Supervised Learning: Naive Bayes

Gaussian, Multinomial, and Bernoulli Variants

Sarwan Ali

Department of Computer Science
Georgia State University

Understanding Naive Bayes

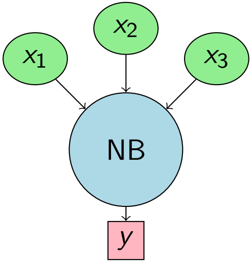
Today's Learning Journey

What is Naive Bayes?

Naive Bayes is a family of probabilistic classifiers based on Bayes' Theorem.

Key Characteristics:

- Simple yet powerful
- Fast training and prediction
- Works well with small datasets
- Handles multiple classes naturally
- "Naive" assumption of feature independence



Bayes' Theorem Refresher

Bayes' Theorem

$$P(y|x_1, x_2, \ldots, x_n) = \frac{P(x_1, x_2, \ldots, x_n|y) \cdot P(y)}{P(x_1, x_2, \ldots, x_n)} \quad (1)$$

Where:

- $P(y|x_1, x_2, \ldots, x_n)$ is the posterior probability
- $P(x_1, x_2, \ldots, x_n|y)$ is the likelihood
- $P(y)$ is the prior probability
- $P(x_1, x_2, \ldots, x_n)$ is the evidence

The Naive Assumption

Features are **conditionally independent** given the class:

$$P(x_1, x_2, \ldots, x_n|y) = \prod_{i=1}^{n} P(x_i|y)$$

Naive Bayes Classification

Classification Rule:

$$\hat{y} = \arg\max_{y} P(y) \prod_{i=1}^{n} P(x_i|y) \tag{2}$$

Steps for Classification:

1. Calculate prior probabilities $P(y)$ for each class
2. Calculate likelihoods $P(x_i|y)$ for each feature given each class
3. For a new instance, compute the posterior for each class
4. Assign the class with the highest posterior probability

Why drop the denominator?

Since $P(x_1, x_2, \ldots, x_n)$ is the same for all classes, we can ignore it for classification decisions.

Gaussian Naive Bayes

Use Case: Continuous features that follow a normal distribution

Assumptions:

- Features are continuous
- Each feature follows a Gaussian (normal) distribution for each class
- Features are conditionally independent

Likelihood Calculation:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_{y,i}^2}} \exp\left(-\frac{(x_i - \mu_{y,i})^2}{2\sigma_{y,i}^2}\right) \tag{3}$$

Where:

- $\mu_{y,i}$ is the mean of feature i for class y
- $\sigma_{y,i}^2$ is the variance of feature i for class y

Gaussian Naive Bayes: Training

Training Process:

1. **Calculate Prior Probabilities:** $P(y) = \frac{\text{Number of samples in class } y}{\text{Total number of samples}}$
2. **Calculate Mean and Variance for each feature-class pair:**

$$\mu_{y,i} = \frac{1}{N_y} \sum_{j:y_j=y} x_{j,i} \tag{4}$$

$$\sigma^2_{y,i} = \frac{1}{N_y} \sum_{j:y_j=y} (x_{j,i} - \mu_{y,i})^2 \tag{5}$$

Where N_y is the number of samples in class y.

Smoothing

To avoid zero variance, add a small smoothing parameter ϵ:

$$\sigma^2_{y,i} = \sigma^2_{y,i} + \epsilon$$

Gaussian Naive Bayes: Example

Dataset: Height and Weight to classify Gender

Height (cm)	Weight (kg)	Gender
180	70	Male
170	60	Female
175	65	Male
165	55	Female
185	75	Male

Training Results:

- $P(\text{Male}) = 3/5 = 0.6$, $P(\text{Female}) = 2/5 = 0.4$
- Male: $\mu_{\text{height}} = 180$, $\sigma^2_{\text{height}} = 25$
- Female: $\mu_{\text{height}} = 167.5$, $\sigma^2_{\text{height}} = 12.5$
- Similar calculations for weight...

