CZ4041/CE4041: Machine Learning

Lesson 3: Naïve Bayes Classifiers

Sinno Jialin PAN

School of Computer Science and Engineering, NTU, Singapore

Acknowledgements: some tables are adapted from the lecture notes of the book "Introduction to Data Mining"

Bayesian Classifiers: Recall

- To learn a prediction function via $P(y|\boldsymbol{x})$ using Bayes rule

$$P(y=c|\boldsymbol{x}) = \frac{P(\boldsymbol{x}, y=c)}{P(\boldsymbol{x})} = \frac{P(\boldsymbol{x}|y=c)P(y=c)}{P(\boldsymbol{x})}$$

- Make predictions based on maximum posterior

$$y^* = c^* \text{ if } c^* = \arg\max_c \frac{P(\boldsymbol{x}|y=c)P(y=c)}{P(\boldsymbol{x})}$$

the 0/1 loss

$$\Updownarrow$$

$$y^* = c^* \text{ if } c^* = \arg\max_c \boxed{P(\boldsymbol{x}|y=c)}\boxed{P(y=c)}$$

Easy to estimate

Still difficult to estimate as $\boldsymbol{x}$ contains many input variables, some are discrete, and others are continuous

Naïve Bayes Classifiers

- How to estimate $P(\boldsymbol{x}|y)$ from the training data
- Assume that the features are conditionally independent given the class label:

$$P(\boldsymbol{x}|y=c)=\prod_{i=1}^{d}P(x_i|y=c)$$

where $\boldsymbol{x}=[x_1,x_2,\ldots,x_d]$

$$P(x_1,x_2,\ldots,x_d|y=c)=\prod_{i=1}^{d}P(x_i|y=c)$$

Independence

- Let A and B be two random variables
- A is said to be independent of B, if the following condition holds:

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

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

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

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

- E.g., let A and B denote the results of two matches in different leagues, knowing the result of one match (e.g., the value of A) does not affect the possibility of result for the other match (i.e., the value of B)

Independence (cont.)

A more general case:

- Let **A** and **B** be two sets of random variables
- The variables in **A** are said to be independent of the variables in **B**, if the following condition holds:

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

$$P(\mathbf{A}|\mathbf{B}) = P(\mathbf{A})$$

Conditional Independence

- Let A, B and C be three random variables
- A is said to be conditionally independent of B, given C, if the following condition holds:

$$P(A|B,C) = P(A|C)$$

Conditional Independence (cont.)

A more general case

- Let **A**, **B**, and **C** be three <u>sets</u> of random variables.
- The variables in **A** are said to be <u>conditionally independent</u> of the variables in **B**, given the variables in **C** are observed, if the following condition holds:

$$P(\mathbf{A}|\mathbf{B},\mathbf{C}) = P(\mathbf{A}|\mathbf{C})$$

$$P(\boldsymbol{x}|y=c) = \prod_{i=1}^{d} P(x_i|y=c)$$

Conditional Independence (cont.)

- The conditional independence between **A** and **B** given **C** can also be written as follows

$$P(\mathbf{V}|\mathbf{C}) = \frac{P(\mathbf{V},\mathbf{C})}{P(\mathbf{C})}$$

where $\mathbf{V} = \{\mathbf{A},\mathbf{B}\}$

$$\boxed{P(\mathbf{A},\mathbf{B}|\mathbf{C})} = \frac{P(\mathbf{A},\mathbf{B},\mathbf{C})}{P(\mathbf{C})}$$ Product rule

$$\frac{P(\mathbf{A},\mathbf{U})}{P(\mathbf{U})} = P(\mathbf{A}|\mathbf{U})$$

where $\mathbf{U} = \{\mathbf{B},\mathbf{C}\}$

$$= \boxed{\frac{P(\mathbf{A},\mathbf{B},\mathbf{C})}{\boxed{P(\mathbf{B},\mathbf{C})}}} \times \frac{\boxed{P(\mathbf{B},\mathbf{C})}}{P(\mathbf{C})}$$

$$= P(\mathbf{A}|\mathbf{B},\mathbf{C}) \times P(\mathbf{B}|\mathbf{C})$$ Product rule

$$= P(\mathbf{A}|\mathbf{C}) \times P(\mathbf{B}|\mathbf{C})$$ Conditional independence

