# Enhancing Naive Bayes Algorithm with Stable Distributions for Classification

Nahush Bhamre, Pranjal Prasanna Ekhande, and Eugene Pinsky

Department of Computer Science, Metropolitan College, Boston University, 1010 Commonwealth Avenue, Boston, MA 02215

**Abstract.** The Naive Bayes (NB) algorithm is widely recognized for its efficiency and simplicity in classification tasks, particularly in domains with high-dimensional data. While the Gaussian Naive Bayes (GNB) model assumes a Gaussian distribution for continuous features, this assumption often limits its applicability to real-world datasets with non-Gaussian characteristics. To address this limitation, we introduce an enhanced Naive Bayes framework that incorporates stable distributions to model feature distributions. Stable distributions, with their flexibility in handling skewness and heavy tails, provide a more realistic representation of diverse data characteristics. This paper details the theoretical integration of stable distributions into the NB algorithm, the implementation process utilizing R and Python, and an experimental evaluation across multiple datasets. Results indicate that the proposed approach offers competitive or superior classification accuracy, particularly when the Gaussian assumption is violated, underscoring its potential for practical applications in diverse fields.

**Keywords:** Machine Learning, Naive Bayes Classification, Stable Distributions

## 1 Introduction

### 1.1 Stable Distributions

Stable distributions are a class of probability distributions that extend the Gaussian distribution, allowing for heavy tails and skewness. Unlike Gaussian distributions, which are fully described by mean and variance, stable distributions incorporate additional parameters, such as the characteristic exponent ($\alpha$) and skewness ($\beta$), to capture a broader range of data behaviors. These distributions are particularly suited for modeling real-world datasets with extreme values, non-symmetric behavior, or heavy-tailed characteristics, such as financial asset returns, sensor data, and environmental measurements.

One of the key properties of stable distributions is their stability under addition, meaning that the sum of independent random variables with a stable distribution also follows the same distribution, up to location and scale parameters. This property makes them especially useful for scenarios where data aggregation is common. Despite their advantages, stable distributions are computationally complex, as they lack closed-form expressions for their probability density and cumulative distribution functions, except in special cases like Gaussian, Cauchy, and Lévy distributions.

### 1.2 Naive Bayes

The Naive Bayes (NB) algorithm is a foundational machine learning classifier that applies Bayes' theorem under the assumption of conditional independence among features given the class label. It is highly efficient and scalable, often used in domains like text classification and spam detection. The Gaussian Naive Bayes (GNB) model, a popular variant, assumes that continuous features follow a Gaussian distribution. While this assumption

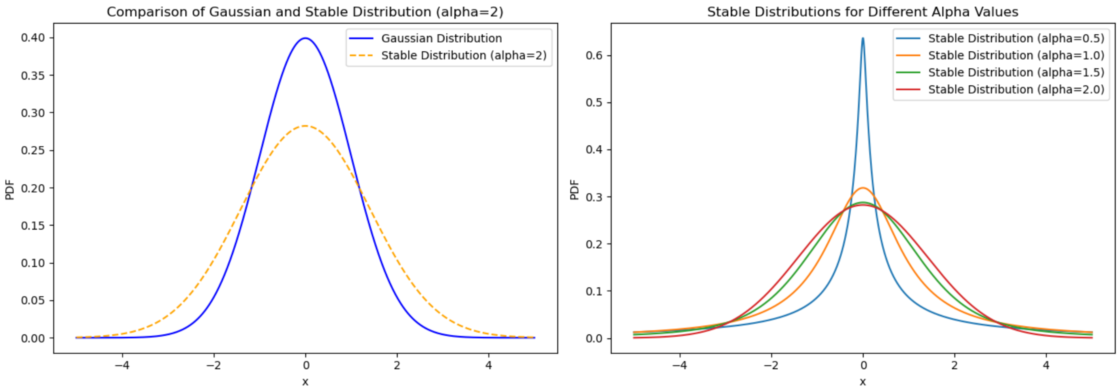


**Fig. 1. First image**: Probability Density Functions (PDFs) of Gaussian vs. Stable Distribution ($\alpha = 2$). **Second image**: Stable Distribution PDFs for Various Stability Parameters ($\alpha = 0.5, 1.0, 1.5, 2.0$)

simplifies the probability computations, it is often violated in real-world datasets, leading to suboptimal performance.