Multinomial Naive Bayes

Use Case: Discrete features representing counts or frequencies

Common Applications:

- Text classification (word counts)
- Document categorization
- Spam filtering

Feature Representation: Features represent counts: $x_i \in \{0, 1, 2, 3, \ldots\}$

Likelihood Calculation:

$$P(x_i|y) = \frac{N_{y,i} + \alpha}{N_y + \alpha \cdot n} \tag{6}$$

Where:

- $N_{y,i}$ is the count of feature i in class y
- N_y is the total count of all features in class y
- α is the smoothing parameter (usually 1)
- n is the number of features

Multinomial Naive Bayes: Text Classification

Example: Email Spam Classification

Email	"free"	"money"	"meeting"	Class
1	2	1	0	Spam
2	0	0	3	Ham
3	1	2	0	Spam
4	0	0	1	Ham

Training Calculations:

- $P(\text{Spam}) = 2/4 = 0.5$, $P(\text{Ham}) = 2/4 = 0.5$
- Spam: $N_{\text{spam,"free"}} = 3$, $N_{\text{spam}} = 6$
- $P(\text{"free"}|\text{Spam}) = \frac{3+1}{6+3\cdot 1} = \frac{4}{9}$

Laplace Smoothing

Adding $\alpha = 1$ prevents zero probabilities for unseen features.

Multinomial vs. Gaussian

Multinomial Naive Bayes

- Discrete count data
- Text classification
- Bag-of-words models
- Uses Laplace smoothing
- Features: word frequencies

Example Features:

- Word count: 5
- Term frequency: 0.02
- Document length: 250

Gaussian Naive Bayes

- Continuous numerical data
- Medical diagnosis
- Sensor measurements
- Uses variance smoothing
- Features: real-valued

Example Features:

- Temperature: 36.5°C
- Blood pressure: 120/80
- Height: 175.3 cm

Bernoulli Naive Bayes

Use Case: Binary features (presence/absence)

Feature Representation: Features are binary: $x_i \in \{0, 1\}$

- $1 =$ feature present
- $0 =$ feature absent

Likelihood Calculation:

$$P(x_i|y) = \begin{cases} p_{y,i} & \text{if } x_i = 1 \\ 1 - p_{y,i} & \text{if } x_i = 0 \end{cases} \tag{7}$$

Where $p_{y,i}$ is the probability that feature i is present in class y:

$$p_{y,i} = \frac{N_{y,i} + \alpha}{N_y + 2\alpha} \tag{8}$$

$N_{y,i} =$ number of samples in class y where feature i is present

Bernoulli Naive Bayes: Example

Example: Document Classification (Binary Features)

Document	"urgent"	"meeting"	"sale"	Class
1	1	0	1	Spam
2	0	1	0	Ham
3	1	0	1	Spam
4	0	1	0	Ham
5	1	1	0	Ham

Training:

- $P(\text{Spam}) = 2/5$, $P(\text{Ham}) = 3/5$
- For "urgent" in Spam: $p_{\text{spam,"urgent"}} = \frac{2+1}{2+2} = 0.75$
- For "urgent" in Ham: $p_{\text{ham,"urgent"}} = \frac{1+1}{3+2} = 0.4$

Key Differences: Multinomial vs. Bernoulli

Multinomial NB

- Considers frequency of features
- Good for: "How many times does 'free' appear?"
- Features can be > 1
- Better for longer documents
- Captures feature importance through counts

Example: Email with "free" appearing 5 times is more likely spam than one with "free" appearing once.

Bernoulli NB

- Considers presence/absence only
- Good for: "Does 'free' appear?"
- Features are binary (0 or 1)
- Better for shorter documents
- Explicitly models feature absence

Example: Only cares whether "free" appears or not, regardless of frequency.

Important

Bernoulli NB explicitly models the probability of features being **absent**, while Multinomial NB ignores absent features.

Choosing the Right Naive Bayes Variant

Aspect	Gaussian	Multinomial	Bernoulli
Data Type	Continuous, real-valued	Discrete counts	Binary (0/1)
Features	Height, temperature, sensor readings	Word counts, frequencies	Feature presence/absence
Applications	Medical diagnosis, sensor data	Text classification, NLP	Document classification, feature selection
Smoothing	Variance smoothing	Laplace smoothing	Laplace smoothing
Zero Handling	Add small ϵ to variance	Add α to counts	Add α to counts

Decision Guide:

- Continuous features? $\rightarrow$ Gaussian NB
- Count/frequency data? $\rightarrow$ Multinomial NB
- Binary features? $\rightarrow$ Bernoulli NB

Real-World Applications

Gaussian Naive Bayes:

- Medical diagnosis
- Iris flower classification
- Stock market prediction
- Weather forecasting
- Quality control in manufacturing

Multinomial Naive Bayes:

- Email spam filtering
- News article categorization
- Product review sentiment
- Language detection
- Topic modeling

Bernoulli Naive Bayes:

- Document classification
- Gene expression analysis
- Market basket analysis
- Feature selection problems
- Boolean query matching

Pro Tip

For text data, try both Multinomial and Bernoulli variants - the best choice depends on your specific dataset and problem characteristics.

