Probability Based Learning

Lecture 7, DD2431 Machine Learning

J. Sullivan, A. Maki

September 2013

Advantages of Probability Based Methods

- **Work with sparse training data.** More powerful than deterministic methods - decision trees - when training data is sparse.
- **Results are interpretable.** More transparent and mathematically rigorous than methods such as *ANN*, *Evolutionary methods*.
- **Tool for interpreting other methods.** Framework for formalizing other methods - *concept learning*, *least squares*.

Outline

- Probability Theory Basics
  - ✓ Bayes' rule
  - ✓ MAP and ML estimation
  - ✓ Minimum Description Length principle
- Naïve Bayes Classifier
- EM Algorithm

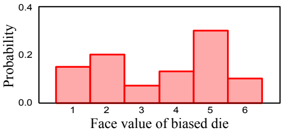
Probability Theory Basics

Random Variables

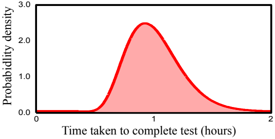
- A random variable x denotes a quantity that is uncertain
  - ✓ the result of flipping a coin,
  - ✓ the result of measuring the temperature
- The *probability distribution* $P(x)$ of a randam variable (r.v.) captures the fact that
  - ✓ the r.v. will have different values when observed **and**
  - ✓ Some values occur more than others.

Random Variables

- A **discrete random variable** takes values from a predefined set.
- For a **Boolean discrete random variable** this predefined set has two members - $\{0, 1\}$, $\{\text{yes}, \text{no}\}$ etc.
- A **continuous random variable** takes values that are real numbers.



discrete pdf

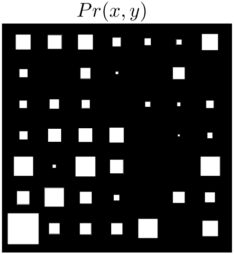


continuous pdf

Figures taken from **Computer Vision: models, learning and inference** by Simon Prince.

Joint Probabilities

- Consider two random variables x and y.
- Observe multiple paired instances of x and y. Some paired outcomes will occur more frequently.
- This information is encoded in the joint probability distribution $P(x, y)$.
- $P(\mathbf{x})$ denotes the joint probability of $\mathbf{x} = (x_1, \ldots, x_K)$.



← **discrete joint pdf**

Figure from **Computer Vision: models, learning and inference** by Simon Prince.

Joint Probabilities (cont.)

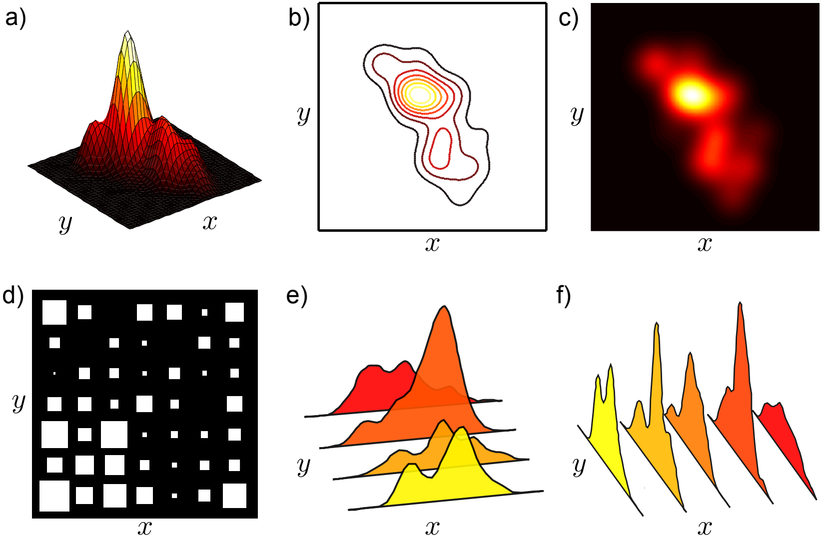


Figure from **Computer Vision: models, learning and inference** by Simon Prince.

Marginalization

The probability distribution of any single variable can be recovered from a joint distribution by summing for the discrete case

$$P(x) = \sum_y P(x, y)$$

and integrating for the continuous case

$$P(x) = \int_y P(x, y)\, dy$$

Marginalization (cont.)

