Semiconductor Optoelectronics
ISSN: 1001-5868

A STUDY OF THE PREDICTION OF THE THREE MODELS LOGISTIC REGRESSION, GAUSSIAN NAÏVE BAYES, AND MUTI-LAYER PERCEPTRON CLASSIFIER IN MACHINE LEARNING

Namkil Kang
Professor, Far East University, South Korea

The ultimate goal of this paper is to analyze the accuracy of three models in machine learning. More specifically, we made the three models Logistic Regression, Gaussian Naïve Bayes, and Multi-Layer Perceptron Classifier predict whether each person had a cold or not. This research was carried out by python. A point to note is that we trained the model Logistic Regression to predict whether each person had a cold or not. When we used grid search, the best parameter was 0.1 and the accuracy rate of the model Logistic Regression was 96%. When it comes to test data, its accuracy was 100%. This in turn suggests that this took place since we trained the model Logistic Regression. In the case of random search, the best parameter was 6 and the accuracy rate of the model Logistic Regression was 93.33%. More importantly, when in the case of test data, the parameter was 6, the best score was 100%. This took place since we trained the model Logistic Regression. A further point to note is that we trained the model Gaussian Naive Bayes to predict our data. Most importantly, when we used grid search and random search, the accuracy rate of the model Gaussian Naive Bayes was 100%. This in turn indicates that this model worked well for 100 sets of data. A major point of this paper is that we trained the model Multi-Layer Perceptron Classifier to predict whether each person had a cold or not. When we used grid search, the best parameter was 50 and the best score was 96%. However, in the case of test data, the accuracy rate of the model Multi-Layer Perceptron Classifier was 100%. It is clear from our findings that the three models worked well for our data, but the model Gaussian Naive Bayes worked best for them.

Keywords: machine learning, model, Logistic Regression, Gaussian NB, Multi-Layer Perceptron Classifier, accuracy

1. Introduction

The main purpose of this paper is to analyze the accuracy of three models in machine learning. More specifically, we make the three models Logistic Regression, Gaussian Naive Bayes, and Muti-Layer Perceptron Classifier predict whether each person has a cold or not. We trained the three models to predict whether each person had a cold or not. With respect to 100 data, it should be pointed out that we made them up ourselves. More specifically, we classified 100 data into 6 symptoms and subdivided them into a patient or no patient based upon 6 symptoms. To go into detail, a cold involves a fever, runny nose, sore throat, and cough, whereas it does not involve a back pain and headache. If a person (even though we made up data) has more than one of a fever, runny nose, sore throat, and cough, then he is a cold patient, whereas if a

person has a back pain or headache, then he is not a cold patient. We assigned 1 to a symptom or symptoms, whereas we assigned 0 to no symptom. For the three models to predict whether each person had a cold or not, we trained them and analyzed their prediction. First, we trained the model Logistic Regression to obtain its accuracy. We used grid search and random search. Also, we evaluated the model Logistic Regression by using the ROC (Receiver Operating Characteristic) analysis. Second, we trained the model Gaussian Naive Bayes to predict whether each person had a cold or not. We used the parameter var_smoothing to predict our data. Also, we used grid search and random search to predict our data. Third, we made the model Muti-Layer Perceptron Classifier predict whether each person was a cold patient or not. We used grid search to obtain its accuracy. Finally, we compare the three models and their accuracy and investigate which model works best for our data.

2. Results

2.1. The Model Logistic Regression

This section is devoted to investigating the accuracy of the model Logistic Regression with respect to 100 sets of data. We trained the model Logistic Regression to predict whether each person had a cold or not.

To begin with, let us examine the accuracy of the model Logistic Regression without considering any parameter. We imported the model Logistic Regression to investigate its accuracy with respect to 100 sets of data. Quite interestingly, with respect to train data, it is worthwhile pointing out that the accuracy of the Logistic Regression model is 98.66%. When it comes to this percentage, the accuracy of the model Logistic Regression is not bad. This in turn indicates that this model works properly for 100 sets of data.