Advantages of Naive Bayes

Computational Advantages:

- Fast training - $O(n \times d)$
- Fast prediction - $O(d)$
- Low memory requirements
- Scales well with data size

Statistical Advantages:

- Works well with small datasets
- Handles multiple classes naturally
- Not sensitive to irrelevant features
- Provides probability estimates

Practical Advantages:

- Simple to implement and understand
- No hyperparameter tuning needed
- Robust to outliers (Multinomial/Bernoulli)
- Good baseline classifier
- Handles missing values well

Performance

Often surprisingly competitive with more complex algorithms, especially for text classification tasks.

Limitations of Naive Bayes

The Independence Assumption:

- Features are rarely truly independent in real data
- Can lead to overconfident predictions
- May miss important feature interactions

Other Limitations:

- Categorical inputs: Need to be converted for Gaussian NB
- Zero frequency problem: Requires smoothing
- Probability calibration: May need recalibration for reliable probabilities
- Continuous features: Gaussian assumption may not hold

When NOT to use Naive Bayes

- Strong feature correlations exist
- Need exact probability estimates
- Complex feature interactions are important

Implementation Tips

Data Preprocessing:

- Gaussian NB: Consider feature scaling/normalization
- Multinomial NB: Ensure non-negative features
- Bernoulli NB: Convert to binary (0/1) features

Smoothing Parameters:

- Start with $\alpha = 1$ (Laplace smoothing)
- Use cross-validation to tune if needed
- Smaller α for larger datasets

Common Pitfalls:

- Forgetting to handle zero probabilities
- Using wrong variant for data type
- Not considering feature correlations
- Ignoring class imbalance

Performance Evaluation

Metrics to Consider:

- Accuracy: Overall correctness
- Precision/Recall: Especially for imbalanced datasets
- F1-Score: Harmonic mean of precision and recall
- Log-loss: For probability quality assessment

Cross-Validation:

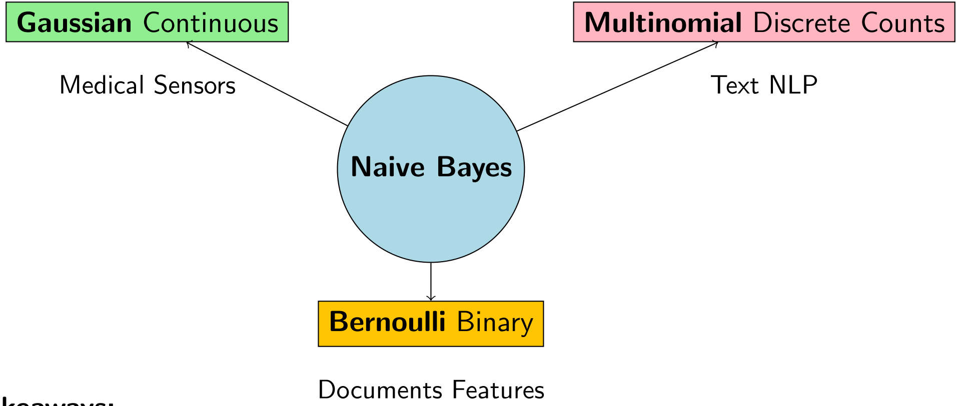
- Use k-fold CV for reliable estimates
- Stratified CV for imbalanced classes
- Compare against baseline models

Benchmark Comparison

Always compare Naive Bayes performance against:

- Logistic Regression
- Decision Trees
- Random Forest

Summary: Naive Bayes Variants



Key Takeaways:

- Choose variant based on feature type and data characteristics
- Simple yet effective for many classification tasks
- Fast training and prediction makes it ideal for large datasets
- Independence assumption is "naive" but often works well in practice

Next Steps

What's Next:

- Practice implementing all three variants
- Experiment with different smoothing parameters
- Compare performance on your datasets
- Explore probability calibration techniques

? Questions? ?

Thank you for your attention!