Data Valorization:
Naïve Bayes Classification

Lionel Fillatre

fillatre@unice.fr

Outline

- Introduction
- Probability Basics
- Naïve Bayes
- Discrete Case
- Continuous Case
- Conclusion

1 Introduction

Background

- There are three methods to establish a classifier

  a) Model a classification rule directly

  Examples: k-NN, decision trees, perceptron, SVM

  b) Model the probability of class memberships given input data

  Example: logistic regression, perceptron with the cross-entropy cost

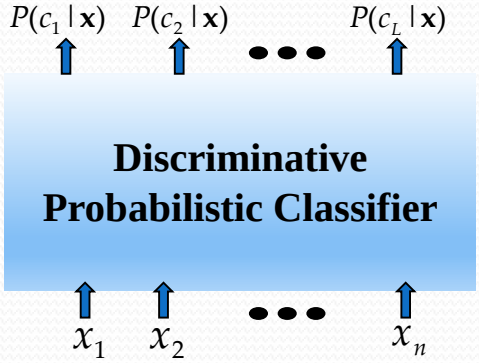
  c) Make a probabilistic model of data within each class

  Examples: naive Bayes, hypothesis testing

- *a*) and *b*) are examples of discriminative classification
- *c*) is an example of generative classification
- *b*) and *c*) are both examples of probabilistic classification

Probabilistic Classification

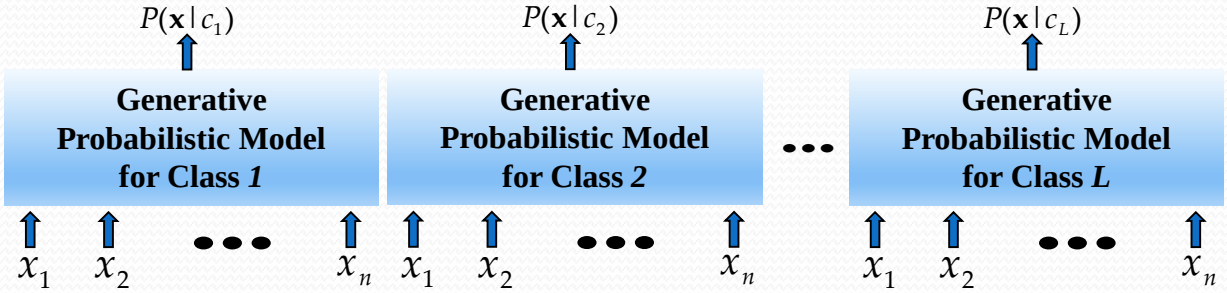
- Establishing a probabilistic model for classification
  - Discriminative model: $P(C|\boldsymbol{X}), \mathrm{C} \in \{c_1, \ldots, c_L\}, \boldsymbol{X} = (X_1, \ldots, X_n)$



$\boldsymbol{x} = (x_1, \ldots, x_n)$: data, features, etc.

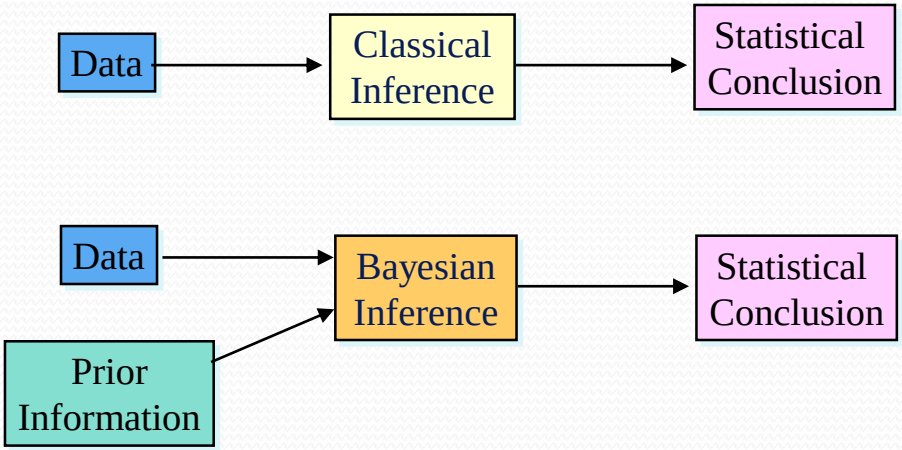
Probabilistic Classification

- Establishing a probabilistic model for classification
  - Generative model: $P(\boldsymbol{X}|C), \mathrm{C} \in \{c_1, \ldots, c_L\}, \boldsymbol{X} = (X_1, \ldots, X_n)$



$$\boldsymbol{x} = (x_1, \ldots, x_n)$$

Bayesian and Classical Statistics



Bayesian statistical analysis incorporates a prior probability distribution and likelihoods of observed data to determine a posterior probability distribution of events.

