Naïve bayes

Hongyu Guo

Guo7407@saskpolytech.ca

Winter 2024

- Let I be the event that you have a deadly incurable disease, called *incuritis*.
    - $I = 1$ means you have it; $I = 0$ means you don't.
    - Abbreviation: i, $\neg i$.
- Let K represent the event that your knee is itchy.
    - $K = 1$ means your knees itch; $K = 0$ the opposite.
    - Abbreviation: k, $\neg k$.
- Medical science tells us that 80% of people who have incuritis also suffer from itchy knees.

$$P(k|i) = 0.8$$

- But you're not terrified, because you know Bayes Rule:

$$P(i|k) \quad = \quad \frac{P(k|i)P(i)}{P(k)}$$

- You consult medical clinicians, and you are told $P(i) = 10^{-5}$ and $P(k) = 0.5$
- A little math, and you show $P(i|k) = 1.6 \times 10^{-5}$
- Probably not incuritis after all!

Bayes' Rule

- The most important formula in probabilistic machine learning

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

(Super Easy) Derivation:

$$\boxed{\begin{aligned} P(A \wedge B) \\ P(B \wedge A) \end{aligned}} \begin{aligned} &= P(A|B) \times P(B) \\ &= P(B|A) \times P(A) \end{aligned}$$

Just set equal...

$$P(A|B) \times P(B) = P(B|A) \times P(A)$$

and solve...

Bayes, Thomas (1763) An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London,* **53:370-418**

Common terminology for Bayes' Rule

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

- Posterior probability (distribution): $P(Y|\mathbf{X})$
  - The probability of Y after considering the data.
- Prior probability (distribution): $P(Y)$
  - The probability of Y before considering the data.
- Likelihood (distribution): $P(\mathbf{X}|Y)$
  - Describes how the data depends on Y.
- Normalization constant: $P(\mathbf{X})$
  - Always boring. Sometimes written α^{-1}.

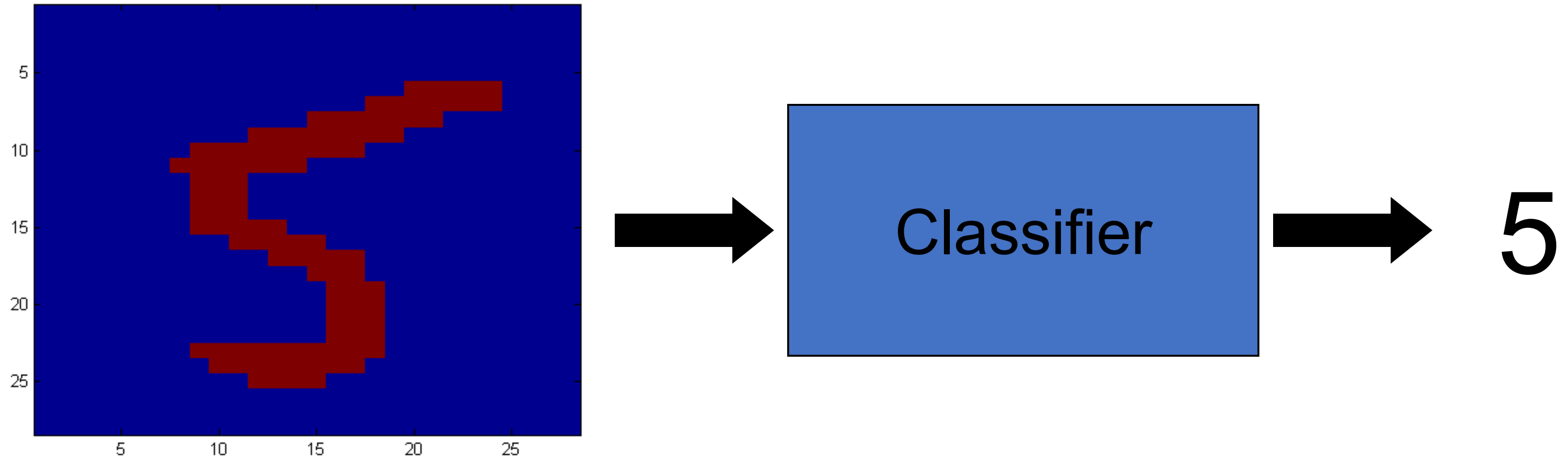
- The model comprises two types of probabilities that can be calculated directly from the training data:
  - the probability of each class
  - the conditional probability for each class given each *x* value.

