongs to c2

1. Then in the text classification :

The vocabulary appearance is represented by word vector ω, which is composed of multiple numerical values, and the number of numerical values is the same as the number of vocabulary in the training set.

Therefore, the above Bayesian conditional probability formula can be expressed as:

p(ci|ω)=p(ω|ci)p(ci)/p(ω)

1. Apply BernoulliNB in python

from sklearn.naive\_bayes import BernoulliNB

clf = BernoulliNB()

clf.fit(trainX, trainY)

result = clf.predict(testX)

1. feature selection

same as before

1. hyper-parameter tuning

alpha : Floating-point type, optional, default 1.0, add Laplace repair / Lidstone smoothing parameters

fit\_prior : Boolean, optional, default True, indicates whether to learn a priori probability,Parameter False means that all class labels have the same prior probability

set fit\_prior= False , alpha change from 0.1 to 2 ,the trend of total acc change is increase then decrease. And the peak point is show in alpha=1.5.

set fit\_prior=True , alpha change from 0.1 to 2 ,the trend of total acc change is increase then decrease. And the peak point is show in alpha=1.5, and this model perform better then set fit\_prior= False.

So the best performance parameter set is alpha = 1.5, fit\_prior=True.

1. evaluation metrics

same as before

## MultinomialNB

1. explanation and justification of methods developed
2. conditional probability ：

The probability of another event happen when an event happen, such as the probability of event A occurring under the condition of event B:

P(A|B)=P(A∩B)/P(B)

multiplication rule of probability :

P(A∩B)=P(A)P(B|A)orP(A∩B)=P(B)P(A|B)

1. Bayes’s Rule :

If there are k mutually exclusive events, B1, B2 ···, Bk, and P (B1) + P (B2) + ·· + P (Bk) = 1 and an observable event A.then:

P(Bi|A)=P(Bi∩A)/P(A)=P(Bi)P(A|Bi)/P(B1)P(A|B1)+P(B2)P(A|B2)+···+P(Bk)P(A|Bk)

Base on above formula.

When only have two class:

If p1 (x, y)> p2 (x, y), then it is classified into class 1

If p1 (x, y)< p2 (x, y), then it is classified into class 2

Combine with Bayes’s Rule :

x, y represents the characteristic variable, ci represents the classification, p (ci | x, y) represents the probability of being classified into the class ci under the x, y. Therefore, combining conditional probability and Bayes' theorem,then:

If p (c1 | x, y)> p (c2 | x, y), then the class belongs to c1

If p (c1 | x, y)< p (c2 | x, y), then the class belongs to c2

1. Then in the text classification :

The vocabulary appearance is represented by word vector ω, which is composed of multiple numerical values, and the number of numerical values is the same as the number of vocabulary in the training set.

Therefore, the above Bayesian conditional probability formula can be expressed as:

p(ci|ω)=p(ω|ci)p(ci)/p(ω)

1. The difference with BernoulliNB :

MultinomialNB is use the number of occurrences as the num of matrix, and BernoulliNB is use 1 and 0 as word used or not as the num of matrix which means ignore the number of occurrences.

1. Apply MultinomialNB in python

from sklearn.naive\_bayes import MultinomialNB clf = BernoulliNB()

clf = MultinomialNB()

clf.fit(trainX, trainY)

result = clf.predict(testX)

1. feature selection

same as before

1. hyper-parameter tuning

alpha : Floating-point type, optional, default 1.0, add Laplace repair / Lidstone smoothing parameters

fit\_prior : Boolean, optional, default True, indicates whether to learn a priori probability,Parameter False means that all class labels have the same prior probability

set fit\_prior= False , alpha change from 0.1 to 2 ,the trend of total acc change is increase then decrease. And the peak point is show in alpha=0.6.

set fit\_prior=True , alpha change from 0.1 to 2 ,the trend of total acc change is decrease. And the peak point is show in alpha=0.1, and this model perform better then set fit\_prior= False.

So the best performance parameter set is alpha = 0.1, fit\_prior=True.

1. evaluation metrics

same as before