Naïve Bayes – Induction

- The conditional independence between **A** and **B** given **C** can also be written as follows

$$P(\mathbf{A}, \mathbf{B}|\mathbf{C}) = P(\mathbf{A}|\mathbf{C}) \times P(\mathbf{B}|\mathbf{C})$$

Naïve Bayes Classifier: $P(\boldsymbol{x}|y = c) = \prod_{i=1}^{d} P(x_i|y = c)$

Assume that the features are conditionally independent given the class label

Naïve Bayes – Induction (cont.)

$$P(\boldsymbol{x}|y=c) = P(x_1, x_2, \ldots, x_d|y=c)$$

Define $\mathbf{X}^{(d-1)} = [x_1, \ldots, x_{d-1}]$

$$= P\left(\mathbf{X}^{(d-1)}, x_d \middle| y=c\right)$$

Features are conditionally independent given class label

$$= P\left(\mathbf{X}^{(d-1)} \middle| y=c\right) P(x_d|y=c)$$

Define $\mathbf{X}^{(d-2)} = [x_1, \ldots, x_{d-2}]$

Features are conditionally independent given class label

$$= P\left(\mathbf{X}^{(d-2)}, x_{d-1} \middle| y=c\right) P(x_d|y=c)$$

$$= P\left(\mathbf{X}^{(d-2)} \middle| y=c\right) P(x_{d-1}|y=c) P(x_d|y=c)$$

Recursively apply conditional independence

$$= P(x_1|y=c)P(x_2|y=c) \cdots P(x_d|y=c)$$

$$= \prod_{i=1}^{d} P(x_i|y=c)$$

How Naïve Bayes Classifier Work

Naïve Bayes Classifier: $P(\boldsymbol{x}|y=c) = \prod_{i=1}^{d} \boxed{P(x_i|y=c)}$ Easier to estimate

- To classify a test record $\boldsymbol{x}^*$, we need to compute the posteriors for each class by using

$$P(y=c|\boldsymbol{x}^*) = \frac{\left(\prod_{i=1}^{d} P(x_i^*|y=c)\right)P(y=c)}{P(\boldsymbol{x}^*)}$$

- $P(\boldsymbol{x}^*)$ is constant for each class c, it is sufficient to choose the class that maximizes the numerator term

$$\left(\prod_{i=1}^{d} P(x_i^*|y=c)\right)P(y=c)$$

Illustrative Example

- Consider the problem of predicting whether a loan applicant will repay his/her loan obligation (no cheat) or become delinquent (cheat).

Predefined categories

Example

Example (cont.)

Tid	**Home Owner**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

Training data

$P(y|\boldsymbol{x})$

Home Owner	**Marital Status**	**Taxable Income**	**Cheat**
No	Married	120K	**?**

Test application $\boldsymbol{x}^*$

- To classify the application, we need to compute the posterior probabilities $P(\text{Yes}|\boldsymbol{x}^*)$ and $P(\text{No}|\boldsymbol{x}^*)$
- If $P(\text{Yes}|\boldsymbol{x}^*) > P(\text{No}|\boldsymbol{x}^*)$ then classified as Yes
- Otherwise classified No

Estimate Priors

- Class:

