Advanced ML assignment 2

Md Mahim Anjum Haque PID(mahim)

March 2022

1 Problem 1

1.1 a

The parameters that are learned in Naive Bayes are the prior probabilities of different classes, as well as the likelihood of different features for each class. We know for Naive Bayes assumption $P(c|x) \propto P(c)P(x_1|c)P(x_2|c)...p(x_n|c)$ where c is a class and x is a test sample. All quantities P(c) and P(xi—c) are parameters which are determined during training and are used during testing.
For this specific example for each of the 2 size attributes we need to calculate 2 parameters each like $p(size = x|C)$ so 4 parameters.
Similarly for color 4 parameters
Similarly for Shape 4 parameters
And finally 2 prior probabilities
so in total 14. Which is 2*(2n+1) parameters where n is the number of binary features and the outcome is also binary. But all those 14 parameters are not actually independent. Exactly half of those parameter values can be calculated from the other halves.

1.2 b

		Good Apple	
		Yes	No
Size	Small	1/4	3/6
	Large	3/4	3/6

		Good Apple	
		Yes	No
Color	Green	0/4	5/6
	Red	4/4	1/6

		Good Apple	
		Yes	No
Shape	Irregular	1/4	4/6
	Circle	3/4	2/6

We also need to calculate the Prior probabilities. P(Yes) = 4/10 , p(No) = 6/10

1.3 c

We can see that using the above parameters we will face zero probability problem for p(green—yes) so we need to use Laplacian correction. Here is the updated tables using laplacian correction.

		Good Apple	
		Yes	No
Size	Small	1+1/4+2	3+1/6+2
	Large	3+1/4+2	3+1/6+2

		Good Apple	
		Yes	No
Color	Green	0+1/4+2	5+1/6+2
	Red	4+1/4+2	1+1/6+2

		Good Apple	
		Yes	No
Shape	Irregular	1+1/4+2	4+1/6+2
	Circle	3+1/4+2	2+1/6+2

So we know,

$$p(y = No|Small, Red, Circle) \propto p(No) * p(Small|No) * p(Red|No) * P(Circle|No)$$

$$= (6/10) * (4/8) * (2/8) * (3/8) = 9/320 = 0.028$$

$$p(y = Yes|Small, Red, Circle) \propto p(Yes) * p(Small|Yes) * p(Red|Yes) * P(Circle|Yes)$$

$$= (4/10) * (2/6) * (5/6) * (4/6) = 2/27 = 0.074$$

The normalized probablilities = $p(y = No|X) = 0.028/(0.028 + 0.074) = 0.2745$
The normalized probablilities = $p(y = Yes|X) = 1 - (p(y = No|X)) = 1 - 0.2745 = 0.7255$

so for this apple model with predict yes with 72.55 % probability.

2 Problem 2

2.1 1

For this specific case the objective function and constraints will he Such that $||w||^2$ is minimized w.r.t.

$$\underset{w,b}{\text{argmax}}\, y_j(W^T x_j + b) + y_k(W^T x_k + b)$$

$$W^T x_j + b - W^T x_k - b[y_j = 1, y_k = -1]$$

$$W^T x_j - W^T x_k$$

2.2 2

If the points can not separated by margins hard SVM wont work because it wont be able to find chose separating syperplanes. So we can use soft SVM to allow some points to be misclassified so that we can penalize the model accordingly and keep a slack variable to tune how much we are going to punish for the misclassifications. We can introduce a variable C C (¿ 0): Controls the trade-off between the penalty and the margin.
– Smaller C: allow more mistake
– Larger C: allow less mistake
this is known as soft SVM.

2.3 3

In it's most simple type, SVM doesn't support multiclass classification natively. But It supports binary classification and separating data points into two classes. For multiclass classification, the same principle is utilized after breaking down the multiclassification problem into multiple binary classification problems.

The idea is to map data points to high dimensional space to gain mutual linear separation between every two classes. This is called a One-to-One approach, which breaks down the multiclass problem into multiple binary classification problems. A binary classifier per each pair of classes.

Another approach one can use is One-to-Rest. In that approach, the breakdown is set to a binary classifier per each class.

A single SVM does binary classification and can differentiate between two classes. In the One-to-Rest approach, the classifier can use 'm' SVMs. Each SVM would predict membership in one of the classes. In the One-to-One approach, the classifier can use m*(m-1)/2 SVMs for creating a model from each class with all other.

3 Problem 3

3.1 a

(1) Accuracy = TP+TN/ALL = 18+174/222 = 0.8648
(2) Error rate = 1-Accuracy = 0.1351
(3) Precision = TP/TP+FP = 18/18+7 = 0.72
(4) Recall = TP/TP+FN = 18/18+23 = 0.439
(5) F1 = 2*precision*recall/(precision+recall) = 0.5454

3.2 b

By looking at the evaluation matrix we can see that the model has high precision and low recall. Also the data is heavily skewed. If the main concern of this classifier is to make sure that most relevant things are retrieved (needs high recall) than this is a terrible classifier because it has a very low recall. If we want to just classify as many data points correctly as possible than by high precision this is a pretty good classifier.

4 Problem 4

4.1 a

Despite the sigmoid function adding non linearity logistic regression is basically a model which separates the data points by a straight line or plane. It's similar to linear regression except the extra non linear activation function layer which makes it possible to squish the outputs from real values to [0, 1]. So for this given figure LR model will be able to perfectly classify the data points.
But if we change the point of (1, 2) than there exists no such possible line that separates the data points correctly so logistic regression wont be able to classify perfectly for the changed figure.