Hypothesis Testing: Bayes Approach

- In the previous lecture, we have assumed that the statistical model of the samples depends on some **non-random** classes C (or some unknown parameter θ)
- The Bayes approach assumes that the class is **random**
- The distribution $P(C)$ of the class is called the **prior distribution.**
  - It is equivalent to a prior distribution $P(\theta)$ on the parameter (in case there is a parameter defining each class).
- The prior distribution is
  - known
  - or assumed to be known
  - or estimated from a training data set

2 Probability Basics

Probability Basics

- Prior, conditional and joint probability for random variables
  - Prior probability: $P(C)$
  - Conditional probability: $P(X \mid C), P(C \mid X)$
  - Joint probability: $Z = (X, C), P(Z) = P(X, C)$
  - Relationship: $P(X, C) = P(X \mid C)P(C) = P(C \mid X)P(X)$
  - Independence: $P(X \mid C) = P(X), P(C \mid X) = P(C), P(X, C) = P(X)P(C)$

- Bayesian Rule

$$P(C \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid C)P(C)}{P(\mathbf{X})}$$

$$Posterior = \frac{Likelihood \times Prior}{Evidence}$$

Probability Basics

- We have two six-sided dice. When they are tolled, it could end up with the following occurance: (*A*) dice 1 lands on side "3", (*B*) dice 2 lands on side "1", and (*C*) Two dice sum to eight.

1) $P(A) = ?$
2) $P(B) = ?$
3) $P(C) = ?$
4) $P(A \mid B) = ?$
5) $P(C \mid A) = ?$
6) $P(A, B) = ?$
7) $P(A, C) = ?$
8) Is $P(A, C)$ equal to $P(A) * P(C)$?

Types of errors

Ground Truth	Decision $Z = 0$	Decision $Z = 1$
$C = 1$	False Negative (FN)	True Positive (TP)
$C = 0$	True Negative (TN)	False Positive (FP)

Example: rare disease detection

- A rare disease is affecting 0.1% of the population:
  $P(C = 1) = 0{,}001$ where $C = 1$ when the patient is ill

- Let us consider a medical test for this rare disease. The decision of the test is denoted Z: it indicates that the person is ill if $Z = 1$, otherwise $Z = 0$

- When administered to an ill person, the test will indicate so with probability 0.92 :
  - It follows that $P(Z = 1|C = 1) = 0{,}92 \Rightarrow P(Z = 0|C = 1) = 0{,}08$
  - The event$(Z = 0|C = 1)$ is a false negative (misdetection or type II error)

- When administered to a person who is not ill, the test will erroneously give a positive result with probability 0.04 :
  - It follows that $P(Z = 1|C = 0) = 0{,}04 \Rightarrow P(Z = 0|C = 0) = 0{,}96$
  - The event $(Z = 1|C = 0)$ is a false positive (false alarm or type I error).

Example (cont.): Conditional Probabilities

Known probabilities:

$P(C = 1) = 0.001$
$P(C = 0) = 0.999$

$P(Z = 1|C = 1) = 0.92$
$P(Z = 1|C = 0) = 0.04$
$P(Z = 0|C = 1) = 0{,}08$
$P(Z = 0|C = 0) = 0{,}96$

Conditional probabilities:

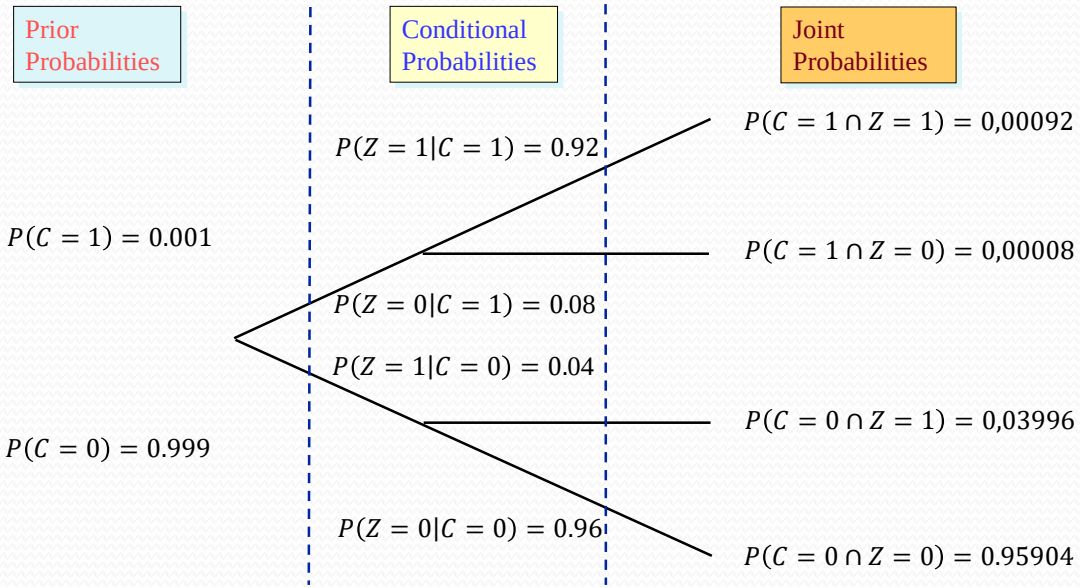
$P(C = 1|Z = 1) = 0{,}0225$
$P(C = 0|Z = 1) = 0.9775$
$P(C = 1|Z = 0) = 0.00008$
$P(C = 0|Z = 0) = 0{,}99992$

Proof:

$$P(C = 1|Z = 1) = \frac{P(C = 1 \cap Z = 1)}{P(Z = 1)} = \frac{P(C = 1 \cap Z = 1)}{P(C = 0 \cap Z = 1) + P(C = 1 \cap Z = 1)}$$

$$= \frac{P(Z = 1|C = 1)P(C = 1)}{P(Z = 1|C = 0)P(C = 0) + P(Z = 1|C = 1)P(C = 1)} = \frac{0.92 \times 0.001}{0.04 \times 0.999 + 0.92 \times 0.001} = 0.0225$$

Example (cont.): Error Probability



Global Decision Error

- The global decision error is $P(E) = P(Z \neq C)$

$$P(E) = P(Z = 0 \cap C = 1) + P(Z = 1 \cap C = 0)$$
$$= P(Z = 0|C = 1)P(C = 1) + P(Z = 1|C = 0)P(C = 0)$$

- This shows that

$$P(E) = \text{false negative} \times P(C = 1) + \text{false positive} \times P(C = 0)$$

- Optimality:
  - **The Bayes test (not the naïve one) minimizes $P(E)$**
  - In case of independent features, the naive Bayes test is optimal.

- Example (cont.):

$$P(E) = 0{,}00008 + 0{,}03996 = 0{,}04004$$

  - In this example, the two errors are quite different: why? Does it care?
  - $P(E) \approx P(Z = 1 \cap C = 0) \approx P(Z = 1|C = 0)$

3 Naïve Bayes

MAP Rule

- MAP classification rule minimizes the Bayes decision error
  - **MAP**: **M**aximum **A** **P**osterior
  - Assign $\boldsymbol{x}$ to c^* if

$$P(C = c^*|\boldsymbol{X} = \boldsymbol{x}) > P(C = c|\boldsymbol{X} = \boldsymbol{x}), \qquad c^* \neq \mathrm{c}, \qquad \mathrm{c} \in \{c_1, \dots, c_L\}$$

- Generative classification with the MAP rule
  - Apply Bayesian rule to calculate posterior probabilities for all c_i

$$P(C = c_i|\boldsymbol{X} = \boldsymbol{x}) = \frac{P(\boldsymbol{X} = \boldsymbol{x}|C = c_i)P(C = c_i)}{P(\boldsymbol{X} = \boldsymbol{x})} \propto P(\boldsymbol{X} = \boldsymbol{x}|C = c_i)P(C = c_i)$$

