Naive Bayes Classification Algorithm for Weather Dataset

Bindeshwar Singh Kushwaha
PostNetwork Academy

Naive Bayes Classifier - Introduction

- **Naive Bayes is a probabilistic classification algorithm.**

Naive Bayes Classifier - Introduction

- **Naive Bayes is a probabilistic classification algorithm.**
- **It is based on Bayes' Theorem and the naive independence assumption.**

Naive Bayes Classifier - Introduction

- **Naive Bayes is a probabilistic classification algorithm.**
- **It is based on Bayes' Theorem and the naive independence assumption.**
- **Suppose we have a feature vector** $\mathbf{X} = (x_1, x_2, ..., x_n)$ **and a class** y**.**

Naive Bayes Classifier - Introduction

- **Naive Bayes is a probabilistic classification algorithm.**
- **It is based on Bayes' Theorem and the naive independence assumption.**
- **Suppose we have a feature vector $\mathbf{X} = (x_1, x_2, ..., x_n)$ and a class y.**
- **Bayes Theorem:**

$$P(y|\mathbf{X}) = \frac{P(\mathbf{X}|y) \cdot P(y)}{P(\mathbf{X})}$$

Naive Bayes Classifier - Introduction

- **Naive Bayes is a probabilistic classification algorithm.**
- **It is based on Bayes' Theorem and the naive independence assumption.**
- **Suppose we have a feature vector $\mathbf{X} = (x_1, x_2, ..., x_n)$ and a class y.**
- **Bayes Theorem:**

$$P(y|\mathbf{X}) = \frac{P(\mathbf{X}|y) \cdot P(y)}{P(\mathbf{X})}$$

- **Here:**
  - $P(y|\mathbf{X})$ = **Posterior probability of class y given features $\mathbf{X}$**
  - $P(\mathbf{X}|y)$ = **Likelihood of features given class y**
  - $P(y)$ = **Prior probability of class y**
  - $P(\mathbf{X})$ = **Evidence (same for all classes, can be ignored)**

Naive Bayes - Independence Assumption

- **Naive Bayes assumes that features are conditionally independent given the class:**

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

Naive Bayes - Independence Assumption

- **Naive Bayes assumes that features are conditionally independent given the class:**

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

- **This makes computation efficient, especially for high-dimensional feature vectors.**

Naive Bayes - Independence Assumption

- **Naive Bayes assumes that features are conditionally independent given the class:**

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

- **This makes computation efficient, especially for high-dimensional feature vectors.**
- **Each feature contributes independently to the probability of the class.**

Naive Bayes - Independence Assumption

- Naive Bayes assumes that features are conditionally independent given the class:

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

- This makes computation efficient, especially for high-dimensional feature vectors.
- Each feature contributes independently to the probability of the class.
- Posterior probability then becomes:

$$P(y|\mathbf{X}) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

Naive Bayes - Classification Rule

- **To classify a new instance X:**

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

Naive Bayes - Classification Rule

- **To classify a new instance X:**

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

- **Compute posterior probabilities for each possible class.**

Naive Bayes - Classification Rule

- **To classify a new instance X:**

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

- **Compute posterior probabilities for each possible class.**
- **Predict the class y with the highest posterior probability.**

Naive Bayes - Classification Rule

- To classify a new instance **X**:

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

- Compute posterior probabilities for each possible class.
- Predict the class y with the highest posterior probability.
- Summary: Naive Bayes converts probability estimates into a simple rule for classification.

Naive Bayes - Pros and Cons

- **Advantages:**
  - **Simple and fast to train**
  - **Works well with small datasets**
  - **Effective for text classification (spam detection, sentiment analysis)**

Naive Bayes - Pros and Cons

- **Advantages:**
  - **Simple and fast to train**
  - **Works well with small datasets**
  - **Effective for text classification (spam detection, sentiment analysis)**
- **Disadvantages:**
  - **Assumes feature independence (rarely perfectly true)**
  - **Can perform poorly with correlated features**

Naive Bayes - Pros and Cons

- **Advantages:**
  - **Simple and fast to train**
  - **Works well with small datasets**
  - **Effective for text classification (spam detection, sentiment analysis)**
- **Disadvantages:**
  - **Assumes feature independence (rarely perfectly true)**
  - **Can perform poorly with correlated features**
- **Despite simplicity, very effective for many real-world applications.**

Naive Bayes - Worked Example (Setup)

- **Suppose we have n features: $x_1, x_2, ..., x_n$.**

Naive Bayes - Worked Example (Setup)

