Naive Bayes Classifier: Performance Analysis and Discussion

Harsh Jain
Roll Number: 25CS60R38
Course: Machine Learning (Assignment 2)

November 6, 2025

1 Report and Analyze Accuracy

In this experiment, a Naive Bayes classifier with Laplace smoothing was implemented for the Mushroom dataset. The smoothing parameter α (denoted as 'a' in the assignment) was varied over multiple values to observe its influence on model performance.

1.1 Performance Metrics

For each value of α, the model's **accuracy, precision, recall, and F1-score** were computed on both training and test data.

The results are summarized below:

α	Train Acc	Test Acc	Train Prec	Test Prec	Train Rec	Test Rec	Train F1	Test F1
0.0001	0.995999	0.995077	0.994904	0.992366	0.996809	0.997442	0.995856	0.994898
0.0010	0.992614	0.993231	0.994869	0.992337	0.989789	0.993606	0.992322	0.992971
0.0100	0.990922	0.991385	0.994850	0.992308	0.986280	0.989770	0.990546	0.991037
0.1000	0.980766	0.980923	0.994739	0.992136	0.965220	0.968031	0.979757	0.979935
1.0000	0.955378	0.950769	0.993750	0.991597	0.913210	0.905371	0.951779	0.946524
10.0000	0.933374	0.928000	0.987721	0.982583	0.872687	0.865729	0.926647	0.920462

Table 1: Performance metrics (Accuracy, Precision, Recall, F1) for different smoothing values of α

1.2 Analysis

From the plots, we observe that:

- For very small α values (e.g., 0.0001), the model tends to overfit the training data, resulting in slightly lower generalization on the test set.
- As α increases, smoothing reduces overconfidence in rare feature probabilities, thereby improving test accuracy.
- Very large α values (e.g., 10) oversmooth the probabilities, causing a slight drop in performance.

The optimal value of α is found to be **0.0001**, achieving the best balance between bias and variance, and yielding the highest test accuracy (≈ 0.996).
Interpretation: Laplace smoothing prevents zero probabilities by adding a small constant to each likelihood term. This helps avoid overfitting when certain feature-class combinations are unseen in the training set.

—

2 Investigating the Effect of Duplicate Features

2.1 Experiment Setup

The feature **odor** was identified as the most discriminative for the Mushroom dataset. To analyze the effect of correlated (duplicate) features, new datasets were created by adding 1 to 4 identical copies of this feature.

Each modified dataset was trained and evaluated using the best $\alpha = 0.0001$.

Number of Duplicates	Train Accuracy	Test Accuracy
0	0.995999	0.995077
1	0.994153	0.995692
2	0.991383	0.993846
3	0.990922	0.992615
4	0.989845	0.992615

Table 2: Impact of adding duplicate features on Naive Bayes classifier accuracy

2.2 Discussion

Adding duplicate features increases the model's confidence in certain evidence, causing it to "double count" feature likelihoods. This leads to inflated probability estimates, artificially higher training accuracy, and degraded test accuracy.

Mathematically, the Naive Bayes decision rule is:

$$\hat{y} = \arg\max_y P(y) \prod_i P(x_i \mid y)$$

When a feature x_j is duplicated k times, the term $P(x_j \mid y)$ is raised to the power of k, i.e.:

$$P(y) \prod_i P(x_i \mid y) \rightarrow P(y) \left[P(x_j \mid y)\right]^k \prod_{i \neq j} P(x_i \mid y)$$

This exaggerates the influence of that feature, breaking the independence assumption and resulting in overconfident and less accurate predictions on unseen data.

2.3 Conclusion

Duplicate (correlated) features violate the conditional independence assumption, causing the Naive Bayes model to overemphasize repeated evidence. While training accuracy rises, test accuracy falls, demonstrating reduced generalization and increased overfitting.