Now let us probe into the accuracy of the model Logistic Regression by using grid search. The so-called grid search refers to setting parameters by a researcher. We tried to set parameters as follows: 0.001,0.01,0.1,1,10,100. More interestingly, the best parameter is 0.1 and the accuracy rate of the model Logistic Regression is 96%. When it comes to test data, its accuracy is 100%. This in turn suggests that this took place since we trained the model Logistic Regression. The split proportion of train data and test data is 70/30.

Now let us consider the accuracy rate of train data and test data:

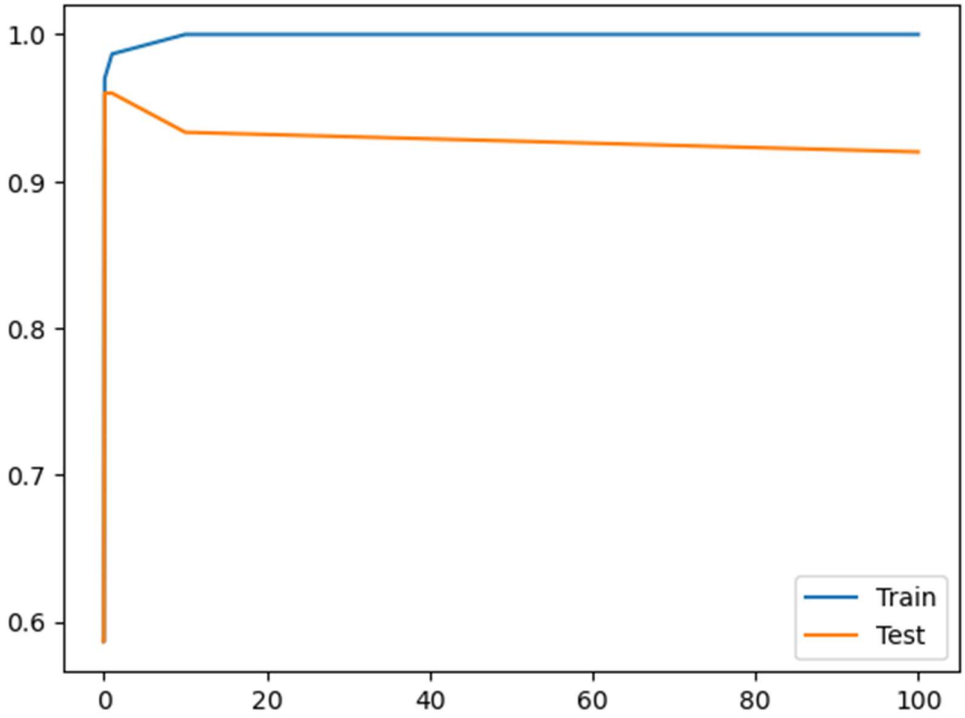

Figure 1 The Accuracy Rate of Train Data and Test Data

As can be seen from the graph, when in the case of train data, the parameter is more than 10, the accuracy rate of the model Logistic Regression is about 100%. As illustrated in Figure 1, when the parameter is less than 10, that of the model Logistic Regression is about 96%. Quite interestingly, when in the case of test data, the parameter is 0, the score is the highest. Quite interestingly, when the parameter increases, there is a gradual decline in the accuracy rate of the model Logistic Regression.

Now let us examine the accuracy rate of the model Logistic Regression by using random search. We imported the model Logistic Regression and trained it. The so-called random search refers to setting the scope of parameters. We set the scope of parameters from 1 to 100. To begin with, we imported the model Logistic Regression and trained it. Quite interestingly, the best parameter is 6 and the accuracy rate of the model Logistic Regression is 93.33%. More importantly, when in the case of test data, the parameter is 6, the score is 100%. This happened since we trained the model Logistic Regression.