- **Suppose we have n features: $x_1, x_2, ..., x_n$.**
- **The target class is y.**

Naive Bayes - Worked Example (Setup)

- **Suppose we have n features: $x_1, x_2, ..., x_n$.**
- **The target class is y.**
- **Step 1: Compute prior probabilities for each class y:**

$$P(y) = \frac{\textbf{count of class } y}{\textbf{total instances}}$$

Naive Bayes - Worked Example (Setup)

- **Suppose we have n features: $x_1, x_2, ..., x_n$.**
- **The target class is y.**
- **Step 1: Compute prior probabilities for each class y:**

$$P(y) = \frac{\text{count of class } y}{\text{total instances}}$$

- **Step 2: Compute likelihoods for each feature:**

$$P(x_i|y) = \frac{\text{count of } x_i \text{ with class } y}{\text{total instances of class } y}$$

Naive Bayes - Worked Example (Setup)

- **Suppose we have n features: $x_1, x_2, ..., x_n$.**
- **The target class is y.**
- **Step 1: Compute prior probabilities for each class y:**

$$P(y) = \frac{\text{count of class } y}{\text{total instances}}$$

- **Step 2: Compute likelihoods for each feature:**

$$P(x_i|y) = \frac{\text{count of } x_i \text{ with class } y}{\text{total instances of class } y}$$

- **Step 3: For a new instance $\mathbf{X}_{\text{new}} = (x_1, x_2, ..., x_n)$, compute posterior probability:**

$$P(y|\mathbf{X}_{\text{new}}) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

Naive Bayes - Prediction Step

- **Compute posterior probabilities for all classes:**

$$P(y_1|\mathbf{X}_{\mathbf{new}}), P(y_2|\mathbf{X}_{\mathbf{new}}), \ldots$$

Naive Bayes - Prediction Step

- **Compute posterior probabilities for all classes:**

$$P(y_1|\mathbf{X}_{\mathbf{new}}), P(y_2|\mathbf{X}_{\mathbf{new}}), \ldots$$

- **Compare posterior probabilities:**

$$\hat{y} = \arg\max_{y} P(y|\mathbf{X}_{\mathbf{new}})$$

Naive Bayes - Prediction Step

- **Compute posterior probabilities for all classes:**

$$P(y_1|\mathbf{X}_{\mathbf{new}}), P(y_2|\mathbf{X}_{\mathbf{new}}), \ldots$$

- **Compare posterior probabilities:**

$$\hat{y} = \arg\max_{y} P(y|\mathbf{X}_{\mathbf{new}})$$

- **The class with the highest posterior probability is the predicted class.**

Naive Bayes - Prediction Step

- **Compute posterior probabilities for all classes:**

$$P(y_1|\mathbf{X}_{\text{new}}), P(y_2|\mathbf{X}_{\text{new}}), \ldots$$

- **Compare posterior probabilities:**

$$\hat{y} = \arg\max_{y} P(y|\mathbf{X}_{\text{new}})$$

- **The class with the highest posterior probability is the predicted class.**
- **Summary of steps:**
  1. **Compute priors** $P(y)$
  2. **Compute likelihoods** $P(x_i|y)$ **for each feature**
  3. **Multiply likelihoods and priors for posterior**
  4. **Select class with maximum posterior**

Weather Dataset - PlayTennis

Features: Outlook (x_1), Temperature (x_2), Humidity (x_3), Wind (x_4)
Target: PlayTennis (y)

Day	Outlook	Temp	Humidity	Wind	PlayTennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Step 1: Compute Prior Probabilities $P(y)$

- **Count of Yes = 9, Count of No = 5, Total instances = 14**

Step 1: Compute Prior Probabilities $P(y)$

- **Count of Yes = 9, Count of No = 5, Total instances = 14**
- **Priors:**

$$P(y = \textbf{Yes}) = \frac{9}{14} \approx 0.643$$

$$P(y = \textbf{No}) = \frac{5}{14} \approx 0.357$$

Step 2: Compute Likelihoods $P(x_i|y)$

- **Outlook** (x_1)**:**

$$P(\textbf{Sunny}|Yes) = 0.222, P(\textbf{Sunny}|No) = 0.6$$
$$P(\textbf{Overcast}|Yes) = 0.444, P(\textbf{Overcast}|No) = 0$$
$$P(\textbf{Rain}|Yes) = 0.333, P(\textbf{Rain}|No) = 0.4$$