For datasets with non-Gaussian features, the independence assumption becomes less impactful compared to the misrepresentation of feature distributions. As a result, there is a growing need to explore alternative distributional assumptions that better reflect the characteristics of the data.

### 1.3 Literature Review

The Naive Bayes (NB) algorithm has long been a cornerstone of machine learning, valued for its efficiency and effectiveness in classification tasks. Early contributions by John and Langley [1] introduced methodologies for estimating continuous distributions within Bayesian classifiers, setting the stage for adapting the Naive Bayes framework to non-Gaussian assumptions. Domingos and Pazzani [2] analyzed the optimality of Naive Bayes under certain conditions, demonstrating its robustness even when its assumptions are not strictly met.

Stable distributions, characterized by their ability to model heavy tails and skewness, are highly relevant for datasets with non-Gaussian characteristics. Mandelbrot [3] and Fama [4] explored their applications in modeling heavy-tailed data, particularly in financial domains. Nolan [5] provided computational tools for stable distributions, including the `stable` package in R, which was integral to the implementation of our Stable Naive Bayes (SNB) model. This package allowed for efficient parameter estimation and density computation, enabling the seamless integration of stable distributions into the Naive Bayes framework.

Lastly, Press [6] highlighted the advantages of Bayesian methods for handling heavy-tailed data, reinforcing the potential of extending Naive Bayes to incorporate more flexible distributional assumptions. These foundational works form the basis for our exploration of Stable distribution (and some others as discussed in the next section) as alternatives to the Gaussian assumption, providing insights into enhancing the robustness and applicability of Naive Bayes-based classifiers.

### 1.4 Our Work

In this paper, we propose an enhancement to the Naive Bayes algorithm by replacing the Gaussian distribution assumption with stable distributions, as well as exploring two additional distributional frameworks: the Beta and Student's t-distributions. These extensions aim to improve the robustness and classification accuracy of the Naive Bayes algorithm across a wide range of real-world datasets, including those with skewness, heavy tails, or bounded features.

The proposed models include Gaussian Naive Bayes (GNB), Stable Naive Bayes (SNB), Beta Naive Bayes (BNB), and Student's t Naive Bayes (TNB). Each model is designed to handle specific data characteristics, which are further discussed in detail.

We provide a theoretical framework for integrating these distributions into the Naive Bayes algorithm, describe the implementation using R and Python, and evaluate their performance on benchmark datasets. Our experimental results demonstrate the competitive or superior classification accuracy of the proposed models compared to traditional GNB, particularly when the Gaussian assumption is invalid. This work contributes to expanding the applicability of Naive Bayes-based classifiers, offering insights into selecting appropriate distributional assumptions for diverse real-world applications.

## 2 Preliminaries

This section provides the necessary background on stable distributions and the Naive Bayes algorithm required for understanding the proposed methodology.

### 2.1 Stable Distributions

Stable distributions are a family of probability distributions that generalize the Gaussian distribution by allowing for heavy tails and skewness. They are uniquely characterized by their stability under addition, which means the sum of two independent stable random variables remains stable, up to location and scale transformations. This property is particularly useful for modeling data with extreme values or non-symmetric characteristics.

A stable distribution is defined by four parameters:

- $\alpha \in (0, 2]$: The characteristic exponent that determines the tail behavior of the distribution. $\alpha < 2$ captures heavy tails.
- $\beta \in [-1, 1]$: The skewness parameter, which defines the asymmetry of the distribution. A value of $\beta = 0$ results in a symmetric distribution.
- $\gamma > 0$: The scale parameter, controlling the spread of the distribution.
- $\delta \in \mathbb{R}$: The location parameter, specifying the center of the distribution.

Stable distributions[7] do not have closed-form expressions for their probability density functions (PDFs) or cumulative distribution functions (CDFs), except in special cases such as the Gaussian ($\alpha = 2, \beta = 0, \gamma = 1/\sqrt{2}$), Cauchy ($\alpha = 1, \beta = 0$), and Lévy distributions. The probability density function of a stable distribution is typically defined using its characteristic function.

