Naive Bayes Classifier

Mitchell [6.7-6.9]

Bayes Classifier

- Assume we want to classify **x** described by attributes $[a_1,...a_n]$.
- Bayes theorem tells us to find Cj for which this is maximum:

$$P(C_j \mid \mathbf{x}) \propto P(\mathbf{x} \mid C_j)\, P(C_j)$$

- This is expressed as

$$C = \underset{j}{\operatorname{argmax}}\; P(\mathbf{x} \mid C_j)\, P(C_j)$$

Bayes Classifier

- But it requires a lot of data to estimate (roughly $O(|A|^n)$ parameters for each class):

$$P(a_1,a_2,\ldots a_n| C_j)$$

- **Naïve Bayesian Approach:** We assume that the attribute values are conditionally independent given the class v_j so that

$$P(a_1,a_2,..,a_n|C_j) = \prod_i P(a_i|C_j)$$

- ***Naïve Bayes Classifier:***

$$C_{NB} = \text{argmax}_{Cj \in C} P(C_j) \prod_i P(a_i|C_j)$$

Independence & Conditional Independence

X and Y are independent if f $P(X,Y)=P(X)P(Y)$

Since $P(X,Y)= P(X \mid Y)\, P(Y)$ by definition, we have the equivalent definition of $P(X|Y) = P(X)$.

- This says that Y does not give me any extra information to change my belief about the probability of X.

X is conditionally independent of Y given Z if $P(X|Y,Z)=P(X|Z)$

Equivalently, if $P(X,Y|Z) = P(X|Z) \times P(Y|Z)$

Compare to $P(X,Y) = P(X) \times P(Y)$ in case of independence of X and Y

Independence & Conditional Independence

- Independence and conditional independence are important because they significantly reduce the number of parameters needed and reduce computation time.
  - Consider estimating the joint probability distribution of two random variables A and B:
    - $10^2=100$ vs $10+10=20$ if each have 10 possible outcomes
    - $100^2=10{,}000$ vs $100+100=200$ if each have 100 possible outcomes

Conditional Independence

- **Good example for intuition**:

- What is the conditional independence between the following three random variables Height, Age and Vocabulary Size? (example credit: Wikipedia)
  - Very small (short) people tend to be children with basic vocabularies.
  - So $P(V \mid H) \neq P(V)$ (I.e. H gives me info about V or H and V are Not independent)

  - But knowing the Age (A), there is no reason to think that one person's vocabulary is larger if we are told that they are taller.
  - I.e., $P(V \mid A, H) = P(V \mid A)$
  - Thus, V is dependent on H, but independent from H given A

Naive Bayes Classifier - Derivation

- **<u>Naïve Bayesian Approach:</u>** We assume that the attribute values are conditionally independent given the class v_j so that:

$$P(a_1,a_2,..,a_n|C_j) = \prod_i P(a_i|C_j)$$

- Use repeated applications of the definition of conditional probability (chain rule).

$$P(a_1,a_2,a_3|\ C) = \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

- If we assume that a_i are conditionally independent given C.

$$P(a_i\ |\ a_j,\ C) = \ldots\ldots\ldots\ldots\ldots\ldots$$

- Then we have (dropping irrelevant terms in the exact formula):

$$P(a_1,a_2,a_3|\ C) = \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

Naive Bayes Classifier - Derivation

- **<u>Naïve Bayesian Approach:</u>** We assume that the attribute values are conditionally independent given the class v_j so that:

$$P(a_1,a_2,..,a_n|C_j) = \prod_i P(a_i|C_j)$$

- Use repeated applications of the definition of conditional probability (chain rule).

$$P(a_1,a_2,a_3|\, C) = P(a_3|a_1,a_2,C)\; P(a_2|a_1,C)\; P(a_1|C)$$

- If we assume that a_i are conditionally independent given C.

$$P(a_i \mid a_j, C) = P(a_i \mid C)$$

- Then we have (dropping irrelevant terms in the exact formula):

$$P(a_1,a_2,a_3|\, C) = P(a_3|C)\; P(a_2|C)\; P(a_1|C)$$

Naïve Bayes Classification

Bayes rule:

$$P(Y = y_k | X_1 \ldots X_n) = \frac{P(Y = y_k) P(X_1 \ldots X_n | Y = y_k)}{\sum_j P(Y = y_j) P(X_1 \ldots X_n | Y = y_j)}$$

Assuming conditional independence among X_i's:

$$P(Y = y_k | X_1 \ldots X_n) = \frac{P(Y = y_k) \prod_i P(X_i | Y = y_k)}{\sum_j P(Y = y_j) \prod_i P(X_i | Y = y_j)}$$

(estimate in training)