Bayesian Classification

- Problem statement:
  - Given features $X_1, X_2, \ldots, X_n$
  - Predict a label Y

Another Application

- **Digit Recognition**



- **$X_1,\ldots,X_n \in \{0,1\}$ (Black vs. White pixels)**
- **$Y \in \{5,6\}$ (predict whether a digit is a 5 or a 6)**

The Bayes Classifier

- A good strategy is to predict:

$$\arg\max_{Y} P(Y|X_1, \ldots, X_n)$$

  - (for example: what is the probability that the image represents a 5 given its pixels?)

- So … How do we compute that?

The Bayes Classifier

- Use Bayes Rule!

$$P(Y|X_1,\ldots,X_n) = \frac{\overbrace{P(X_1,\ldots,X_n|Y)}^{\text{Likelihood}}\,\overbrace{P(Y)}^{\text{Prior}}}{\underbrace{P(X_1,\ldots,X_n)}_{\text{Normalization Constant}}}$$

- Why did this help? Well, we think that we might be able to specify how features are “generated” by the class label

The Bayes Classifier

- Let's expand this for our digit recognition task:

$$
\begin{aligned}
P(Y=5|X_1,\ldots,X_n) &= \frac{P(X_1,\ldots,X_n|Y=5)P(Y=5)}{P(X_1,\ldots,X_n|Y=5)P(Y=5)+P(X_1,\ldots,X_n|Y=6)P(Y=6)} \\
P(Y=6|X_1,\ldots,X_n) &= \frac{P(X_1,\ldots,X_n|Y=6)P(Y=6)}{P(X_1,\ldots,X_n|Y=5)P(Y=5)+P(X_1,\ldots,X_n|Y=6)P(Y=6)}
\end{aligned}
$$

- To classify, we'll simply compute these two probabilities and predict based on which one is greater

Model Parameters

- How many parameters are required to specify the likelihood?
  - (Supposing that each image is 30x30 pixels)

?

Model Parameters

- The problem with explicitly modeling $P(X_1,\ldots,X_n|Y)$ is that there are usually way too many parameters:
  - We'll run out of space
  - We'll run out of time
  - And we'll need tons of training data (which is usually not available)

The Naïve Bayes Model

- The *Naïve Bayes Assumption*: Assume that all features are independent **given the class label Y**
- Equationally speaking:

$$P(X_1, \ldots, X_n | Y) = \prod_{i=1}^{n} P(X_i | Y)$$

Naïve Bayes Example

$$v_{NB} = argmax_{v_j \in V} \mathrm{P}(\mathrm{v_j}) \prod_i P(x_i|v_j)$$

Day	Outlook	Temperature	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

Estimating Probabilities

- $v_{NB} = argmax_{v \in \{yes, no\}} \mathrm{P(v)} \prod_i P(x_i = observation | v)$
- How do we estimate $P(observation | v)$?

Example

$$v_{NB} = argmax_{v_j \in V} \mathrm{P}(\mathrm{v_j}) \prod_i P(x_i|v_j)$$

- Compute $P(PlayTennis = \ yes)$; $P(PlayTennis = \ no)$
- Compute $P(outlook = \ s/oc/r \ | PlayTennis = \ yes/no)$ (6 numbers)
- Compute $P(Temp = \ h/mild/cool \,| PlayTennis = \ yes/no)$ (6 numbers)
- Compute $P(humidity = \ hi/nor \,| PlayTennis = \ yes/no)$ (4 numbers)
- Compute $P(wind = \ w/st \,| PlayTennis = \ yes/no)$ (4 numbers)

Example

$$v_{NB} = argmax_{v_j \in V} P(v_j) \prod_i P(x_i | v_j)$$

- Compute $P(PlayTennis = \ yes)$; $P(PlayTennis = \ no)$
- Compute $P(outlook = \ s/oc/r \mid PlayTennis = \ yes/no)$ (6 numbers)
- Compute $P(Temp = \ h/mild/cool \mid PlayTennis = \ yes/no)$ (6 numbers)
- Compute $P(humidity = \ hi/nor \mid PlayTennis = \ yes/no)$ (4 numbers)
- Compute $P(wind = \ w/st \mid PlayTennis = \ yes/no)$ (4 numbers)