#instances in class *c*

$$P(y=c)=\frac{\boxed{|y=c|}}{\boxed{N}}$$

#training instances

- e.g.,

$$P(\text{No})=0.7$$

$$P(\text{Yes})=0.3$$

Tid	**Home Owner**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

Estimate Conditional Probabilities for Discrete Features

#instances in class c, whose values of feature x_i are k

$$P(x_i = k | y = c) = \frac{|(x_i = k) \wedge (y = c)|}{|y = c|}$$

A specific value k of the feature x_i

Tid	Home Owner	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

$$P(\text{Status} = \text{Married} | \text{Cheat} = \text{No})$$

$$= \frac{\#(\text{Status} = \text{Married} \wedge \text{Cheat} = \text{No})}{\#(\text{Cheat} = \text{No})} = \frac{4}{7}$$

$$P(\text{Home Owner} = \text{Yes} | \text{Cheat} = \text{Yes})$$

$$= \frac{\#(\text{Home Owner} = \text{Yes} \wedge \text{Cheat} = \text{Yes})}{\#(\text{Cheat} = \text{Yes})} = \frac{0}{3} = 0$$

Estimate Conditional Probabilities for Continuous Features

- For continuous features:
  - Probability density estimation (more details will be introduced in the 2nd half of the semester):
    - Assume the values of a feature given a class label follows a Guassian distribution, i.e., assume $P(x_i|y = c)$ is a Guassian distribution
    - Use training data in the class c to estimate parameters of distribution (e.g., mean μ and variance σ^2)
    - Once probability density function is known, we can use it to estimate the conditional probability

Estimate Conditional Probabilities for Continuous Features (cont.)

- For each class c, assume values of the feature x_i follow a Gaussian distribution:

$$P(x_i|y=c) = \frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(x_i-\mu_{ic})^2}{2\sigma_{ic}^2}}$$

The mean of x_i of the training data in class c

The variance of x_i of the training data in class c

- Suppose there are N_c instances in class c, then

$$\mu_{ic} = \frac{1}{N_c}\sum_{j=1}^{N_c} \boxed{x_{i_j}} \qquad \sigma_{ic}^2 = \frac{1}{N_c-1}\sum_{j=1}^{N_c}\left(x_{i_j}-\mu_{ic}\right)^2$$

Value of feature x_i of the j-th training data in class c

Estimate Conditional Probabilities for Continuous Features (cont.)

$$P(x_i|y=c) = \frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(x_i-\mu_{ic})^2}{2\sigma_{ic}^2}}$$

- For {Income, Cheat = No}:
  - sample mean = 110
  - sample variance = 2975
    (standard deviation = 54.54)

Tid	**Home Owner**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

$$P(\text{Income}|\text{Cheat}=\text{No}) = \frac{1}{\sqrt{2\pi}\times 54.54} e^{-\frac{(\boxed{x_i}-110)^2}{2\times 2975}}$$

Income

Estimate Conditional Probabilities for Continuous Features (cont.)

- The estimated Gaussian distribution for {Income, Cheat = No}:

A function of x_i

$$\boxed{P(\text{Income}|\text{Cheat} = \text{No})} = \frac{1}{\sqrt{2\pi} \times 54.54} e^{-\frac{(\boxed{x_i}-110)^2}{2\times 2975}}$$

Tid	Home Owner	Marital Status	Taxable Income	Cheat
4	Yes	Married	120K	No

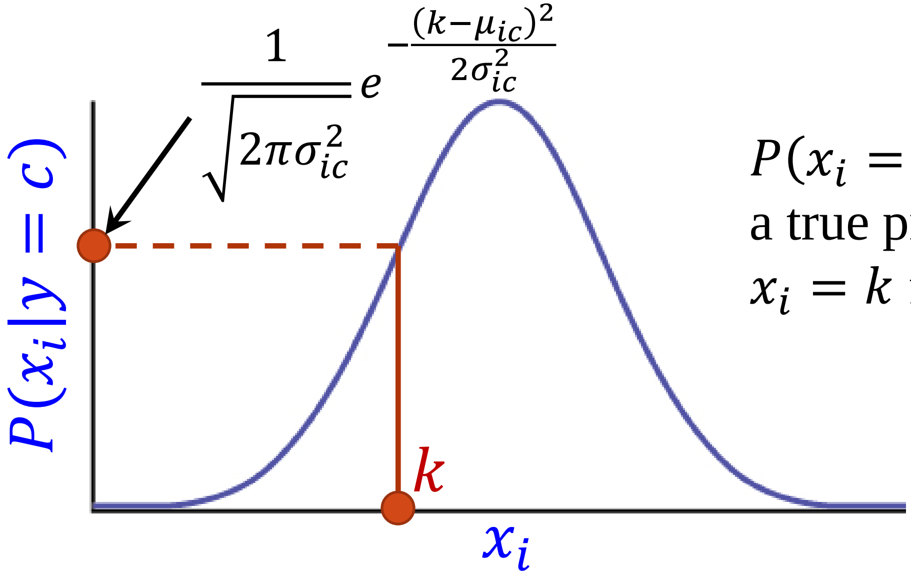
$$P(\text{Income} = 120|\text{No}) = \frac{1}{\sqrt{2\pi} \times 54.54} e^{-\frac{(\boxed{120}-110)^2}{2\times 2975}} = 0.0072$$