Now take a look at the following graph. Figure 2 shows the accuracy rate of train data and test data when we use random search:

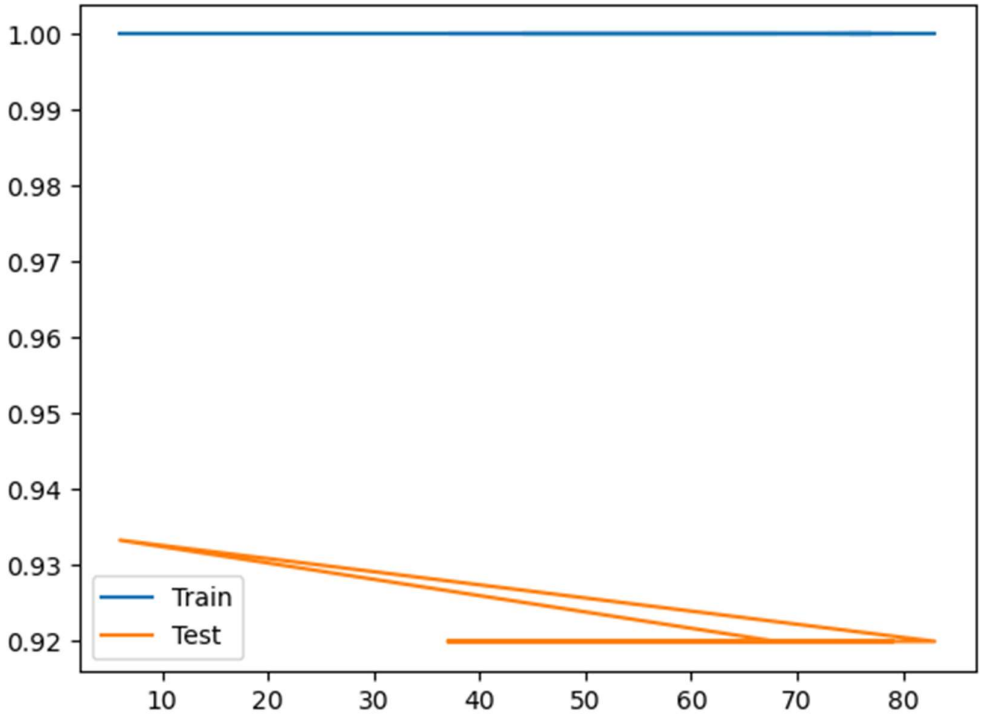

Figure 2 The Accuracy Rate of Train Data and Test Data

It is worth noting that when in the case of train data, the parameter is from 10 to 80, the accuracy rate of the model Logistic Regression is the same (100%). The accuracy rate is not influenced by the value of the parameters. It must be noted, however, that when in the case of test data, the figure of parameters increases, there is a steady fall in the accuracy rate of the model Logistic Regression. More interestingly, when the parameter is less than 10, the score is the highest (about 93%).

Now let us try to evaluate the model Logistic Regression. The so-called the ROC (Receiver Operating Characteristic) analysis leads us to evaluate the relevant model. When C is 10, we think of it as the best parameter. When the parameter is10, let us examine the accuracy rate of the model Logistic Regression. Most importantly, the accuracy rate of train data and test data is 100%, respectively. When it comes to train data, the model Logistic Regression predicted that 44 people were cold patients, whereas 31 people were not. The confusion rate of the model Logistic Regression is 0%. Let us take a look at Figure 3. As exemplified in Figure 3, the false positive rate refers to the error rate, whereas the true positive rate refers to the rate of the right prediction. Most importantly, the space between the false positive rate and the true positive rate is significant. The more the space between the false positive rate and the true positive rate is big, the relevant model is good. As illustrated in Figure 3, the space between the false positive rate and the true positive rate forms a big triangle shape. This in turn indicates that the model Logistic Regression is good enough and works properly for 100 sets of data.