- Given a new instance:

  (Outlook=sunny; Temperature=cool; Humidity=high; Wind=strong)

- Predict: $PlayTennis = ?$

Example

$$v_{NB} = argmax_{v_j \in V} P(v_j) \prod_i P(x_i | v_j)$$

- Given: (Outlook=sunny; Temperature=cool; Humidity=high; Wind=strong)
- $P(PlayTennis = yes) = 9/14 = 0.64)$ $\quad P(PlayTennis = no) = 5/14 = 0.36$

- $P(outlook = sunny \mid yes) = 2/9$ $\quad P(outlook = sunny \mid no) = 3/5$
- $P(temp = cool \mid yes) = 3/9$ $\quad P(temp = cool \mid no) = 1/5$
- $P(humidity = hi \mid yes) = 3/9$ $\quad P(humidity = hi \mid no) = 4/5$
- $P(wind = strong \mid yes) = 3/9$ $\quad P(wind = strong \mid no) = 3/5$

- $P(yes, \ldots\ldots) \sim 0.0053$ $\quad P(no, \ldots\ldots) \sim 0.0206$

Example

$$v_{NB} = argmax_{v_j \in V} P(v_j) \prod_i P(x_i | v_j)$$

- Given: (Outlook=sunny; Temperature=cool; Humidity=high; Wind=strong)
- $P(PlayTennis = yes) = 9/14 = 0.64)$ $\quad$ $P(PlayTennis = no) = 5/14 = 0.36$

- $P(outlook = sunny \mid yes) = 2/9$ $\quad$ $P(outlook = sunny \mid no) = 3/5$
- $P(temp = cool \mid yes) = 3/9$ $\quad$ $P(temp = cool \mid no) = 1/5$
- $P(humidity = hi \mid yes) = 3/9$ $\quad$ $P(humidity = hi \mid no) = 4/5$
- $P(wind = strong \mid yes) = 3/9$ $\quad$ $P(wind = strong \mid no) = 3/5$

- $P(yes, \ldots\ldots) \sim 0.0053$ $\quad$ $P(no, \ldots\ldots) \sim 0.0206$
- $P(no|instance) = 0.0206/(0.0053 + 0.0206) = 0.795$

What if we were asked about Outlook=OC ?

- Advantages of Using Naive Bayes Classifier
  - Simple to Implement. The conditional probabilities are easy to evaluate.
  - Very fast – no iterations since the probabilities can be directly computed.
  - If the conditional Independence assumption holds, it could give great results.
- Disadvantages of Using Naive Bayes Classifier
  - Conditional Independence Assumption does not always hold. In most situations, the feature show some form of dependency.

- Naive Bayes is called naive because it assumes that each input variable is independent.
- This is a strong assumption and unrealistic for real data; however, the technique is very effective on a large range of complex problems.