Note: in practice, 0.0072 can be used as to approximate the conditional probability, but in theory it is not a true probability

Additional Notes

Probability density function $P(x_i|y=c) = \frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(x_i-\mu_{ic})^2}{2\sigma_{ic}^2}}$

- The probability density function is continuous, the probability is defined as the area under the curve of the probability density function



$P(x_i = k|y = c)$ is not a true probability that $x_i = k$ for class c

Additional Notes (cont.)

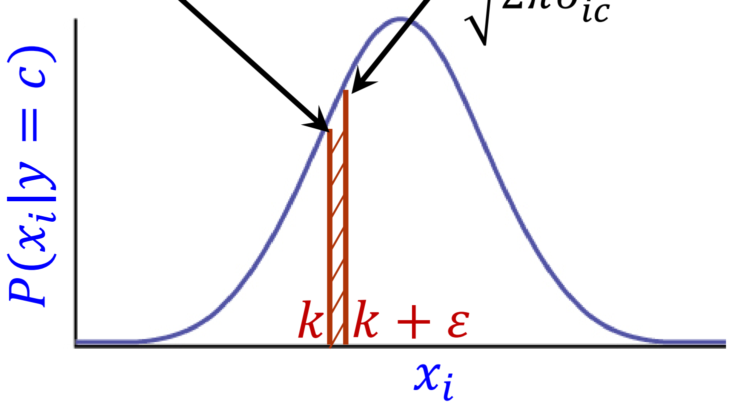
- Instead, we should compute

Small positive constant

$$P(k \le x_i \le k + \varepsilon | y = c) = \int_{k}^{k+\varepsilon} \frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(x_i-\mu_{ic})^2}{2\sigma_{ic}^2}} dx_i$$

$$\frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(k-\mu_{ic})^2}{2\sigma_{ic}^2}}$$

$$\frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(k+\varepsilon-\mu_{ic})^2}{2\sigma_{ic}^2}}$$



Assume $\frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(k-\mu_{ic})^2}{2\sigma_{ic}^2}} \approx \frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(k+\varepsilon-\mu_{ic})^2}{2\sigma_{ic}^2}}$

$$\approx \frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(k-\mu_{ic})^2}{2\sigma_{ic}^2}} \times \varepsilon$$

Additional Notes (cont.)

- Since ε appears as a constant multiplicative factor for each class, it cancels out when comparing posterior probabilities $P(y = c|\boldsymbol{x})$ for each class
- E.g., consider binary classification and instance is represented by a single feature of continues values

$$P(y = 0|x = k) \quad vs. \quad P(y = 1|x = k)$$

$$\Downarrow$$

$$P(x = k|y = 0)P(y = 0) \quad vs. \quad P(x = k|y = 1)P(y = 1)$$

$$\Downarrow$$

$$\frac{1}{\sqrt{2\pi\sigma_0^2}} e^{-\frac{(k-\mu_0)^2}{2\sigma_0^2}} \times \cancel{\varepsilon} \times P(y = 0) \quad vs. \quad \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{(k-\mu_1)^2}{2\sigma_1^2}} \times \cancel{\varepsilon} \times P(y = 1)$$

Additional Notes (cont.)