So, to pick most probable Y for $X^{new} = (X_1, \ldots, X_n)$

$$Y^{new} \leftarrow \arg\max_{y_k} \; P(Y = y_k) \prod_i P(X_i^{new} | Y = y_k)$$

(testing)

Naïve Bayes: Training and testing

- Train Naïve Bayes (examples)

  for each* value y_k

  estimate $\pi_k \equiv P(Y = y_k)$

  for each* value x_{ij} of each attribute X_i

  estimate $\theta_{ijk} \equiv P(X_i = x_{ij} | Y = y_k)$

- Classify (X^{new})

$$Y^{new} \leftarrow \arg\max_{y_k} \; P(Y = y_k) \prod_i P(X_i^{new} | Y = y_k)$$

$$Y^{new} \leftarrow \arg\max_{y_k} \; \pi_k \prod_i \theta_{ijk}$$

Example with categorical variables

Patient comes with symptoms, the task is to assess whether it is simply Flu or something else.

We have some statistics (5 samples ☺) from years of medical data:

Fever	Cough	Body Ache	Flu
Yes	Yes	Yes	Yes
Yes	Yes	No	Yes
Yes	No	Yes	No
No	Yes	Yes	No
No	Yes	No	No

Training

Fever	Cough	Body Ache	Flu
Yes	Yes	Yes	Yes
Yes	Yes	No	Yes
Yes	No	Yes	No
No	Yes	Yes	No
No	Yes	No	No

Calculate Probabilities

Prior Probabilities:

$P(Flu = Yes) = 2/5 = 0.4$

$P(Flu = No) = 3/5 = 0.6$

Likelihoods:

$P(Fever = Yes \mid Flu = Yes) = 2/2 = 1$

$P(Cough = Yes \mid Flu = Yes) = 2/2 = 1$

$P(Body\ Ache = No \mid Flu = Yes) = 1/2 = 0.5$

$P(Fever = Yes \mid Flu = No) = 1/3 \approx 0.33$

$P(Cough = Yes \mid Flu = No) = 2/3 \approx 0.67$

$P(Body\ Ache = No \mid Flu = No) = 2/3 \approx 0.67$

Testing

Prior Probabilities:

$P(\text{Flu} = \text{Yes}) = 2/5 = 0.4$

$P(\text{Flu} = \text{No}) = 3/5 = 0.6$

Likelihoods:

$P(\text{Fever} = \text{Yes} \mid \text{Flu} = \text{Yes}) = 2/2 = 1$

$P(\text{Cough} = \text{Yes} \mid \text{Flu} = \text{Yes}) = 2/2 = 1$

$P(\text{Body Ache} = \text{No} \mid \text{Flu} = \text{Yes}) = 1/2 = 0.5$

$P(\text{Fever} = \text{Yes} \mid \text{Flu} = \text{No}) = 1/3 \approx 0.33$

$P(\text{Cough} = \text{Yes} \mid \text{Flu} = \text{No}) = 2/3 \approx 0.67$

$P(\text{Body Ache} = \text{No} \mid \text{Flu} = \text{No}) = 2/3 \approx 0.67$

Inference

Given a patient with **Fever = Yes, Cough = Yes, Body Ache = No**, predict whether they have Flu or Else

- **$P(\text{Flu} = \text{Yes} \mid \text{Data}) \propto$** ....

- **$P(\text{Flu} = \text{No} \mid \text{Data}) \propto$** ....

Testing

Prior Probabilities:

P(Flu = Yes) = 2/5 = 0.4

P(Flu = No) = 3/5 = 0.6

Likelihoods:

P(Fever = Yes | Flu = Yes) = 2/2 = 1

P(Cough = Yes | Flu = Yes) = 2/2 = 1

P(Body Ache = No | Flu = Yes) = 1/2 = 0.5

P(Fever = Yes | Flu = No) = 1/3 ≈ 0.33

P(Cough = Yes | Flu = No) = 2/3 ≈ 0.67

P(Body Ache = No | Flu = No) = 2/3 ≈ 0.67

Inference

Given a patient with **Fever = Yes, Cough = Yes, Body Ache = No**, predict whether they have Flu or Else

P(Flu = Yes | Data) ∝ P(Flu = Yes) × P(Fever = Yes | Flu = Yes) × P(Cough = Yes | Flu = Yes) × P(Body Ache = No | Flu = Yes)
= 0.4 × 1 × 1 × 0.5 = **0.2**

P(Flu = No | Data) ∝ P(Flu = No) × P(Fever = Yes | Flu = No) × P(Cough = Yes | Flu = No) × P(Body Ache = No | Flu = No)
= 0.6 × 0.33 × 0.67 × 0.67 ≈ **0.089**

Testing

$\mathbf{P(Flu = Yes \mid Data)} \propto P(Flu = Yes) \times P(Fever = Yes \mid Flu = Yes) \times P(Cough = Yes \mid Flu = Yes) \times P(Body\ Ache = No \mid Flu = Yes)$
$= 0.4 \times 1 \times 1 \times 0.5 = \mathbf{0.2}$