  - Then apply the MAP rule

Naïve Bayes

- Difficulty: learning the joint probability $P(\boldsymbol{X}_1, \dots, \boldsymbol{X}_n | C)$

$$P(C = c_i | \boldsymbol{X} = \boldsymbol{x}) \propto P(\boldsymbol{X} = \boldsymbol{x} | C = c_i) P(C = c_i) = P(\boldsymbol{X}_1 = \boldsymbol{x}_1, \dots, \boldsymbol{X}_n = \boldsymbol{x}_n | C = c_i) P(C = c_i)$$

- Naïve Bayes classification
  - **Assumption**: all input features are **conditionally independent**!

$$\begin{aligned} P(\boldsymbol{X}_1, \dots, \boldsymbol{X}_n | C) &= P(\boldsymbol{X}_1 | \boldsymbol{X}_2, \dots, \boldsymbol{X}_n, C) P(\boldsymbol{X}_2, \dots, \boldsymbol{X}_n | C) \\ &= P(\boldsymbol{X}_1 | \, C) P(\boldsymbol{X}_2, \dots, \boldsymbol{X}_n | C) \\ &= \cdots \\ &= P(\boldsymbol{X}_1 | \, C) \cdots P(\boldsymbol{X}_n | \, C) \end{aligned}$$

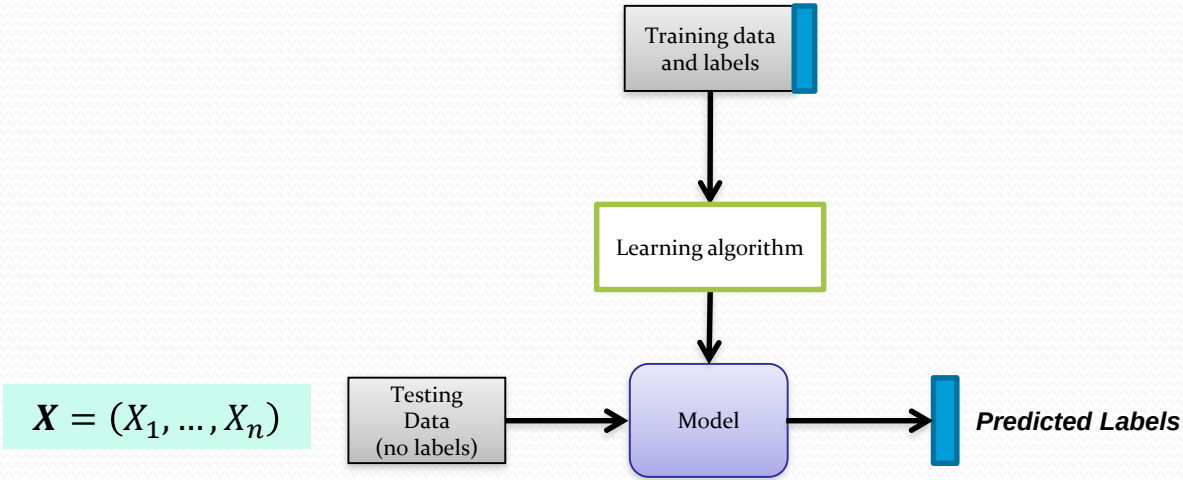
  - MAP classification rule

$$c^* = \underset{c \in \{c_1, \dots, c_L\}}{\operatorname{argmax}} \, P(\boldsymbol{X}_1 = \boldsymbol{x}_1 | \, C = c) \cdots P(\boldsymbol{X}_n = \boldsymbol{x}_n | \, C = c) P(C = c)$$

    - In case of tie-breaking (several maxima), choose randomly the class

The Supervised Learning Pipeline

$$S = \{(\boldsymbol{X}_1, y_1), \ldots, (\boldsymbol{X}_N, y_N)\}, y_i \in \{c_1, \ldots, c_L\}$$



$$\boldsymbol{X} = (X_1, \ldots, X_n)$$

4 Discrete Case

$\boldsymbol{X}_j \in \{a_{j1}, \ldots, a_{jK_j}\}$ with K_j possible values for all j

- Given a training set $S = \{(\boldsymbol{X}_1, y_1), \ldots, (\boldsymbol{X}_N, y_N)\}, y_i \in \{c_1, \ldots, c_L\}$