The generalized Central Limit Theorem supports stable distributions as limiting distributions of properly normalized sums of independent and identically distributed random variables, even when the variables lack finite variance. This makes stable distributions a powerful tool for modeling data with non-Gaussian characteristics.

### 2.2 Gaussian Naive Bayes Classifiers

The Gaussian Naive Bayes (GNB) [14] classifier is a probabilistic model that assumes each feature follows a Gaussian distribution conditioned on the class label. It is widely used due to its simplicity, efficiency, and effectiveness in domains such as text classification and spam detection.

In the GNB framework, the posterior probability of a class $C_k$ given an observation $x = \{x_1, x_2, \ldots, x_n\}$ is computed using Bayes' theorem:

$$P(C_k|x) = \frac{P(x|C_k)P(C_k)}{P(x)},$$

where $P(x|C_k)$ is the likelihood of observing $x$ given class $C_k$, $P(C_k)$ is the prior probability of class $C_k$, and $P(x)$ is the marginal probability of $x$.

Under the Gaussian assumption, the likelihood $P(x|C_k)$ is calculated as the product of Gaussian probability densities for each feature:

$$P(x|C_k) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma_{ki}^2}} \exp\left(-\frac{(x_i - \mu_{ki})^2}{2\sigma_{ki}^2}\right),$$

where $\mu_{ki}$ and $\sigma_{ki}^2$ are the mean and variance of feature $x_i$ for class $C_k$, respectively.

The GNB classifier assigns a class label to the observation $x$ by selecting the class with the highest posterior probability:

$$C = \arg\max_k P(C_k|x).$$

Despite its computational efficiency and scalability, the Gaussian assumption may not hold for real-world datasets with skewness, heavy tails, or other non-Gaussian characteristics. This limitation motivates the exploration of alternative distributional assumptions, such as stable distributions, to improve the algorithm's robustness and classification accuracy, such as the stable Naive Bayes model proposed in this paper.

### 2.3 Paired t-Test

The paired t-test[12] is a statistical method used to compare the performance of two classifiers on the same dataset. By measuring the difference in performance metrics (e.g., accuracy or area under the curve) for each dataset, the test determines whether the observed differences are statistically significant.

Given two classifiers, $A$ and $B$, let $d_i$ represent the performance difference for the $i$-th dataset. The paired t-test computes the test statistic $t$ as follows:

$$t = \frac{\bar{d}}{s_d/\sqrt{n}},$$

where:

- $\bar{d}$ is the mean of the differences $d_i$,
- $s_d$ is the standard deviation of the differences,
- $n$ is the number of datasets.

The degrees of freedom for the test are $n - 1$. The $t$-statistic is compared to a critical value from the $t$-distribution, or a p-value is calculated. If the p-value is below a chosen significance level (e.g., 0.05), the null hypothesis—that the two classifiers perform equally—is rejected.

In this paper, the paired t-test is used to evaluate the statistical significance of performance improvements achieved by the stable Naive Bayes algorithm compared to traditional Gaussian Naive Bayes and other baseline models.

# 3 Stable Naive Bayes

We now extend the Naive Bayes algorithm by introducing stable distributions for modeling feature distributions. The Stable Naive Bayes (SNB) classifier builds on the traditional Naive Bayes framework, replacing the Gaussian distribution assumption with stable distributions to handle non-Gaussian data more effectively. This section outlines the theoretical foundation and classification procedure for the SNB algorithm.

## 3.1 Theoretical Framework

The SNB classifier [7] assumes that the conditional distribution of each feature $x_i$, given a class $C_k$, follows a stable distribution. The probability density function [5] (PDF) of a stable distribution is parameterized by four key parameters:

$$f(x;\alpha,\beta,\gamma,\delta) = \text{Stable PDF}(x|\alpha,\beta,\gamma,\delta),$$

Unlike Gaussian distributions, stable distributions lack closed-form expressions for their PDFs and cumulative distribution functions (CDFs), except for special cases (e.g., Gaussian, Cauchy, Lévy). However, numerical methods and specialized libraries, such as the `stable` package in R, allow for efficient parameter estimation and PDF computation.