$\mathbf{P(Flu = No \mid Data)} \propto P(Flu = No) \times P(Fever = Yes \mid Flu = No) \times P(Cough = Yes \mid Flu = No) \times P(Body\ Ache = No \mid Flu = No)$
$= 0.6 \times 0.33 \times 0.67 \times 0.67 \approx \mathbf{0.089}$

Normalize Probabilities

$P(Flu = Yes \mid Data) = 0.2 / (0.2 + 0.089) \approx 0.69$

$P(Flu = No \mid Data) = 0.089 / (0.2 + 0.089) \approx 0.31$

From Mitchell Book for self-study

Example with categorical variables

Example from Mitchell Chp 3.

PlayTennis: training examples

Target

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Training

First the probabilities of the different target values can easilty be estimated based on their frequencies over the 14 training examples.

This is based on the intuitive solution provided by the Maximum Likelihood Estimation. That is the probability of a class is estimated over the ratio of that class observed in the training data:

$$P(PlayTennis = yes) = 9/14 = 0.64$$
$$P(PlayTennis = no) = 5/14 = 0.36$$

Similarly, we can estimate the conditional probabilities as the ratios observed in the given class, for the particular attribute value. For example, those for $Wind = strong$ are:

$$P(Wind = strong \mid PlayTennis = yes) = 3/9 = 0.33$$
$$P(Wind = strong \mid PlayTennis = no) = 3/5 = 0.60$$

Training

Consider each feature independenlty and for each class label y_k and $x_{i,j}$ value of feature i estimate $\mathbf{P}\left(X_i = x_{i,j} \,|\, Y = y_k\right)$:

$$
\begin{array}{ll}
\mathbf{P}\left(O = \texttt{sunny} \,|\, Play = \texttt{Yes}\right) & \mathbf{P}\left(O = \texttt{sunny} \,|\, Play = \texttt{No}\right) \\
\mathbf{P}\left(O = \texttt{overcast} \,|\, Play = \texttt{Yes}\right) & \mathbf{P}\left(O = \texttt{overcast} \,|\, Play = \texttt{No}\right) \\
\ldots & \\
\mathbf{P}\left(T = \texttt{hot} \,|\, Play = \texttt{Yes}\right) & \mathbf{P}\left(T = \texttt{hot} \,|\, Play = \texttt{No}\right) \\
\ldots & \\
\mathbf{P}\left(H = \texttt{high} \,|\, Play = \texttt{Yes}\right) & \mathbf{P}\left(H = \texttt{high} \,|\, Play = \texttt{No}\right) \\
\ldots & \\
\mathbf{P}\left(W = \texttt{true} \,|\, Play = \texttt{Yes}\right) & \mathbf{P}\left(W = \texttt{true} \,|\, Play = \texttt{No}\right)\ldots
\end{array}
$$

And estimate the class prior $\mathbf{P}\left(Y = y_k\right)$:

$\mathbf{P}\left(Play = \texttt{Yes}\right)$ (Note that $\mathbf{P}\left(Play = \texttt{No}\right) = 1 - \mathbf{P}\left(Play = \texttt{Yes}\right)$)

Testing

- Posterior probability for a new instance with the feature vector:
- $X_{\texttt{new}}$ `= (sunny, cool, high, strong)`

Posterior Likelihood Prior

$$\mathbf{P}\left(Play \,|\, X\right) \propto \mathbf{P}\left(X \,|\, Play\right) \mathbf{P}\left(Play\right)$$

$$\mathbf{P}\left(Play = Y \,|\, X\right) \propto \mathbf{P}\left(X \,|\, Play = Y\right) \mathbf{P}\left(Play = Y\right)$$
$$\mathbf{P}\left(Play = N \,|\, X\right) \propto \mathbf{P}\left(X \,|\, Play = N\right) \mathbf{P}\left(Play = N\right)$$

Example

- Estimating the likelihood:

$$\mathbf{P}(X \mid Play = Y) = \mathbf{P}(O = sunny \mid Play = Y)\,\mathbf{P}(T = cool \mid Play = Y)\,\mathbf{P}(H = high \mid Play = Y)\,\mathbf{P}(W = strong \mid Play = Y)$$
$$= \frac{2}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} \approx 0.0082$$
$$\mathbf{P}(X \mid Play = N) = \mathbf{P}(O = sunny \mid Play = N)\,\mathbf{P}(T = cool \mid Play = N)\,\mathbf{P}(H = high \mid Play = N)\,\mathbf{P}(W = strong \mid Play = N)$$
$$= \frac{3}{5} \cdot \frac{1}{5} \cdot \frac{4}{5} \cdot \frac{3}{5} = 0.0576$$

- Estimating the posterior:

$$\mathbf{P}(Play = Y \mid X) \propto \mathbf{P}(X \mid Play = Y)\,\mathbf{P}(Play = Y) = 0.0082 * \frac{9}{14} \approx 0.0052$$
$$\mathbf{P}(Play = N \mid X) \propto \mathbf{P}(X \mid Play = N)\,\mathbf{P}(Play = N) = 0.0576 * \frac{5}{14} \approx 0.0205$$

- Class label predicted for X is then Play = No

Obtaining Normalized Probabilities

By normalizing the above quantities to sum to one we can calculate the conditional probability that the target value is no, given the observed attribute values.