- **Learning phase**
  - For each c_i, calculate an estimate $\hat{P}(C = c_i)$ of $P(C = c_i)$ from S
  - For every feature value a_{jk} of each feature $\boldsymbol{X}_j$, calculate an estimate $\hat{P}(\boldsymbol{X_j} = a_{jk} | C = c_i)$ of $P(\boldsymbol{X_j} = a_{jk} | C = c_i)$ from S
  - Finally, we get $n \times L$ conditional probabilistic (generative) models and L estimates $\hat{P}(C = c_i)$

- **Test phase**
  - Given an unknown instance $\boldsymbol{x}' = (\boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_n)$
  - Assign the label c^* to $\boldsymbol{x}'$ such that

$$c^* = \underset{c \in \{c_1, \ldots, c_L\}}{\operatorname{argmax}} \hat{P}(\boldsymbol{X}_1 = \boldsymbol{x}'_1 | C = c) \cdots \hat{P}(\boldsymbol{X}_n = \boldsymbol{x}'_1 | C = c) \hat{P}(C = c)$$

Example: Play Tennis

PlayTennis: training examples

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

Example: Learning Phase

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

Temperature	Play=*Yes*	Play=*No*
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

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

Wind	Play=*Yes*	Play=*No*
Strong	3/9	3/5
Weak	6/9	2/5

$P(\text{Play}=Yes) = 9/14$ $P(\text{Play}=No) = 5/14$

Example: Test Phase

- Given a new instance, predict its label

  $\mathbf{x}'$=(Outlook=*Sunny*, Temperature=*Cool*, Humidity=*High*, Wind=*Strong*)

- Look up tables achieved in the learning phrase

  P(Outlook=*Sunny*|Play=*Yes*) = 2/9
  P(Temperature=*Cool*|Play=*Yes*) = 3/9
  P(Humidity=*High*|Play=*Yes*) = 3/9
  P(Wind=*Strong*|Play=*Yes*) = 3/9
  P(Play=*Yes*) = 9/14

  P(Outlook=S*unny*|Play=*No*) = 3/5
  P(Temperature=*Cool*|Play==*No*) = 1/5
  P(Humidity=*High*|Play=*No*) = 4/5
  P(Wind=*Strong*|Play=*No*) = 3/5
  P(Play=*No*) = 5/14

- Decision making with the MAP rule

  $P(Yes|\mathbf{x}') \approx [P(Sunny|Yes)P(Cool|Yes)P(High|Yes)P(Strong|Yes)]P(Play=Yes) = 0.0053$
  $P(No|\mathbf{x}') \approx [P(Sunny|No)\ P(Cool|No)P(High|No)P(Strong|No)]P(Play=No) = 0.0206$

  Given the fact $P(Yes|\mathbf{x}') < P(No|\mathbf{x}')$, we label $\mathbf{x}'$ to be "*No*".

Relevant Issues

- **Independence Assumption**
  - For many real world tasks, $P(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n|C) \neq P(\boldsymbol{X}_1|\, C) \cdots P(\boldsymbol{X}_n|\, C)$
  - Nevertheless, naïve Bayes works surprisingly well anyway!

- **Zero conditional probability Problem**
  - If no example contains the feature value $\boldsymbol{X}_j = a_{jk}$, $\hat{P}(\boldsymbol{X}_j = a_{jk}|C = c_i) = 0$
  - In this circumstance, during test, if $\boldsymbol{x}'_j = a_{jk}$, then

$$\hat{P}(\boldsymbol{X}_1 = \boldsymbol{x}'_1|\, C = c_i) \cdots \hat{P}(\boldsymbol{X}_j = a_{jk}|\, C = c_i) \cdots \hat{P}(\boldsymbol{X}_n = \boldsymbol{x}'_1|\, C = c_i) = 0$$

  - Lidstone's smoothing: for a remedy, conditional probabilities can be re-estimated with

$$\hat{P}(\boldsymbol{X}_j = a_{jk}|C = c_i) = \frac{N_{jki} + \lambda}{N_i + \lambda K_j}$$

    - N_{jki}: number of training examples for which $\boldsymbol{X}_j = a_{jk}$ and $C = c_i$
    - N_i: number of training examples such that $C = c_i$
    - K_j: number of possible values of $\boldsymbol{X}_j$
    - $\lambda > 0$: smoothing parameter
      - $\lambda = 0$: no-smoothing, $\lambda = 1$: add-one smoothing and $\lambda = \infty$: uniform probability $1/K_j$

