JOHNS HOPKINS
WHITING SCHOOL of ENGINEERING

Algorithms for Data Science

Statistical Algorithms: Bayes Classifiers

Classifying Under Uncertainty

Challenge

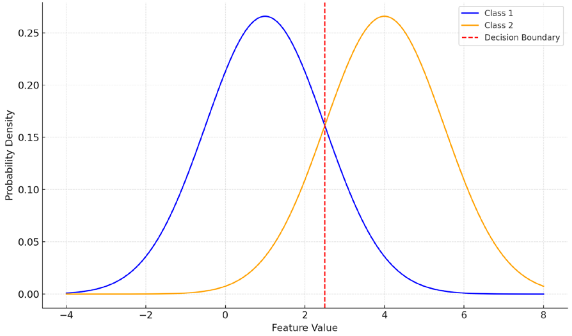
How do we classify data in uncertain environments?

Solution

Leverage probabilistic reason and Bayesian statistics.

Key Elements:

- Provides prior probabilities rather than binary outcomes.
- Incorporates prior knowledge (prior).
- Adjusts dynamically based on observed evidence.



Bayes Classifier: Mathematical Foundations

- **Bayes' Theorem:**

$$P(C_k \mid x) = \frac{P(x \mid C_k)P(C_k)}{P(x)}$$

Where:

$P(C_k \mid x)$: Posterior probability

$P(C_k)$: Prior probability

$P(x \mid C_k)$: Likelihood

$P(x)$: Marginal probability

Classification Rule:

$$C(x) = \arg\max_k P(C_k \mid x)$$

Bayes Classifier: Algorithm Analysis

1. **Compute Priors ($P(C_k)$):** $\leftarrow$ $O(N)$
   - Calculate the prior probability for each class based on class freq.
2. **Joint Likelihood Calculation:** $\leftarrow$ $O\left(N \cdot K \cdot 2^D\right)$
   - For each class C_k compute:
     $P\left(x_1, x_2, \ldots, x_D \mid C_k\right)$
   - Requires estimating probabilities for all feature combos.
3. **Apply Bayes' Theorem** $\leftarrow$ $O(N \cdot K)$
4. **Apply Classification Rule**

Total Runtime Complexity:
$O\left(N \cdot K \cdot 2^D\right)$

Bayes Classifier: Correctness Proof

Theorem: **The Bayes Classifier minimizes the probability of misclassification under the assumption that the true distributions are known.**

1. **Posterior Probability:**
   - Bayes' theorem computes posterior probabilities optimally, incorporating priors and likelihoods.
2. **Classification Rule:**
   - Assign data point x to the class C_k with the highest posterior.
3. **Minimizing Expected Loss:**
   - By choosing the class with the highest posterior, the classifier minimizes the expected probability of misclassification.

Bayes Classifier: Application

Medical Diagnostics

- **Task:** Predict the probability of a disease based on symptoms and test results.
- **How it Works:** Combine disease prevalence (prior) with test accuracy (likelihood).
- **Example:** Determining the likelihood of diabetes given glucose levels and patient history.

Spam Filtering

- **Task:** Classify emails as spam or not spam based on word usage patterns.
- **How it Works:** Use word frequencies as features and applies Bayes' theorem.
- **Example:** "Win now" email classified as spam with high confidence.

Risk Assessment

- **Task:** Evaluate the probability of default or fraud in financial transactions.
- **How it Works:** Incorporates default rates (prior) with transaction details.
- **Example:** Flagging high-risk loan applicants based on credit scores.

From Bayes to Naïve Bayes

Bayes Classifier

- Requires estimating **joint probabilities** for high-dimensional data.
- **Computationally expensive** with large datasets and many features.
- **Prone to overfitting** when data is limited.

Naïve Bayes

- **Assumes conditional independence** between features.
- **Reduces the complexity** of probability estimation.
- Sacrifices some modeling accuracy for significant **computational efficiency**.

Naïve Bayes: Mathematical Foundations

- **Naïve Bayes Assumption**: Features are conditionally independent given the class:

$$P\left(x_1, x_2, \ldots, x_D \mid C_k\right)=\prod_{d=1}^D P\left(x_d \mid C_k\right)$$

- **Posterior Probability:**

$$P\left(C_k \mid x\right) \propto P\left(C_k\right) \prod_{d=1}^D P\left(x_d \mid C_k\right)$$

Where:

$P\left(C_k\right)$: Prior probability of class C_k

$P\left(x_d \mid C_k\right)$: Likelihood of feature x_d given class C_k

- **Classification Rule:**

$$C(x)=\arg \max _k P\left(C_k\right) \prod_{d=1}^D P\left(x_d \mid C_k\right)$$

Naïve Bayes: Algorithm Analysis

1. **Compute Priors ($P(C_k)$):** ⟵ $O(N)$
   - Calculate the prior probability for each class based on class freq.
2. **Compute Independent Likelihoods ($P(x \mid C_k)$):** ⟵ $O(N \cdot K \cdot D)$
   - Estimate probabilities for each feature and class.
     - For continuous features: Use PDF (e.g. Gaussian)
     - For categorical features: Use freq. counts.
3. **Apply Bayes' Theorem** ⟵ $O(D \cdot K)$
4. **Apply Classification Rule**

Total Runtime Complexity:
$O(N \cdot K \cdot D)$

Naïve Bayes: Strengths and Limitations

Strengths

- Computationally **efficient** and **scalable**.
- **Effectively separates classes** when features provide complementary evidence**.**
- **Robust** with small datasets.

Limitations

- **Independence Assumption**: fails to model features correlations.
- **Sensitive to class imbalance** due to heavy reliance on prior probabilities.

JOHNS HOPKINS

WHITING SCHOOL of ENGINEERING