P(Play=No | X) = $\frac{0.0206}{0.0206+0.0053} = 0.795$

Notice that we use equality here, taking into consideration P(X).

- What is P(X) here? Try to write it in terms of known terms.

Naïve Bayes Subtleties

Naïve Bayes Subtleties

1. Usually features are not conditionally independent

$$\boxed{P(X_1 ... X_n|Y) \neq \prod_i P(X_i|Y)}$$

- It does not produce accurate probability estimates when its independence assumptions are violated, but it works well in many cases, as picks the correct maximum-probability class [Domingos&Pazzani, 1996].

- Typically handles noise well since it does not even focus on completely fitting the training data.

Naive Bayes Subtleties

2. What if none of the training instances with target value Y_j have attribute value a_i?

$$\hat{P}(a_i|Y_j) = 0, \text{ and...}$$

$$\hat{P}(Y_j)\prod_i \hat{P}(a_i|Y_j) = 0$$

- Naively setting zero probabilities to small number results in probabilities not summing to 1 (what if there are many attribute values).
- **Solution**: In Laplace smoothing, we assume **each attribute value** is observed in one added virtual instance and add as many virtual instances as there are **attribute values** to the denominator.

Laplace Smoothing

Outlook	Play=*Yes*	Play=*No*
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Humidity	Play=*Yes*	Play=*No*
High	3/9	4/5
Normal	6/9	1/5

Laplace Smoothing

Outlook	Play=*Yes*	Play=*No*
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Humidity	Play=*Yes*	Play=N*o*
High	3/9	4/5
Normal	6/9	1/5

P(O=Sunny | Play = Yes) = (2+1)/(9+3) = 3/12

P(O=Overcast | Play = Yes) = (4+1)/(9+3) = 5/12

P(O=Rain | Play = Yes) = (3+1)/(9+3) = 4/12

P(O=Sunny | Play = No) = (3+1)/(5+3) = 4/8

P(O=Overcast | Play = No) = (0+1)/(5+3) = 1/8

P(O=Rain | Play = No) = (2+1)/(5+3) = 3/8

Naïve Bayes subleties

3. Naive Bayes posteriors often unrealistically close to 1 or 0 ☹

   This is a common problem, not very specific to NB.

3. Naive Bayes is not affected by missing attribute values! ☺

NB Missing Values

- How can we do that?
- Based on conditional independence. Assume X_j is missing:

$$
\begin{aligned}
\mathbf{P}\left(X_1, \ldots, X_j, ..., X_n \mid Y\right) &= \mathbf{P}\left(X_1 \mid Y\right) \ldots \mathbf{P}\left(X_j \mid Y\right) \ldots \mathbf{P}\left(X_n \mid Y\right) \\
&= \mathbf{P}\left(X_1 \mid Y\right) \ldots \sum_{x_j} \mathbf{P}\left(X_j = x_j \mid Y\right) \ldots \mathbf{P}\left(X_n \mid Y\right) \\
&= \mathbf{P}\left(X_1 \mid Y\right) \ldots 1 \ldots \mathbf{P}\left(X_n \mid Y\right)
\end{aligned}
$$

- So we can just ignore the missing value/dimension..

NB Multi-class classification

- If we have more than one class, it naturally handles multiple classes

Practical Detail

- We are multiplying lots of small numbers. Danger of underflow!

- Underflow occurs when you perform an operation that's smaller than the smallest magnitude non-zero number.
- Solution:
  – $p1 * p2 = e^{\log(p1)+\log(p2)}$
  – Perform all computations by summing logs of probabilities rather than multiplying probabilities.

Incremental Updates

- Training is fast (linear in the number of examples, features and classes)

- If the model is going to be updated very often as new data come, you may implement it such that it allows cheap incremental updates.

- For example: Store raw counts instead of probabilities
  - New example of class k:
    - For each feature update the counts based on the example feature vector
    - Update the class counts, update the number of training data
    - When need to classify compute the probabilities

Naive Bayes

- Very fast, low storage requirements
- Robust to irrelevant features
- Optimal if the conditional independence assumptions hold
- A good dependable baseline for text classification