- Therefore, we can still apply the following equation to approximate the probability of $x_i = k$ for class c

$$P(x_i = k | y = c) = \frac{1}{\sqrt{2\pi\sigma_{ic}^2}} e^{-\frac{(k-\mu_{ic})^2}{2\sigma_{ic}^2}}$$

Example of Naïve Bayes Classifier

Naïve Bayes Classifier:

P(HomO=Yes|No) = 3/7
P(HomO=No|No) = 4/7
P(HomO=Yes|Yes) = 0
P(HomO=No|Yes) = 1

P(Marital Status = Single|No) = 2/7
P(Marital Status = Divorced|No)=1/7
P(Marital Status = Married|No) = 4/7
P(Marital Status = Single|Yes) = 2/3
P(Marital Status = Divorced|Yes)=1/3
P(Marital Status = Married|Yes) = 0

For taxable income:
If class=No: sample mean=110
sample variance=2975
If class=Yes: sample mean=90
sample variance=25

P(Class = No) = 7/10
P(Class = Yes) = 3/10

Tid	**Home Owner**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

$$P(\boldsymbol{x}|y=c) = \prod_{i=1}^{d} P(x_i|y=c)$$

Test example:

Home Owner	Marital Status	Taxable Income	Cheat
No	Married	120K	?

$$P(\boldsymbol{x}^*|y=c)=\prod_{i=1}^{d}P(x_i^*|y=c)$$

P(HomO=Yes|No) = 3/7
P(HomO=No|No) = 4/7
P(HomO=Yes|Yes) = 0
P(HomO=No|Yes) = 1

P(Marital Status = Single|No) = 2/7
P(Marital Status = Divorced|No)=1/7
P(Marital Status = Married|No) = 4/7
P(Marital Status = Single|Yes) = 2/3
P(Marital Status = Divorced|Yes)=1/3
P(Marital Status = Married|Yes) = 0

For taxable income:
If class=No: sample mean=110
sample variance=2975
If class=Yes: sample mean=90
sample variance=25

P(Class = No) = 7/10
P(Class = Yes) = 3/10

$$P(\boldsymbol{x}^*|\text{Class=No}) = P(\text{HomO=No}|\text{Class=No}) \times P(\text{Status=Married}|\text{Class=No}) \times P(\text{Income=120K}|\text{Class=No})$$

$$= \frac{4}{7}\times\frac{4}{7}\times 0.0072 = 0.0024$$

$$P(\boldsymbol{x}^*|\text{Class=Yes}) = P(\text{HomO=No}|\text{Class=Yes}) \times P(\text{Status=Married}|\text{Class=Yes}) \times P(\text{Income=120K}|\text{Class=Yes})$$

one of the conditional probability is 0, the entire expression is 0

$$= \boxed{1\times 0\times(1.2\ \times 10^{-9}) = 0}$$

$$P(\boldsymbol{x}^*|\text{No})\times P(\text{No}) = 0.0024\times 0.7 = 0.00168$$

$$> P(\boldsymbol{x}^*|\text{Yes})\times P(\text{Yes}) = 0\times 0.3 = 0$$

Therefore $P(\text{No}|\boldsymbol{x}^*) > P(\text{Yes}|\boldsymbol{x}^*)$ **Class = No**

Laplace Estimate

- Alternative probability estimation (discrete features):

Original: $P(x_i = k|y = c) = \frac{|(x_i = k) \wedge (y = c)|}{|y = c|}$

Laplace: $P(x_i = k|y = c) = \frac{|(x_i = k) \wedge (y = c)| + 1}{|y = c| + \boxed{n_i}}$

#distinct values of x_i

$P(\text{Married}|\text{Yes}) = \frac{\#(\text{Married} \wedge \text{Yes})}{\#(\text{Yes})} = \frac{0}{3}$