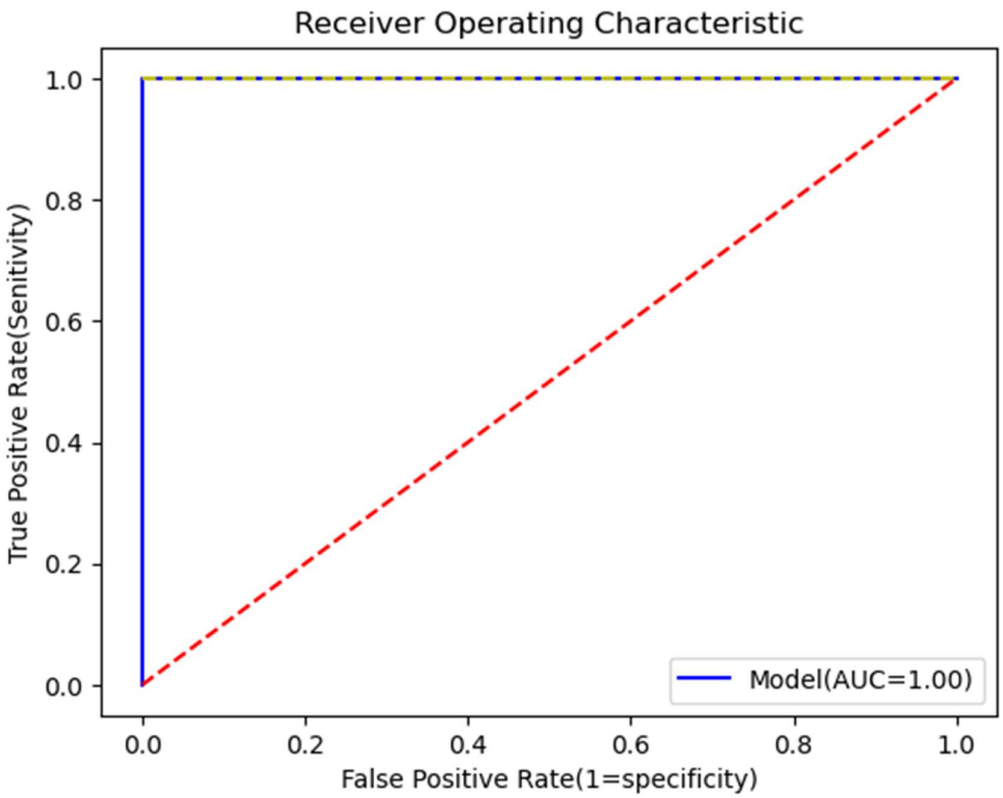

Figure 3 Receiver Operating Characteristic

2.2. The Model Gaussian Naive Bayes

This section focuses on investigating the accuracy rate of the classification model Gaussian Naive Bayes. We trained it to predict whether each person had a cold or not. To begin with, we investigate the accuracy of the model Gaussian Naive Bayes without considering grid search and random search. We imported the model Gaussian NB and trained it to predict whether each person had a cold or not. Quite interestingly, the model Gaussian NB predicted that in the case of train data, 44 people had a cold, whereas 31 did not. The classification report (Table 1) shows this fact:

Table 1 Classification Report

	Precision	**Recall**	**F1-score**	**Support**
0	1.00	1.00	1.00	31
1	1.00	1.00	1.00	44
Accuracy			1.00	75
Macro Avg	1.00	1.00	1.00	75
Weighted Avg	1.00	1.00	1.00	75

As exemplified in Table 1, 0 refers to no patient. 1 refers to a cold patient. The term accuracy indicates that the relevant model judges true as true or judges false as false. On the other hand, the term recall indicates that the relevant model judges true as true. The term precision refers to the proportion of being true in what the relevant model judges as true. Finally, the term support refers to the number of train data or test data. More interestingly, as exemplified in

Table 1, all figures are 1.00, which in turn indicates 100%. Most importantly, when it comes to train data, the accuracy rate of the model Gaussian NB is 100%.