Step 2: Compute Likelihoods $P(x_i|y)$

- **Outlook** (x_1):

$$P(\textbf{Sunny}|Yes) = 0.222, P(\textbf{Sunny}|No) = 0.6$$
$$P(\textbf{Overcast}|Yes) = 0.444, P(\textbf{Overcast}|No) = 0$$
$$P(\textbf{Rain}|Yes) = 0.333, P(\textbf{Rain}|No) = 0.4$$

- **Temperature** (x_2):

$$P(\textbf{Hot}|Yes) = 0.222, P(\textbf{Cool}|Yes) = 0.333, P(\textbf{Mild}|Yes) = 0.444$$
$$P(\textbf{Hot}|No) = 0.4, P(\textbf{Cool}|No) = 0.2, P(\textbf{Mild}|No) = 0.4$$

Step 2: Compute Likelihoods $P(x_i|y)$

- **Outlook** (x_1):

$$P(\textbf{Sunny}|Yes) = 0.222, P(\textbf{Sunny}|No) = 0.6$$
$$P(\textbf{Overcast}|Yes) = 0.444, P(\textbf{Overcast}|No) = 0$$
$$P(\textbf{Rain}|Yes) = 0.333, P(\textbf{Rain}|No) = 0.4$$

- **Temperature** (x_2):

$$P(\textbf{Hot}|Yes) = 0.222, P(\textbf{Cool}|Yes) = 0.333, P(\textbf{Mild}|Yes) = 0.444$$
$$P(\textbf{Hot}|No) = 0.4, P(\textbf{Cool}|No) = 0.2, P(\textbf{Mild}|No) = 0.4$$

- **Humidity** (x_3):

$$P(\textbf{High}|Yes) = 0.333, P(\textbf{Normal}|Yes) = 0.667$$
$$P(\textbf{High}|No) = 0.8, P(\textbf{Normal}|No) = 0.2$$

Step 2: Compute Likelihoods $P(x_i|y)$

- **Outlook** (x_1):

$$P(\textbf{Sunny}|Yes) = 0.222, P(\textbf{Sunny}|No) = 0.6$$

$$P(\textbf{Overcast}|Yes) = 0.444, P(\textbf{Overcast}|No) = 0$$

$$P(\textbf{Rain}|Yes) = 0.333, P(\textbf{Rain}|No) = 0.4$$

- **Temperature** (x_2):

$$P(\textbf{Hot}|Yes) = 0.222, P(\textbf{Cool}|Yes) = 0.333, P(\textbf{Mild}|Yes) = 0.444$$

$$P(\textbf{Hot}|No) = 0.4, P(\textbf{Cool}|No) = 0.2, P(\textbf{Mild}|No) = 0.4$$

- **Humidity** (x_3):

$$P(\textbf{High}|Yes) = 0.333, P(\textbf{Normal}|Yes) = 0.667$$

$$P(\textbf{High}|No) = 0.8, P(\textbf{Normal}|No) = 0.2$$

- **Wind** (x_4):

$$P(\textbf{Weak}|Yes) = 0.667, P(\textbf{Strong}|Yes) = 0.333$$

$$P(\textbf{Weak}|No) = 0.4, P(\textbf{Strong}|No) = 0.6$$

Step 3: Posterior Probability for Yes

$$\mathbf{X}_{15} = (\textbf{Sunny, Cool, High, Strong})$$

$$P(y = \textbf{Yes}|\mathbf{X}_{15}) \propto P(y = \textbf{Yes}) \cdot P(x_1|Yes) \cdot P(x_2|Yes) \cdot P(x_3|Yes) \cdot P(x_4|Yes)$$
$$= 0.643 \cdot 0.222 \cdot 0.333 \cdot 0.333 \cdot 0.333$$

Step 3: Posterior Probability for Yes

$$\mathbf{X}_{15} = (\textbf{Sunny, Cool, High, Strong})$$

$$P(y = \textbf{Yes}|\mathbf{X}_{15}) \propto P(y = \textbf{Yes}) \cdot P(x_1|Yes) \cdot P(x_2|Yes) \cdot P(x_3|Yes) \cdot P(x_4|Yes)$$
$$= 0.643 \cdot 0.222 \cdot 0.333 \cdot 0.333 \cdot 0.333$$

- **Step-by-step:**
    - $0.643 \cdot 0.222 \approx 0.1425$
    - $0.1425 \cdot 0.333 \approx 0.0475$
    - $0.0475 \cdot 0.333 \approx 0.0158$
    - $0.0158 \cdot 0.333 \approx 0.00525$

