CS 581 – Advanced Artificial Intelligence

Topic: Naïve Bayes

Mustafa Bilgic

http://www.cs.iit.edu/~mbilgic

CLASSIFICATION

- Given a dataset $\mathcal{D} = \left\{ \left\langle \vec{X}[m], Y[m] \right\rangle \right\}$ where
  - $\vec{X}$ is the input
  - Y is the discrete-valued output
- Learn a function $f\left(\vec{X}\right) \rightarrow Y$
- We would like f to **generalize** to **unseen** data
  - As opposed to memorizing the given data

Classification Examples

- Email classification
- Medical diagnosis
- Face recognition
- Optical character/digit recognition
- Sentiment classification
- …

Algorithms

- Decision trees
- Nearest neighbor classification
- Naïve Bayes
- Logistic regression
- Support vector machines
- Neural networks
- …

Bayes Classifier

- Input: $\vec{X} = \langle X_1, X_2, \dots, X_n \rangle$
- Output: Y
- Bayes classifier

$$P(Y \mid \vec{X}) = \frac{P(\vec{X} \mid Y)P(Y)}{P(\vec{X})} = \frac{P(Y)P(X_1, X_2, \dots, X_n \mid Y)}{P(X_1, X_2, \dots, X_n)}$$

$$P(X_1, X_2, \dots, X_n) = \sum_{y} P(Y = y)P(X_1, X_2, \dots, X_n \mid Y = y)$$

Assuming all variables are binary, how many independent parameters are needed for the Bayes classifier?

Naïve Bayes Assumption

$$X_i \perp X_j \mid Y$$

NAÏVE BAYES

Bayes rule:

$$P(Y \mid X_1, X_2, \dots, X_n) = \frac{P(Y)P(X_1, X_2, \dots, X_n \mid Y)}{\sum_y P(y)P(X_1, X_2, \dots, X_n \mid y)}$$

Assuming $X_i \perp X_j \mid Y$,
naïve Bayes:

$$P(Y \mid X_1, X_2, \dots, X_n) = \frac{P(Y)\prod P(X_i \mid Y)}{\sum_y P(y)\prod P(X_i \mid y)}$$

Assuming all variables are binary, how many independent parameters are needed for the naive Bayes classifier?

EXAMPLE

- See OneNote

Parameter Estimation

- Given a dataset $\mathcal{D} = \left\{ \left\langle \vec{X}[m], Y[m] \right\rangle \right\}$, how can we estimate
  - $P(Y)$
  - $P(X_i \mid Y)$
- Intuitive idea: count and normalize
  - But, why is this the right idea? Or, is it even the right idea?

Excursion

- Maximum likelihood estimation

Zero Probabilities

- Assume
  - $P(X_i = T|yes) = 0$ and $P(X_i = T|no) > 0$; and
  - $P(X_j \mid yes) > 0$ and $P(X_j \mid no) > 0$ for all other features
  - What is $P\left(yes \mid \vec{X}\right)$ if $X_i = T$?
- Assume
  - $P(X_i = T|yes) = 0$ and $P(X_i = T|no) = 0$; and
  - $P(X_j \mid yes) > 0$ and $P(X_j \mid no) > 0$ for all other features
  - What is $P\left(yes \mid \vec{X}\right)$ if $X_i = T$?
- Are these common?
- What can we do?

EXCURSION

- Bayesian estimation

Multiplying Several Probability Values

- Assume we have 1,000 features
- What is the product of 1,001 probability values?
  - `p = np.random.random(1001)`
  - `p_c = np.clip(p, 0.01, 0.99)`
  - `print(np.product(p_c))`
- In naïve Bayes,
  - $a = P(Y = T) \prod P(X_i | Y = T)$
  - $b = P(Y = F) \prod P(X_i | Y = F)$
  - $P\left(Y = T \middle| \vec{X}\right) = \frac{a}{a+b}$
  - If $a = b = 0$ in your code, then what?

Converting LogJoint to Cond Probs

- $\log\left(P\left(Y = yes, \vec{X}\right)\right) =$
  - $\log(P(Y = yes)) + \sum \log(P(X_i \mid Y = yes))$
- $\log\left(P\left(Y = no, \vec{X}\right)\right) =$
  - $\log(P(Y = no)) + \sum \log(P(X_i \mid Y = no))$
- What is $P\left(Y = yes \mid \vec{X}\right)$?
- Calculate it without causing an underflow.
  - You cannot use `np.exp(`$\log\left(P\left(Y = yes, \vec{X}\right)\right)$`)`
  - For example, try `np.exp(-1000)`

Converting LogJoint to Cond Probs

- $\log\left(P\left(Y = yes, \vec{X}\right)\right) = -1001$
- $\log\left(P\left(Y = no, \vec{X}\right)\right) = -1002$
- $P\left(Y = yes \mid \vec{X}\right) = ?$

Naïve Bayes Implementations

- Bernoulli / categorical naïve Bayes
  - Features are assumed to be binary / categorical
- Multinomial naïve Bayes
  - $P(\vec{X} \mid y)$ is a multinomial distribution
- Gaussian naïve Bayes
  - Each $p(x_i \mid y)$ is a Gaussian distribution

Reading

- http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf
- https://scikit-learn.org/stable/modules/naive_bayes.html