Learning Resources

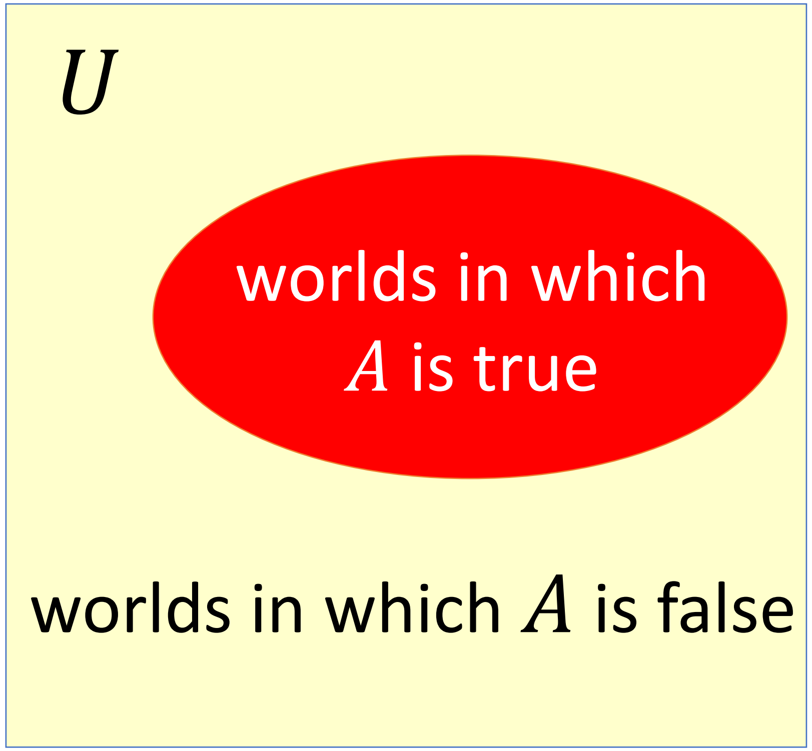
1. https://brilliant.org/wiki/classification/
2. https://www.toptal.com/machine-learning/machine-learning-theory-an-introductory-primer
3. https://developers.google.com/machine-learning/problem-framing/cases
4. https://learning.oreilly.com/library/view/hyperparameter-tuning-with/9781803235875/B18753_02_ePub.xhtml
5. https://learning.oreilly.com/library/view/hyperparameter-tuning-with/9781803235875/B18753_03_ePub.xhtml
6. https://scikit-learn.org/stable/
7. https://learning.oreilly.com/library/view/hands-on-machine-learning/9781492032632/ch04.html#idm45022192214984

Probability

Probability

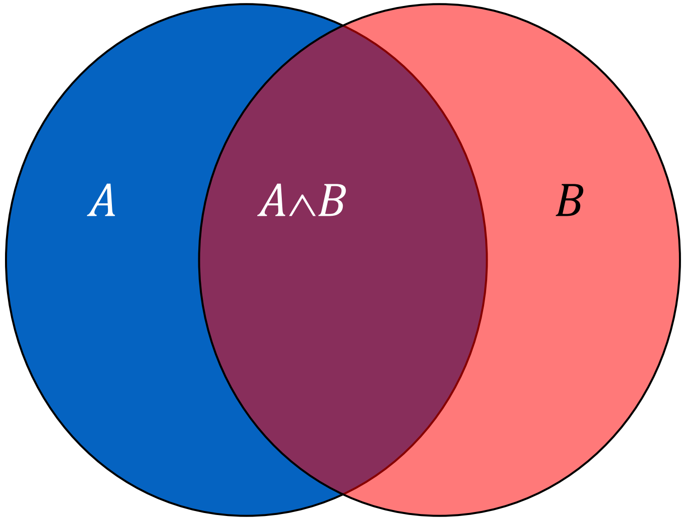
- Universe U is the event space of all possible worlds
  - Its area is 1
  - $P(U) = 1$
- $P(A)$ = area of red oval
- Therefore:

$$P(A) + P(\neg A) = 1$$
$$P(\neg A) = 1 - P(A)$$



Interpreting the Axioms

- $0 \leq P(A) \leq 1$
- $P(true) = 1$
- $P(false) = 0$
- $P(A \vee B) = P(A) + P(B) - P(A \wedge B)$



- From these you can prove other properties:
- $P(\neg A) = 1 - P(A)$
- $P(A) = P(A \wedge B) + P(A \wedge \neg B)$

Example: Conditional Probabilities

$$P(A|B) = \frac{P(A \wedge B)}{P(B)}$$

$$P(A \wedge B) = P(A|B) \times P(B)$$

$P(alarm, burglary)$ =

	alarm	¬alarm
burglary	0.09	0.01
¬burglary	0.1	0.8

$$P(burglary \mid alarm) = P(burglary \wedge alarm) \,/\, P(alarm) = 0.09 \,/\, 0.19 = 0.47$$

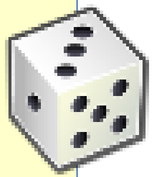
$$P(alarm \mid burglary) = P(burglary \wedge alarm) \,/\, P(burglary) = 0.09/0.1 = 0.9$$

$$P(burglary \wedge alarm) = P(burglary \mid alarm)\, P(alarm) = 0.47 * 0.19 = 0.09$$

Independence

- When two event do not affect each others' probabilities, we call them independent
- Formal definition:

$$A \perp\!\!\!\perp B \quad \leftrightarrow P(A \wedge B) = P(A) \times P(B)$$
$$\leftrightarrow P(A|B) = P(A)$$