Step 3: Posterior Probability for Yes

$$\mathbf{X}_{15} = (\textbf{Sunny, Cool, High, Strong})$$

$$P(y = \textbf{Yes}|\mathbf{X}_{15}) \propto P(y = \textbf{Yes}) \cdot P(x_1|Yes) \cdot P(x_2|Yes) \cdot P(x_3|Yes) \cdot P(x_4|Yes)$$
$$= 0.643 \cdot 0.222 \cdot 0.333 \cdot 0.333 \cdot 0.333$$

- **Step-by-step:**
  - $0.643 \cdot 0.222 \approx 0.1425$
  - $0.1425 \cdot 0.333 \approx 0.0475$
  - $0.0475 \cdot 0.333 \approx 0.0158$
  - $0.0158 \cdot 0.333 \approx 0.00525$
- **So,** $P(y = \textbf{Yes}|\mathbf{X}_{15}) \approx 0.00525$

Step 4: Posterior Probability for No

$$P(y = \mathbf{No}|\mathbf{X}_{15}) \propto 0.357 \cdot 0.6 \cdot 0.2 \cdot 0.8 \cdot 0.6$$

Step 4: Posterior Probability for No

$$P(y = \mathbf{No} | \mathbf{X}_{15}) \propto 0.357 \cdot 0.6 \cdot 0.2 \cdot 0.8 \cdot 0.6$$

- **Step-by-step:**
  - $0.357 \cdot 0.6 \approx 0.2142$
  - $0.2142 \cdot 0.2 \approx 0.04284$
  - $0.04284 \cdot 0.8 \approx 0.03427$
  - $0.03427 \cdot 0.6 \approx 0.02056$

Step 4: Posterior Probability for No

$$P(y = \mathbf{No} | \mathbf{X}_{15}) \propto 0.357 \cdot 0.6 \cdot 0.2 \cdot 0.8 \cdot 0.6$$

- **Step-by-step:**
  - $0.357 \cdot 0.6 \approx 0.2142$
  - $0.2142 \cdot 0.2 \approx 0.04284$
  - $0.04284 \cdot 0.8 \approx 0.03427$
  - $0.03427 \cdot 0.6 \approx 0.02056$
- **So,** $P(y = \mathbf{No} | \mathbf{X}_{15}) \approx 0.0206$

Step 5: Classification Decision

- **Posterior probabilities:**

$$P(y = \textbf{Yes}|\mathbf{X}_{15}) \approx 0.00525$$
$$P(y = \textbf{No}|\mathbf{X}_{15}) \approx 0.0206$$

Step 5: Classification Decision

- **Posterior probabilities:**

$$P(y = \textbf{Yes}|\mathbf{X}_{15}) \approx 0.00525$$

$$P(y = \textbf{No}|\mathbf{X}_{15}) \approx 0.0206$$

- **Since** $0.0206 > 0.00525$**, the 15th instance is classified as:**

$$\hat{y} = \textbf{No}$$

Step 5: Classification Decision

- **Posterior probabilities:**

$$P(y = \textbf{Yes}|\mathbf{X}_{15}) \approx 0.00525$$

$$P(y = \textbf{No}|\mathbf{X}_{15}) \approx 0.0206$$

- **Since $0.0206 > 0.00525$, the 15th instance is classified as:**

$$\hat{y} = \textbf{No}$$

- **Summary of Steps:**
  1. **Compute priors $P(y)$**
  2. **Compute likelihoods $P(x_i|y)$ for each feature**
  3. **Multiply likelihoods and priors for posterior**
  4. **Select class with maximum posterior**

Reach PostNetwork Academy

Website

www.postnetwork.co

Reach PostNetwork Academy

Website

www.postnetwork.co

YouTube Channel

www.youtube.com/@postnetworkacademy

Reach PostNetwork Academy

Website

www.postnetwork.co

YouTube Channel

www.youtube.com/@postnetworkacademy

Facebook Page

www.facebook.com/postnetworkacademy

Reach PostNetwork Academy

Website

www.postnetwork.co

YouTube Channel

www.youtube.com/@postnetworkacademy

Facebook Page

www.facebook.com/postnetworkacademy

LinkedIn Page

www.linkedin.com/company/postnetworkacademy

Reach PostNetwork Academy

Website

www.postnetwork.co

YouTube Channel

www.youtube.com/@postnetworkacademy

Facebook Page

www.facebook.com/postnetworkacademy

LinkedIn Page

www.linkedin.com/company/postnetworkacademy

GitHub Repositories

www.github.com/postnetworkacademy

Thank You!