$P(\text{Married}|\text{Yes}) = \frac{\#(\text{Married} \wedge \text{Yes}) + 1}{\#(\text{Yes}) + 3} = \frac{1}{6}$

The same to $P(\text{Single}|\text{Yes})$ and $P(\text{Divorced}|\text{Yes})$

Extreme case - no training data:

$P(\text{Single}|\text{Yes}) = P(\text{Married}|\text{Yes}) = P(\text{Divorced}|\text{Yes}) = \frac{1}{3}$

Tid	**Home Owner**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

M-estimate

- A more general estimation:

User-specified parameters

Original: $P(x_i = k|y = c) = \frac{|(x_i = k) \land (y = c)|}{|y = c|}$

M-estimate: $P(x_i = k|y = c) = \frac{|(x_i = k) \land (y = c)| + \boxed{m} \times \boxed{p}}{|y = c| + m}$

e.g., if prior information of $P(x_i = k|y = c)$ is available, then we can set p as the prior

For example, based on domain knowledge, you have the prior information:

Domain knowledge, not learned from data

$$\tilde{P}(\text{Single}|\text{Yes}) = \frac{1}{2} \quad \tilde{P}(\text{Divorced}|\text{Yes}) = \frac{1}{3} \quad \tilde{P}(\text{Married}|\text{Yes}) = \frac{1}{6}$$

Extreme case - no training data:

$$P(\text{Single}|\text{Yes}) = \frac{\#(\text{Single} \land \text{Yes}) + m \times \tilde{P}(\text{Single}|\text{Yes})}{\#(\text{Yes}) + m} = \frac{m \times \tilde{P}(\text{Single}|\text{Yes})}{m} = \tilde{P}(\text{Single}|\text{Yes})$$

Home Owner	Marital Status	Taxable Income	Cheat
No	Married	120K	?

$m = 3$

$p = 1/3$ for all discrete features of class **Yes**

$p = 2/3$ for all discrete features of class **No**

P(HomO=Yes|No) = 3/7
P(HomO=No|No) = 4/7
P(HomO=Yes|Yes) = 0/3
P(HomO=No|Yes) = 1

P(Marital Status = Single|No) = 2/7
P(Marital Status = Divorced|No)=1/7
P(Marital Status = Married|No) = 4/7
P(Marital Status = Single|Yes) = 2/3
P(Marital Status = Divorced|Yes)=1/3
P(Marital Status = Married|Yes) = 0/3

For taxable income:
If class=No: sample mean=110
sample variance=2975
If class=Yes: sample mean=90
sample variance=25

P(Class = No) = 7/10
P(Class = Yes) = 3/10

P(HomO=Yes|No) = ?
P(HomO=No|No) = ?
P(HomO=Yes|Yes) = ?
P(HomO=No|Yes) = ?

P(Marital Status = Single|No) = ?
P(Marital Status = Divorced|No) = ?
P(Marital Status = Married|No) = ?
P(Marital Status = Single|Yes) = ?
P(Marital Status = Divorced|Yes)=?
P(Marital Status = Married|Yes) = ?

M-estimate

$$P(x_i = k|y = c) = \frac{|(x_i = k) \wedge (y = c)| + m \times p}{|y = c| + m}$$

$$P(\boldsymbol{x}^*|\text{Class} = \text{No}) = ? \quad P(\boldsymbol{x}^*|\text{Class} = \text{Yes}) = ?$$

Tutorial

Implementation using scikit-learn

- API: sklearn.naive_bayes: Naive Bayes
  https://scikit-learn.org/stable/modules/classes.html#module-sklearn.naive_bayes

sklearn.naive_bayes: Naive Bayes

The sklearn.naive_bayes module implements Naive Bayes algorithms. These are supervised learning methods based on applying Bayes' theorem with strong (naive) feature independence assumptions.