Exercise: Independence

P(smart ∧ study ∧ prep)	smart		¬smart	
	study	**¬ study**	**study**	**¬ study**
prepared	0.432	0.16	0.084	0.008
¬prepared	0.048	0.16	0.036	0.072

Is *smart* independent of *study*?

$$P(study \wedge smart) = 0.432 + 0.048 = \boxed{0.48}$$
$$P(study) = 0.432 + 0.048 + 0.084 + 0.036 = 0.6$$
$$P(smart) = 0.432 + 0.048 + 0.16 + 0.16 = 0.8$$
$$P(study) \times P(smart) = 0.6 \times 0.8 = \boxed{0.48}$$

So yes!

Is *prepared* independent of *study*?

Bayes' Rule for Machine Learning

- Allows us to reason from evidence to hypotheses
- Another way of thinking about Bayes' rule:

$$\boxed{P(\text{hypothesis} \mid \text{evidence}) = \frac{P(\text{evidence} \mid \text{hypothesis}) \times P(\text{hypothesis})}{P(\text{evidence})}}$$

Goal:

find the best hypothesis from some space H of hypotheses, given the observed data (evidence) D.

Bayesian Classifier

- $f: \boldsymbol{X} \to V$, finite set of values
- Instances $\boldsymbol{x} \in \boldsymbol{X}$ can be described as a collection of features

$$\boldsymbol{x} = (x_1, x_2, \dots x_n) \quad x_i \in \{0,1\}$$

- Given an example, assign it the most probable value in V
- Bayes Rule:

$$v_{MAP} = argmax_{v_j \in V} P(v_j|x) = argmax_{v_j \in V} P(v_j|x_1, x_2, \dots x_n)$$

$$v_{MAP} = argmax_{v_j \in V} \frac{P(x_1, x_2, \dots, x_n|v_j)P(v_j)}{P(x_1, x_2, \dots, x_n)} = argmax_{v_j \in V} P(x_1, x_2, \dots, x_n|v_j)P(v_j)$$

- Notational convention: $P(y)$ means $P(Y = y)$

Naive Bayes

$$V_{MAP} = argmax_v \, P(x_1, x_2, \dots, x_n \mid v)P(v)$$

$$P(x_1, x_2, \dots, x_n | v_j) = P(x_1 | x_2, \dots, x_n, v_j) P(x_2, \dots, x_n | v_j)$$
$$= P(x_1 | x_2, \dots, x_n, v_j) \, P(x_2 | x_3, \dots, x_n, v_j) P(x_3, \dots, x_n | v_j)$$
$$= \cdots$$
$$= P(x_1 | x_2, \dots, x_n, v_j) \, P(x_2 | x_3, \dots, x_n, v_j) P(x_3 | x_4, \dots, x_n, v_j) \dots P(x_n | v_j)$$
$$= \prod_{i=1}^{n} P(x_i | v_j)$$

- Assumption: feature values are independent given the target value

Naive Bayes (2)

$$V_{MAP} = argmax_v\, P(x_1, x_2, \dots, x_n \mid v)P(v)$$

- Assumption: feature values are <u>independent given the target value</u>

$$P(x_1 = b_1, x_2 = b_2, \dots, x_n = b_n | v = v_j) \prod_{i=1}^{n} (x_i = b_i | v = v_j)$$

- Generative model:
- First choose a value $v_j \in V$ according to $P(v)$
- For each v_j : choose $x_1\, x_2, \dots, x_n$ according to $P(x_k | v_j)$

Naive Bayes (3)

$$V_{MAP} = argmax_v \, P(x_1, x_2, \ldots, x_n \mid v) P(v)$$

- Assumption: feature values are <u>independent given the target value</u>

$$P(x_1 = b_1, x_2 = b_2, \ldots, x_n = b_n | v = v_j) \prod_{i=1}^{n} (x_i = b_i | v = v_j)$$

- Learning method: Estimate $n|V| + |V|$ parameters and use them to make a prediction. (How to estimate?)
- Notice that this is learning without search. Given a collection of training examples, you just compute the best hypothesis (given the assumptions).
- This is learning without trying to achieve consistency or even approximate consistency.
- Why does it work?