5 Continuous Case

$X_j \in \mathbb{R}$ for all j

- Given a training set $S = \{(\boldsymbol{X}_1, y_1), \ldots, (\boldsymbol{X}_N, y_N)\}, y_i \in \{c_1, \ldots, c_L\}$
- **Learning phase**
  - For each c_i, calculate an estimate $\hat{P}(C = c_i)$ of $P(C = c_i)$ from S
  - Conditional probability of each feature $\boldsymbol{X}_j$ often modeled with the normal distribution

$$\hat{P}(\boldsymbol{X_j} = \boldsymbol{x}_j | C = c_i) = \frac{1}{\sqrt{2\pi\sigma_{ji}^2}} \exp\left(-\frac{(\boldsymbol{x}_j - \mu_{ji})^2}{2\sigma_{ji}^2}\right)$$

    - μ_{ji}: mean of feature $\boldsymbol{X_j}$ of examples for which $C = c_i$
    - σ_{ji}: standard deviation of feature $\boldsymbol{X_j}$ of examples for which $C = c_i$
  - For every feature feature $\boldsymbol{X}_j$, calculate some estimates $\hat{\mu}_{ji}$ of μ_{ji} and $\hat{\sigma}_{ji}^2$ of σ_{ji}^2 from S
  - Finally, we get $n \times L$ conditional probabilistic (generative) models and L estimates $\hat{P}(C = c_i)$
- **Test phase**
  - Given an unknown instance $\boldsymbol{x}' = (\boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_n)$
  - Calculate $\hat{P}(\boldsymbol{X_j} = \boldsymbol{x}'_j | C = c_i)$ for all j and all i
  - Assign the label c^* to $\boldsymbol{x}'$ such that $c^* = \underset{c \in \{c_1, \ldots, c_L\}}{\operatorname{argmax}} \hat{P}(\boldsymbol{X}_1 = \boldsymbol{x}'_1 | C = c) \cdots \hat{P}(\boldsymbol{X}_n = \boldsymbol{x}'_1 | C = c) \hat{P}(C = c)$

Example: Monitoring a System Temperature

- Two classes:
  - Yes: the system works well
  - No: the system has a failure
- Temperature is naturally a continuous value X:

  Yes: 25.2, 19.3, 18.5, 21.7, 20.1, 24.3, 22.8, 23.1, 19.8

  No: 27.3, 30.1, 17.4, 29.5, 15.1
- Estimate mean and variance for each class:

$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} x_i$, $\widehat{\sigma^2} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \hat{\mu})^2$ gives $\begin{cases} \hat{\mu}_{\text{Yes}} = 21.64, & \hat{\sigma}_{\text{Yes}} = 2.35 \\ \hat{\mu}_{\text{No}} = 23.88, & \hat{\sigma}_{\text{No}} = 7.09 \end{cases}$

- **Learning Phase**: output two Gaussian models for $P(\text{temperature}|C)$

$$\hat{P}(x \mid Yes) = \frac{1}{2.35\sqrt{2\pi}} \exp\left(-\frac{(x-21.64)^2}{2\times 2.35^2}\right) = \frac{1}{2.35\sqrt{2\pi}} \exp\left(-\frac{(x-21.64)^2}{11.09}\right)$$

$$\hat{P}(x \mid No) = \frac{1}{7.09\sqrt{2\pi}} \exp\left(-\frac{(x-23.88)^2}{2\times 7.09^2}\right) = \frac{1}{7.09\sqrt{2\pi}} \exp\left(-\frac{(x-23.88)^2}{50.25}\right)$$

6 Conclusion

Conclusion

- Naïve Bayes is based on the conditional independence assumption between attributes
  - Training is very easy and fast; just requiring considering each attribute in each class separately
  - Test is straightforward; just looking up tables or calculating conditional probabilities with estimated distributions

- The Bayes test is generally more difficult to build when taking care of the dependence between the features.

- A popular generative model
  - Performance competitive to most of state-of-the-art classifiers even in presence of violating independence assumption
  - Many successful applications, e.g., spam mail filtering
  - A target test for many learning based classifiers.