4.2 b

For this figure I think logistic regression is better because as we can see the data neither follow any linear trend nor the data has many real valued outputs. It's a task of classification not regression. So I think logistic regression is better for this figure.

4.3 c

Logistic regression is another generalized linear model (GLM) procedure using the same basic formula as the linear regression, but instead of the continuous Y it uses an activatin layer in the end to map the continues values into a class. The Support Vector Machines algorithm on the otherhand is much more geometrically motivated. Instead of assuming a probabilistic model, we're trying to find a particular optimal separating hyperplanes, where we define "optimality" in the context of the support vectors. But linear regression just finds a single separating hyperplane based on statistics not geometry. Based on pros and cons here are some differences.

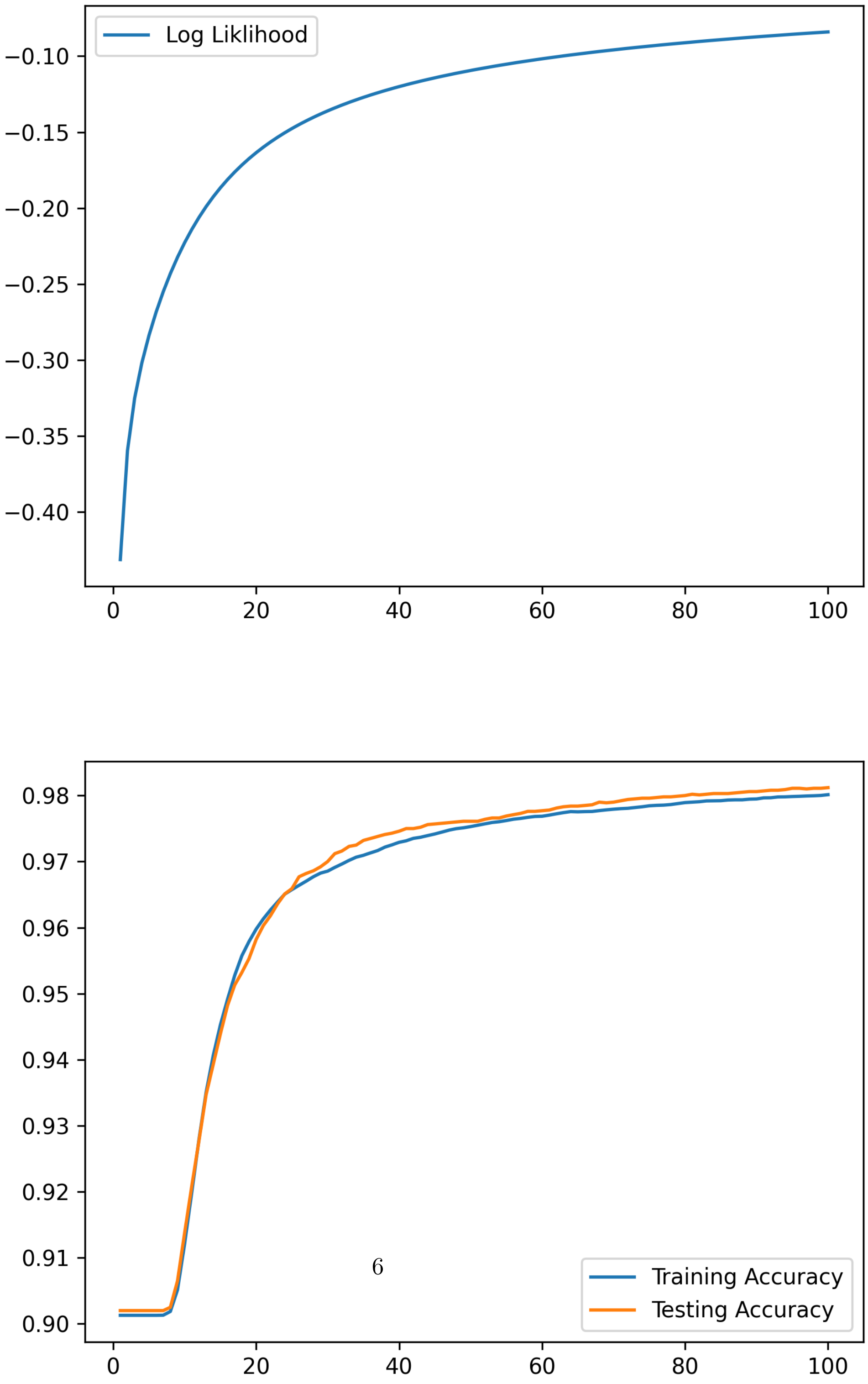
- SVM works well with unstructured and semi-structured data like text and images while logistic regression works with already identified independent variables.
- SVM is based on geometrical properties of the data while logistic regression is based on statistical approaches.

- The risk of overfitting is less in SVM, while Logistic regression is vulnerable to overfitting.

4.4 d

Here we can see the graphs on iteration(1-100) on X-axis and Log liklihood on the Y axis of the first graph and train-test accuracy on y axis of the second graph. We can see that the log liklihood which basically means the overall liklihood probability of the data is increasing. And as it's proof we can also see both training and testing accuracy is increasing as well. And with improving training accuracy, testing accuracy is not decreasing which means the model is not overfitting as well till 100 iterations.

Figure 1: Log liklihood and accuracy vs iteration


Log Liklihood
Training Accuracy
Testing Accuracy