It must be stressed that in the case of test data, the accuracy rate of the model Gaussian NB is 100%. More specifically, the model Gaussian NB predicted that 14 people had a cold, whereas 11 people did not.

Now attention is paid to grid search. To begin with, we imported Grid Search CV and trained the model Gaussian NB to predict whether each person had a cold. Our parameter, namely var_smoothing is 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10. More interestingly, the best parameter is 1, whereas the best score is 100%. On the other hand, in the case of test data, the best score is also 100%, which in turn indicates that the best accuracy of the model Gaussian NB is 100%.

Now attention is paid to random search. We imported the model Gaussian NB and RandomizedSearchCV. The parameter is from 0 to 20. It is worth noting that the best parameter is 11 and that in the case of train data, the best score is 100%. Quite interestingly, in the case of test data, the best accuracy rate of the model Gaussian NB is also 100%. This leads us to conclude that the model Gaussian NB works well for 100 sets of data. From this, it is clear that the model Gaussian NB is good enough.

2.3. The Model Muti-Layer Perceptron Classifier

This section is focused on probing into the accuracy rate of the model Muti-Layer Perceptron Classifier. We trained it to predict whether each person had a cold or not. To begin with, we inquire into the accuracy rate of the model Multi-Layer Perceptron Classifier without considering grid search. After this, we use grid search to see the model Muti-Layer Perceptron Classifier work.

It must be noted that the split proportion of train data and test data is 70/30. We imported the MLP Classifier to see it predict our data. Most importantly, the accuracy rate of the model MLP Classifier is100%. Quite interestingly, the model MLP Classifier predicted that 44 had a cold, whereas 31 did not. Exactly the same can be said of test data. The accuracy rate of the model MLP Classifier is 100%. The following table shows this fact:

Table 2 Classification Report (Test Data)

	Precision	**Recall**	**F1-score**	**Support**
0	1.00	1.00	1.00	11
1	1.00	1.00	1.00	14
Accuracy			1.00	25
Macro Avg	1.00	1.00	1.00	25
Weighted Avg	1.00	1.00	1.00	25

As indicated in Table 2, our model judged 11 people as no patients, whereas it judged 14 people as patients, which is all correct. On the other hand, the accuracy rate of the model MLP Classifier is 100%. That is to say, the model MLP Classifier judged true as true or judged false as false.

Finally, let us turn our attention to grid search. We trained the model MLP Classifier to predict whether each person had a cold by using grid search. Our parameter is 10, 30, 50, and 100. Just

as in the case of train data, test data shows 100%. That is to say, in the case of test data, the accuracy rate of the model MLP Classifier is 100%. The best parameter is 50, which in turn indicates that hidden layer sizes are 50. It is worthwhile noting that the best score is 96%, whereas the test set score is 100%. It is clear from our findings that the model MLP Classifier work well for 100 sets of data. We thus conclude that it is good enough.

2.4. The Evaluation of 3 Models and Their Accuracy

In what follows, we aim to evaluate the three models on the basis of their accuracy. To begin with, without using grid search and random search, the model Logistic Regression worked properly for 100 sets of data. To be more specific, when it comes to train data, the accuracy rate of the model Logistic Regression is 98.66%. When we use grid search, things are different. More specifically, the best parameter is 0.1 and the accuracy rate of the model Logistic Regression is 96%. When it comes to test data, its accuracy rate is 100%. When we used random search, we obtained different results. That is to say, when it comes to train data, the accuracy rate of the model Logistic Regression is 93.33, whereas talking about test data, that of the model Logistic Regression is 100%.

Now attention is paid to the model Gaussian NB. When it comes to train data and test data, the accuracy rate of the model Gaussian NB is 100%, respectively. In the case of test data, the model Gaussian NB predicted that 14 people had a cold, whereas 11 did not. When we used grid search, just as in the case of train data, test data show 100% in the accuracy rate of the model Gaussian NB. Exactly the same can be said about random search. When it comes to train data and test data, the accuracy rate of the model Gaussian NB is also 100%.