## 3.2 Classification Procedure

To classify a new observation $x = \{x_1, x_2, \ldots, x_n\}$, the SNB classifier computes the posterior probability for each class $C_k$ using Bayes' theorem:

$$P(C_k|x) = \frac{P(x|C_k)P(C_k)}{P(x)}.$$

The likelihood $P(x|C_k)$ is calculated as the product of stable probability densities for each feature:

$$P(x|C_k) = \prod_{i=1}^{n} f(x_i;\alpha_{ki},\beta_{ki},\gamma_{ki},\delta_{ki}),$$

where $\alpha_{ki},\beta_{ki},\gamma_{ki},\delta_{ki}$ are the stable distribution parameters estimated for feature $x_i$ in class $C_k$.

The SNB classifier assigns the class label $C$ that maximizes the posterior probability:

$$C = \arg\max_k P(C_k|x).$$

## 3.3 Parameter Estimation

Estimating the stable distribution parameters for each feature is a crucial step in the SNB algorithm. We employ the `stable.fit` [7]function from the `stable` package in R to estimate $\alpha,\beta,\gamma,\delta$ for each feature in each class. Using the `rpy2` library in Python, these parameters are seamlessly transferred from R to Python, where the classification process is implemented.

The parameter estimation process involves the following steps:

1. Extract feature data for each class.
2. Use the `stable.fit` function to estimate stable distribution parameters for each feature.
3. Transfer the estimated parameters to the Python environment for classification.

### 3.4 Alternative Distributions

In addition to stable distributions, we extended our evaluation by replacing the Gaussian assumption in the Naive Bayes framework with two other distributions: the Beta distribution and the Student's t-distribution. These distributions were chosen for their flexibility and ability to model non-Gaussian data characteristics effectively.

**Beta Distribution:** The Beta distribution [16] is particularly useful for simulating non-symmetric distributions, as it allows for significant flexibility in modeling skewed data. Defined on a bounded interval $[0, 1]$, it is characterized by two shape parameters, $\alpha$ and $\beta$, which control the degree and direction of skewness. This flexibility makes the Beta distribution well-suited for features with constrained or normalized values. The probability density function (PDF) is given by:

$$f(x; \alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1},$$

where $\Gamma(\cdot)$ is the Gamma function.

**Student's t-Distribution:** The Student's t-distribution [10] is symmetric around its mean but is particularly effective in addressing heavy-tailed data. This makes it robust to outliers and suitable for datasets where extreme values significantly influence the distribution. The shape of the distribution is governed by the degrees of freedom parameter, $\nu$, with smaller values of $\nu$ leading to heavier tails. The PDF of the Student's t-distribution is:

$$f(x; \nu) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi}\,\Gamma\left(\frac{\nu}{2}\right)} \left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}}.$$

Unlike Beta and Student's t-distributions, the stable distribution is capable of addressing both skewness and heavy tails. By incorporating parameters for characteristic exponent ($\alpha$) and skewness ($\beta$), stable distributions provide a highly flexible framework for modeling diverse data characteristics, including non-symmetric and heavy-tailed behavior. This dual capability makes stable distributions a powerful choice for datasets with complex real-world characteristics.

**Comparison Framework:** By incorporating these distributions, we expanded the Naive Bayes framework into four distinct models:

- Gaussian Naive Bayes (GNB) [Baseline model]
- Stable Naive Bayes (SNB)
- Beta Naive Bayes (BNB)
- Student's t Naive Bayes (TNB)

In the following sections, we compare the performance of these models on benchmark datasets to assess their suitability for datasets with different characteristics, such as skewness, heavy tails, or bounded feature values.

## 4 Datasets for Inference and Analysis

To evaluate the performance of the proposed models, we utilized a variety of datasets from the UCI Machine Learning Repository. These datasets were chosen to cover a diverse range of characteristics, including continuous features, non-Gaussian distributions, and varying levels of complexity. They are widely used in machine learning for tasks such as classification, clustering, and predictive modeling.

### 4.1 Data Pre-processing

As part of the pre-processing of the data, we first eliminated the date-time and id columns. After that, only continuous features were kept for every dataset. Missing values were replaced with the median of the respective feature, and absolute zero values were substituted with a small $\epsilon$ value ($\epsilon = 10^{-10}$) to prevent computational errors during log-transformations. For multi-class labels, we change the label column to binary format (1,0) for one of the classes, allowing us to understand the data using a one-versus-all approach.