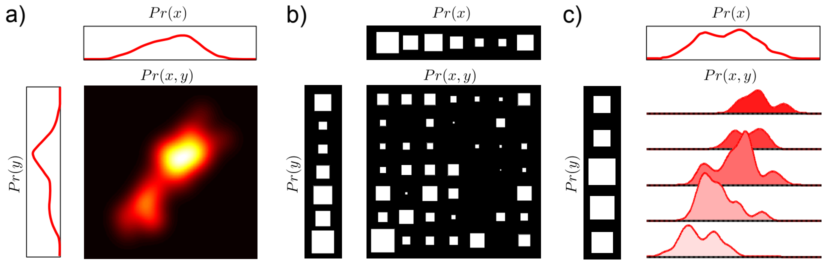


Figure from **Computer Vision: models, learning and inference** by Simon Prince.

Conditional Probability

- The conditional probability of x given that y takes value y^* indicates the different values of r.v. x which we'll observe given that y is fixed to value y^*.
- The conditional probability can be recovered from the joint distribution $P(x, y)$:

$$P(x \mid y = y^*) = \frac{P(x, y = y^*)}{P(y = y^*)} = \frac{P(x, y = y^*)}{\int_x P(x, y = y^*)\, dx}$$

- Extract an appropriate slice, and then normalize it.

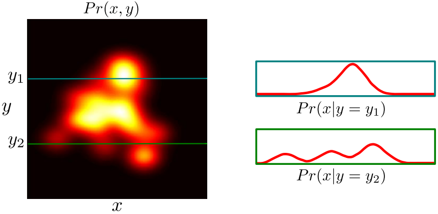


Figure from **Computer Vision: models, learning and inference** by Simon Prince.

Bayes' Rule

Bayes' Rule

$$P(y \mid x) = \frac{P(x \mid y)P(y)}{P(x)} = \frac{P(x \mid y)P(y)}{\sum_y P(x \mid y)P(y)}$$

Each term in Bayes' rule has a name:

- $P(y \mid x) \leftarrow$ *Posterior* (what we know about y given x.)
- $P(y) \leftarrow$ *Prior* (what we know about y before we consider x.)
- $P(x \mid y) \leftarrow$ *Likelihood* (propensity for observing a certain value of x given a certain value of y)
- $P(x) \leftarrow$ *Evidence* (a constant to ensure that the l.h.s. is a valid distribution)

Bayes' Rule

In many of our applications y is a discrete variable and $\mathbf{x}$ is a multi-dimensional data vector extracted from the world.

$$P(y \mid \mathbf{x}) = \frac{P(\mathbf{x} \mid y)P(y)}{P(\mathbf{x})}$$

Then

- $P(\mathbf{x} \mid y) \leftarrow$ *Likelihood* represents the probability of observing data $\mathbf{x}$ given the hypothesis y.
- $P(y) \leftarrow$ *Prior of y* represents the background knowledge of hypothesis y being correct.
- $P(y \mid \mathbf{x}) \leftarrow$ *Posterior* represents the probability that hypothesis y is true after data $\mathbf{x}$ has been observed.

Learning and Inference

- **Bayesian Inference:** The process of calculating the posterior probability distribution $P(y \,|\, \mathbf{x})$ for certain data $\mathbf{x}$.

- **Bayesian Learning:** The process of learning the likelihood distribution $P(\mathbf{x} \,|\, y)$ and prior probability distribution $P(y)$ from a set of training points

$$\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$$

Example: Which Gender?

Task: Determine the gender of a person given their measured hair length.

Notation:

- Let $g \in \{\text{'f'}, \text{'m'}\}$ be a r.v. denoting the gender of a person.
- Let x be the measured length of the hair.

Information given:

- The hair length observation was made at a boy's school thus

$$P(g = \text{'m'}) = .95, \quad P(g = \text{'f'}) = .05$$

- Knowledge of the likelihood distributions $P(x \mid g = \text{'f'})$ and $P(x \mid g = \text{'m'})$

Example: Which Gender?

Task: Determine the gender of a person given their measured hair length.