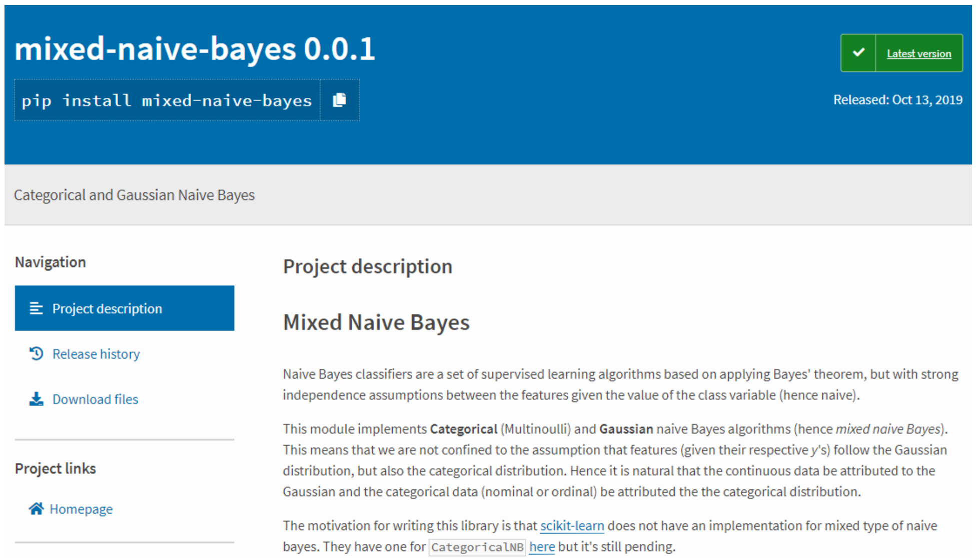
User guide: See the Naive Bayes section for further details.

naive_bayes.BernoulliNB(*[, alpha, ...])	Naive Bayes classifier for multivariate Bernoulli models.
naive_bayes.CategoricalNB(*[, alpha, ...])	Naive Bayes classifier for categorical features
naive_bayes.ComplementNB(*[, alpha, ...])	The Complement Naive Bayes classifier described in Rennie et al.
naive_bayes.GaussianNB(*[, priors, ...])	Gaussian Naive Bayes (GaussianNB)
naive_bayes.MultinomialNB(*[, alpha, ...])	Naive Bayes classifier for multinomial models

Documentation: https://scikit-learn.org/stable/modules/naive_bayes.html

Mixed Naïve Bayes Implementation

https://pypi.org/project/mixed-naive-bayes/#installation


mixed-naive-bayes 0.0.1

pip install mixed-naive-bayes

Latest version

Released: Oct 13, 2019

Categorical and Gaussian Naive Bayes

Navigation

Project description

Release history

Download files

Project links

Homepage

Project description

Mixed Naive Bayes

Naive Bayes classifiers are a set of supervised learning algorithms based on applying Bayes' theorem, but with strong independence assumptions between the features given the value of the class variable (hence naive).

This module implements **Categorical** (Multinoulli) and **Gaussian** naive Bayes algorithms (hence *mixed naive Bayes*). This means that we are not confined to the assumption that features (given their respective *y*'s) follow the Gaussian distribution, but also the categorical distribution. Hence it is natural that the continuous data be attributed to the Gaussian and the categorical data (nominal or ordinal) be attributed the the categorical distribution.

The motivation for writing this library is that scikit-learn does not have an implementation for mixed type of naive bayes. They have one for `CategoricalNB` here but it's still pending.

```
>>> from mixed_naive_bayes import MixedNB
```

```
>>> nbC = MixedNB(categorical_features=[0,1,3])
>>> nbC.fit(X, y)
>>> nbC.predict(X)
```

Specify which columns are categorical features

Naïve Bayes Classifier: Summary

- Based on a very strong assumption on conditional independence: all the input features are independent to each other given a class label

$$P(\boldsymbol{x}|y=c)=\prod_{i=1}^{d}P(x_i|y=c)$$

- Computationally efficient
- Independence assumption may not hold in practice (for most of time), that is why it is called “naïve”
  - Correlated features can degrade the performance
  - To be continued …

Thank you!