Finally, let us turn our attention to the model MLP Classifier. Its accuracy is 100% without using grid search. That is to say, the model MLP Classifier predicted that 44 had a cold, whereas 31 did not. However, things are different when we use grid search. The best parameter is 50 and the best score is 96%. Note, however, that the test set score is 100%.

To sum up, the model Gaussian NB works best for 100 sets of data and the model MLP Classifier follows it. The model Logistic Regression works properly, but it ranks third, compared to the other models. It is clear from our findings that the model Gaussian Naive Bayes worked best for our data. For big data and machine learning, see Kang (2023a, 2023b, 2023c, 2023d, 2023e, 2023f, 2024a, 2024b).

3. Conclusion

To sum up, we have analyzed the accuracy of three models in machine learning. More specifically, we made the three models Logistic Regression, Gaussian Naive Bayes, and Muti-Layer Perceptron Classifier predict whether each person had a cold or not. In section 2.1, we trained the model Logistic Regression to predict whether each person had a cold or not. When we used grid search, the best parameter was 0.1 and the accuracy rate of the model Logistic Regression was 96%. When it comes to test data, its accuracy was 100%. This in turn suggests that this took place since we trained the model Logistic Regression. In the case of random search, the best parameter was 6 and the accuracy rate of the model Logistic Regression was 93.33%. More importantly, when in the case of test data, the parameter was 6, the score was 100%. This happened since we trained the model Logistic Regression. In section 2.2, we trained the model Gaussian Naive Bayes to predict our data. Most importantly, when we used

grid search and random search, the accuracy rate of the model Gaussian Naive Bayes was 100%. This in turn indicates that this model worked well for 100 sets of data. In section 2.3, we trained the model Multi-Layer Perceptron Classifier to predict whether each person had a cold or not. When we used grid search, the best parameter was 50 and the best score was 96%. However, in the case of test data, the accuracy rate of the model Multi-Layer Perceptron Classifier was 100%. It is clear from our findings that the three models worked well for our data, but the model Gaussian Naive Bayes worked best for them.

References

[1] Kang, N. (2023a). K-Pop in BBC News: A Big Data Analysis. *Advances in Social Sciences Research Journal 10*(2), 156-169.

[2] Kang, N. (2023b). K-Dramas in Google: A NetMiner Analysis. *Transaction on Engineering and Computing Sciences 11*(1), 193-216.

[3] Kang, N. (2023c). A Comparative Analysis of Tolerate and Put up with in the COCA. *Semiconductor and optoelectronics* 42(1): 1468-1476.

[4] Kang, N. (2023d). Sure of and Sure about in Corpora and ChatGPT. *Journal of Harbin Engineering University* 44(7): 1347-1351.

[5] Kang, N. (2023e). Turn out adj and Turn out to be adj in the Now Corpus and ChatGPT. *Journal of Harbin Engineering University* 44(8): 825-831.

[6] Kang, N. (2023f). Care for and Like in Corpora and ChatGPT. *Semiconductor and optoelectronics* 42(2): 188-198.

[7] Kang, N. (2024a). A Big Data Analysis of a Hot Political Issue. Studies in Linguistics 70: 149-165

[8] Kang, N. (2024b). A study of the Prediction of the Model Logistic Regression in Machine Learning : Focusing on a Survey.