Notation:

- Let $g \in \{\text{'f'}, \text{'m'}\}$ be a r.v. denoting the gender of a person.
- Let x be the measured length of the hair.

Information given:

- The hair length observation was made at a boy's school thus

$$P(g = \text{'m'}) = .95, \quad P(g = \text{'f'}) = .05$$

- Knowledge of the likelihood distributions $P(x \mid g = \text{'f'})$ and $P(x \mid g = \text{'m'})$

Example: Which Gender?

Task: Determine the gender of a person given their measured hair length.

Notation:

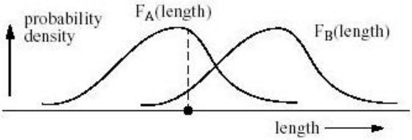
- Let $g \in \{\text{'f'}, \text{'m'}\}$ be a r.v. denoting the gender of a person.
- Let x be the measured length of the hair.

Information given:

- The hair length observation was made at a boy's school thus

$$P(g = \text{'m'}) = .95, \quad P(g = \text{'f'}) = .05$$

- Knowledge of the likelihood distributions $P(x \,|\, g = \text{'f'})$ and $P(x \,|\, g = \text{'m'})$



Example: Which Gender?

Task: Determine the gender of a person given their measured hair length $\implies$ calculate $P(g \,|\, x)$.

Solution:

Apply Bayes' Rule to get

$$\begin{aligned} P(g = \text{'m'} \,|\, x) &= \frac{P(x \,|\, g = \text{'m'})P(g = \text{'m'})}{P(x)} \\ &= \frac{P(x \,|\, g = \text{'m'})P(g = \text{'m'})}{P(x \,|\, g = \text{'f'})P(g = \text{'f'}) + P(x \,|\, g = \text{'m'})P(g = \text{'m'})} \end{aligned}$$

Can calculate $P(g = \text{'f'} \,|\, x) = 1 - P(g = \text{'m'} \,|\, x)$

Selecting the most probably hypothesis

- **Maximum A Posteriori (*MAP*) Estimate**:

  Hypothesis with highest probability given observed data

$$\begin{aligned} y_{\mathrm{MAP}} &= \arg\max_{y \in \mathcal{Y}} P(y \,|\, \mathbf{x}) \\ &= \arg\max_{y \in \mathcal{Y}} \frac{P(\mathbf{x} \,|\, y)\, P(y)}{P(\mathbf{x})} \\ &= \arg\max_{y \in \mathcal{Y}} P(\mathbf{x} \,|\, y)\, P(y) \end{aligned}$$

- **Maximum Likelihood Estimate (*MLE*):**

  Hypothesis with highest likelihood of generating observed data.

$$y_{\mathrm{MLE}} = \arg\max_{y \in \mathcal{Y}} P(\mathbf{x} \,|\, y)$$

  Useful if we do not know prior distribution or if it is uniform.

Selecting the most probably hypothesis

- **Maximum A Posteriori (*MAP*) Estimate**:

  Hypothesis with highest probability given observed data

$$
\begin{aligned}
y_{\mathrm{MAP}} &= \arg\max_{y \in \mathcal{Y}} P(y \,|\, \mathbf{x}) \\
&= \arg\max_{y \in \mathcal{Y}} \frac{P(\mathbf{x} \,|\, y)\, P(y)}{P(\mathbf{x})} \\
&= \arg\max_{y \in \mathcal{Y}} P(\mathbf{x} \,|\, y)\, P(y)
\end{aligned}
$$

- **Maximum Likelihood Estimate (*MLE*):**

  Hypothesis with highest likelihood of generating observed data.

$$
y_{\mathrm{MLE}} = \arg\max_{y \in \mathcal{Y}} P(\mathbf{x} \,|\, y)
$$

  Useful if we do not know prior distribution or if it is uniform.

Example: Cancer or Not?

Scenario:
A patient takes a lab test and the result comes back positive. The test returns a correct positive result in only 98% of the cases in which the disease is actually present, and a correct negative result in only 97% of the cases in which the disease is not present. Furthermore, 0.8% of the entire population have cancer.