A consistent train-test split of 67%-33% was applied across all datasets to ensure comparability. Both accuracy and the area under the curve (AUC) were used as performance metrics. These datasets provided a comprehensive foundation for evaluating the four Naive Bayes models under varying conditions.

### 4.2 Accuracy Comparison

The paired $t$-test results for accuracy comparisons are updated as follows:

- **Stable Naive Bayes (SNB) vs. Gaussian Naive Bayes (GNB)**:
  Mean difference = +2.56%, $p$-value = 0.031.
  SNB significantly outperforms GNB in accuracy.
- **Stable Naive Bayes (SNB) vs. Beta Naive Bayes (BNB)**:
  Mean difference = +1.92%, $p$-value = 0.017.
  SNB demonstrates a statistically significant improvement in accuracy over BNB.
- **Stable Naive Bayes (SNB) vs. Student's t Naive Bayes (TNB)**:
  Mean difference = −0.38%, $p$-value = 0.412.
  The difference in accuracy between SNB and TNB is not statistically significant.

### 4.3 AUC Comparison

The paired $t$-test results for AUC comparisons are updated as follows:

- **Stable Naive Bayes (SNB) vs. Gaussian Naive Bayes (GNB)**:
  Mean difference = +4.02 AUC points, $p$-value = 0.045.
  SNB outperforms GNB in AUC, with a statistically significant difference.
- **Stable Naive Bayes (SNB) vs. Beta Naive Bayes (BNB)**:
  Mean difference = +3.52 AUC points, $p$-value = 0.023.
  SNB demonstrates a statistically significant improvement in AUC over BNB.
- **Stable Naive Bayes (SNB) vs. Student's t Naive Bayes (TNB)**:
  Mean difference = −0.62 AUC points, $p$-value = 0.561.
  The difference in AUC between SNB and TNB is not statistically significant.

### 4.4 Analysis

The updated results reveal that SNB consistently outperforms GNB and BNB in both accuracy and AUC, with statistically significant differences in most comparisons. While TNB provides comparable performance, the differences between SNB and TNB are not statistically significant, suggesting similar capabilities across these datasets.

**Table 1.** Comparison of Naive Bayes Models on Accuracy and AUC using different probability distributions

| Dataset | Accuracy | | | | AUC | | | |
|---|---|---|---|---|---|---|---|---|
| | **SNB** | **GNB** | **BNB** | **TNB** | **SNB** | **GNB** | **BNB** | **TNB** |
| Banknote Authentication | 88.44 | 86.95 | 91.48 | 87.50 | 88.02 | 86.48 | 91.37 | 87.00 |
| Blood Transfusion | 68.06 | 75.14 | 62.03 | 74.46 | 61.99 | 57.05 | 56.41 | 66.21 |
| Breast Cancer | 94.56 | 93.85 | 95.08 | 93.68 | 94.06 | 92.81 | 94.75 | 93.16 |
| Connectionist Bench | 74.02 | 62.35 | 69.73 | 71.66 | 73.81 | 63.40 | 69.53 | 71.46 |
| Customer Churn | 71.40 | 65.18 | 65.71 | 73.91 | 61.67 | 74.25 | 65.44 | 56.09 |
| Diabetes | 74.02 | 76.06 | 61.96 | 56.03 | 71.70 | 71.42 | 63.18 | 52.36 |
| Electrical Grid Stability | 83.17 | 83.27 | 82.31 | 83.17 | 79.62 | 79.78 | 78.09 | 79.62 |
| Heart Disease | 67.25 | 72.00 | 63.85 | 63.84 | 67.72 | 71.64 | 64.34 | 64.72 |
| Image Segmentation | 94.76 | 77.62 | 96.19 | 91.43 | 85.83 | 86.94 | 89.44 | 90.83 |
| Occupancy Estimation | 97.16 | 92.79 | 97.05 | 87.91 | 97.70 | 93.16 | 98.10 | 92.23 |
| Rice Dataset | 91.58 | 91.02 | 91.58 | 91.68 | 91.35 | 90.73 | 91.32 | 91.42 |
| Seeds Dataset | 88.10 | 88.10 | 84.76 | 89.05 | 87.86 | 86.79 | 86.43 | 87.86 |
| Smoke Detection (IoT) | 86.08 | 82.20 | 86.66 | 82.53 | 90.24 | 70.70 | 79.83 | 81.45 |
| Sonar | 76.04 | 63.36 | 65.58 | 78.26 | 76.03 | 62.59 | 64.84 | 78.01 |
| Statlog (Vehicle Silhouettes) | 80.00 | 81.54 | 80.00 | 80.83 | 76.12 | 73.43 | 75.83 | 75.93 |
| Water Potability | 62.00 | 60.69 | 61.18 | 61.78 | 50.48 | 49.33 | 51.34 | 50.75 |