Appendix

Data

Number	cough	fever	runny nose	sore throat	headache	back pain	cold
1	0	0	1	0	0	1	1
2	1	0	0	0	0	0	1
3	0	0	0	0	0	1	0
4	0	0	0	0	0	1	0
5	0	1	0	0	0	0	1
6	1	1	1	1	1	0	1
7	0	0	0	0	0	1	0
8	1	1	1	0	0	1	1
9	0	0	0	0	0	0	0
Number	**cough**	**fever**	**runny nose**	**sore throat**	**headache**	**back pain**	**cold**
10	0	0	0	1	0	0	1

11	1	1	0	0	1	1	1
12	0	0	1	1	0	1	1
13	0	0	0	0	0	1	0
14	0	0	0	0	0	0	0
15	0	0	0	0	0	1	0
16	0	0	0	0	0	0	0
17	0	0	0	0	0	1	0
18	0	0	0	0	0	1	0
19	1	1	1	1	1	1	1
20	1	1	0	0	0	0	1
21	0	0	0	0	1	0	0
22	1	1	0	0	0	0	1
23	1	1	1	1	0	0	1
24	1	0	0	0	0	0	1
25	0	0	0	0	1	1	0
26	0	0	0	0	1	1	0
27	0	0	1	1	0	0	1
28	1	1	0	0	0	0	1
29	0	0	0	0	1	1	0
30	1	1	1	1	0	0	1
31	0	0	1	1	0	0	1
32	0	0	0	0	0	1	0
33	0	0	0	0	1	1	0
34	1	1	1	0	0	0	1
35	0	0	0	1	0	0	1
36	1	1	1	0	0	0	1
37	0	0	0	1	1	1	1
38	0	1	1	1	0	0	1
39	0	0	0	0	0	1	0
40	1	1	1	0	0	1	1
41	0	0	1	1	0	0	1
42	0	0	1	1	0	0	1
43	0	0	0	0	1	1	0
44	1	0	0	1	0	0	1
45	1	1	1	0	1	1	1
46	0	0	0	0	0	0	0
47	1	1	1	0	0	0	1
48	0	0	1	1	0	0	1
49	1	1	0	0	1	1	1
Number	**cough**	**fever**	**runny nose**	**sore throat**	**headache**	**back pain**	**cold**
50	1	1	1	1	0	0	1

51	0	0	1	1	1	1	1
52	1	1	1	0	1	1	1
53	0	0	0	0	1	1	0
54	1	1	1	0	0	0	1
55	0	0	0	0	0	1	0
56	1	1	1	1	1	1	1
57	0	0	0	0	0	1	0
58	1	1	1	1	1	0	1
59	0	0	1	0	0	0	1
60	1	1	1	0	0	0	1
61	0	0	0	0	1	0	0
62	1	1	1	1	1	1	1
63	0	0	0	0	0	1	0
64	1	1	1	1	0	0	1
65	0	0	0	0	0	1	0
66	0	0	0	0	1	1	0
67	1	1	1	1	1	1	1
68	0	0	0	0	1	1	0
69	1	1	1	1	0	0	1
70	0	0	0	0	1	0	0
71	0	0	0	0	0	0	0
72	1	1	1	0	0	0	1
73	0	0	0	0	0	0	0
74	1	1	1	1	0	0	1
75	0	0	0	0	0	1	0
76	1	1	1	1	0	0	1
77	0	0	0	0	1	1	0
78	0	0	0	0	0	0	0
79	1	1	0	0	0	0	1
80	1	1	1	1	0	0	1
81	0	0	0	1	1	1	1
82	0	0	0	0	0	0	0
83	0	0	0	0	1	0	0
84	0	0	0	0	0	1	0
85	1	1	1	1	0	0	1
86	0	0	0	0	0	1	0
87	1	1	0	0	0	0	1
88	0	0	0	1	0	0	1
89	1	1	0	0	0	0	1
Number	**cough**	**fever**	**runny nose**	**sore throat**	**headache**	**back pain**	**cold**
90	1	1	1	1	0	0	1

91	0	0	0	0	1	1	0
92	1	1	1	1	1	1	1
93	0	0	0	0	0	1	0
94	1	1	1	1	0	0	1
95	0	0	0	0	1	1	0
96	1	1	1	1	1	1	1
97	0	0	0	0	0	1	0
98	1	1	1	0	0	0	1
99	0	0	0	0	1	0	0
100	1	1	1	1	0	0	1