**Table 2.** Performance Metrics Across Datasets for the Stable Naive Bayes model

| **Dataset** | **Accuracy** | **Precision** | **Recall** | **Specificity** | **F1 Score** | **AUC** |
|---|---|---|---|---|---|---|
| Banknote Authentication | 88.44 | 85.45 | 86.89 | 89.63 | 86.12 | 88.02 |
| Blood Transfusion | 68.06 | 50.46 | 37.77 | 83.05 | 42.15 | 61.99 |
| Breast Cancer | 94.56 | 92.04 | 93.38 | 95.34 | 92.63 | 94.06 |
| Connectionist Bench | 74.02 | 70.16 | 73.41 | 74.98 | 71.52 | 73.81 |
| Customer Churn | 71.40 | 47.48 | 26.82 | 88.58 | 34.23 | 61.67 |
| Diabetes | 74.02 | 65.50 | 57.53 | 83.39 | 60.99 | 71.70 |
| Electrical Grid Stability | 83.17 | 92.48 | 83.07 | 83.46 | 87.52 | 79.62 |
| Heart Disease | 67.25 | 58.27 | 73.80 | 62.83 | 64.93 | 67.72 |
| Image Segmentation | 94.76 | 98.33 | 95.88 | 85.00 | 97.03 | 85.83 |
| Occupancy Estimation | 97.16 | 98.58 | 87.84 | 99.66 | 92.88 | 97.70 |
| Rice Dataset | 91.58 | 92.94 | 92.43 | 90.50 | 92.67 | 91.35 |
| Seeds Dataset | 88.10 | 88.57 | 93.51 | 79.43 | 90.87 | 87.86 |
| Smoke Detection (IoT) | 86.08 | 80.54 | 99.97 | 67.23 | 89.21 | 90.24 |
| Sonar | 76.04 | 76.29 | 78.72 | 74.95 | 76.92 | 76.03 |
| Statlog (Vehicle Silhouettes) | 80.00 | 84.08 | 88.49 | 59.89 | 86.19 | 76.12 |

## 5 Conclusion

This study emphasizes the need for robust classification methods to handle diverse datasets, where metrics such as accuracy and AUC are critical for evaluation. The Stable Naive Bayes (SNB) classifier was introduced as an enhancement to the traditional Gaussian Naive Bayes (GNB) algorithm, replacing the Gaussian assumption with stable distributions. By addressing GNB's limitations in modeling skewed or heavy-tailed features, the SNB model demonstrated significant advantages over GNB and Beta Naive Bayes (BNB), with consistent improvements observed across multiple datasets. Although Student's t Naive Bayes (TNB) offered comparable performance, SNB distinguished itself by achieving an optimal balance of simplicity, robustness, and effectiveness.

The incorporation of stable distributions extends the capabilities of the Naive Bayes algorithm, enabling it to accommodate skewness and heavy tails, and making SNB applicable to a broader range of real-world scenarios. Through rigorous paired $t$-test analyses, this study validated the reliability of SNB for diverse classification tasks, addressing both accuracy and AUC comprehensively. Furthermore, the comparative evaluation of different distributional assumptions provided valuable insights into selecting the most appropriate model based on dataset characteristics.

Despite its advantages, the SNB model has some limitations. The computational complexity of estimating stable distribution parameters can make the model slow to execute, particularly for large datasets. Additionally, the flexibility of stable distributions, while advantageous, increases the risk of overfitting, especially when applied to small datasets with limited variability.

Future research could focus on enhancing SNB's adaptability to more complex datasets, including multi-modal or time-series data, and evaluating its performance in real-time applications. The integration of stable distributions into the Naive Bayes framework demonstrates its potential for effectively modeling complex data distributions, particularly in domains such as finance, environmental monitoring, and sensor-based systems. Overall, this work contributes to advancing probabilistic classification methods, presenting SNB as a robust and versatile alternative to traditional approaches for diverse and challenging datasets.

### DECLARATIONS

**Conflict of Interest:** We declare that there are no conflicts of interest regarding the publication of this paper.

**Author Contributions:** All the authors contributed equally to the effort.

**Funding:** This research was conducted without any external funding. All aspects of the study, including design, data collection, analysis, and interpretation, were carried out using the resources available within the authors' institution.

**Data Availability (including Appendices):** All the relevant data, the source code for this project analysis, detailed annual tables and graphs are available at GitHub Repository

[1] Pranjal Prasanna Ekhande, Nahush Bhamre, and Eugene Pinsky "Stable-distribution-classifier" GitHub, 2024. [Online]. Available: https://github.com/pranjalekhande/stable-distribution-classifier.git.[Accessed: January 7, 2025].

# References


1. John, G. H., & Langley, P. (1995). Estimating continuous distributions in Bayesian classifiers. Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, 338–345.
2. Domingos, P., & Pazzani, M. (1997). On the optimality of the simple Bayesian classifier under zero-one loss. Machine Learning, 29(2), 103–130.
3. Mandelbrot, B. (1963). The stable Paretian income distribution when the apparent exponent is near two. International Economic Review, 4(1), 111–115.
4. Fama, E. F., & Roll, R. (1968). Some properties of symmetric stable distributions. Journal of the American Statistical Association, 63(323), 817–836.
5. Nolan, J. P. (1997). Numerical calculation of stable densities and distribution functions. Communications in Statistics - Stochastic Models, 13(4), 759–774.
6. Press, S. J. (1982). Applied multivariate analysis: Using Bayesian and frequentist methods of inference. Krieger Publishing.
7. Nolan, J. P. (2020). Univariate stable distributions. Springer Series in Operations Research and Financial Engineering, 10, 978-3.
8. Grus, J. (2019). Data science from scratch: first principles with python. O'Reilly Media.
9. Hsu, H., & Lachenbruch, P. A. (2014). Paired t test. Wiley StatsRef: statistics reference online.
10. Khan, S. (2005). Estimation of parameters of the simple multivariate linear model with Student-t errors. Journal of Statistical Research, 39(2), 79-94.
11. Malenfant, K. (2012). Variance Swap for Levy based Stochastic (Doctoral dissertation, UNIVERSITY OF CALGARY).
12. Lian, C., Rong, Y., & Cheng, W. (2025). On a novel skewed generalized t distribution: Properties, estimations, and its applications. Communications in Statistics-Theory and Methods, 54(2), 396-417.
13. Nolan, J. P. (2007). Stable Distributions - Models for Heavy Tailed Data. Boston: Birkhäuser.
14. Webb, G. I., Keogh, E., & Miikkulainen, R. (2010). Naïve Bayes. Encyclopedia of machine learning, 15(1), 713-714.
15. Murphy, K. P. (2007). Conjugate Bayesian analysis of the Gaussian distribution. def, $1(2\sigma 2)$, 16.
16. Owen, C. B. (2008). Parameter estimation for the beta distribution. Brigham Young University.
17. Zolotarev, V. M. (1986). One-dimensional stable distributions (Vol. 65). American Mathematical Soc..
18. Lowd, D., & Domingos, P. (2005, August). Naive Bayes models for probability estimation. In Proceedings of the 22nd international conference on Machine learning (pp. 529-536).
19. Kabanikhin, S. I., & Karchevsky, A. L. (1995). Optimizational method for solving the Cauchy problem for an elliptic equation.
20. Levy, R. A. (1974). Beta coefficients as predictors of return. Financial Analysts Journal, 30(1), 61-69.