Scenario in probabilities:

- **Priors:**

$$P(\text{disease}) = .008 \qquad P(\text{not disease}) = .992$$

- **Likelihoods:**

$$P(+\,|\,\text{disease}) = .98 \qquad P(+\,|\,\text{not disease}) = .03$$
$$P(-\,|\,\text{disease}) = .02 \qquad P(-\,|\,\text{not disease}) = .97$$

Example: Cancer or Not?

Find MAP estimate:
When test returned a positive result,

$$\begin{aligned} y_{\text{MAP}} &= \arg \max_{y \in \{\text{disease, not disease}\}} P(y \mid +) \\ &= \arg \max_{y \in \{\text{disease, not disease}\}} P(+ \mid y)\, P(y) \end{aligned}$$

Substituting in the correct values get

$$\begin{aligned} P(+ \mid \text{disease})\, P(\text{disease}) &= .98 \times .008 = .0078 \\ P(+ \mid \text{not disease})\, P(\text{not disease}) &= .03 \times .992 = .0298 \end{aligned}$$

Therefore y_{MAP} = "not disease".

The Posterior probabilities:

$$\begin{aligned} P(\text{disease} \mid +) &= \frac{.0078}{(.0078 + .0298)} = .21 \\ P(\text{not disease} \mid +) &= \frac{.0298}{(.0078 + .0298)} = .79 \end{aligned}$$

Example: Cancer or Not?

Find MAP estimate:
When test returned a positive result,

$$\begin{aligned} y_{\text{MAP}} &= \arg \max_{y \in \{\text{disease, not disease}\}} P(y \mid +) \\ &= \arg \max_{y \in \{\text{disease, not disease}\}} P(+ \mid y)\, P(y) \end{aligned}$$

Substituting in the correct values get

$$\begin{aligned} P(+ \mid \text{disease})\, P(\text{disease}) &= .98 \times .008 = .0078 \\ P(+ \mid \text{not disease})\, P(\text{not disease}) &= .03 \times .992 = .0298 \end{aligned}$$

Therefore $y_{\text{MAP}} =$ "not disease".

The Posterior probabilities:

$$\begin{aligned} P(\text{disease} \mid +) &= \frac{.0078}{(.0078 + .0298)} = .21 \\ P(\text{not disease} \mid +) &= \frac{.0298}{(.0078 + .0298)} = .79 \end{aligned}$$

Example: Cancer or Not?

Find MAP estimate:
When test returned a positive result,

$$\begin{aligned} y_{\text{MAP}} &= \arg \max_{y \in \{\text{disease, not disease}\}} P(y \,|\, +) \\ &= \arg \max_{y \in \{\text{disease, not disease}\}} P(+ \,|\, y)\, P(y) \end{aligned}$$

Substituting in the correct values get

$$\begin{aligned} P(+ \,|\, \text{disease})\, P(\text{disease}) &= .98 \times .008 = .0078 \\ P(+ \,|\, \text{not disease})\, P(\text{not disease}) &= .03 \times .992 = .0298 \end{aligned}$$

Therefore $y_{\text{MAP}} =$ "not disease".

The Posterior probabilities:

$$\begin{aligned} P(\text{disease} \,|\, +) &= \frac{.0078}{(.0078 + .0298)} = .21 \\ P(\text{not disease} \,|\, +) &= \frac{.0298}{(.0078 + .0298)} = .79 \end{aligned}$$

Relation to *Occams's Razor*

Occam's Razor:

Choose the simplest explanation for the observed data

- Information theoretic perspective Occam's razor corresponds to choosing the explanation requiring the fewest bits to represent.
- The optimal representation requires $-\log_2 p(y \,|\, \mathbf{x})$ bits to store. (Remember: the Shannon information content)
- Minimum description length principle: Choose hypothesis

$$y_{\mathrm{MDL}} = \arg\min_{y \in \mathcal{Y}} -\log_2 P(y \,|\, \mathbf{x})$$
$$= \arg\min_{y \in \mathcal{Y}} -\log_2 P(\mathbf{x} \,|\, y) - \log_2 P(y)$$

- The MDL estimate is equal to the MAP estimate

$$y_{\mathrm{MAP}} = \arg\max_{y \in \mathcal{Y}} \log_2 P(\mathbf{x} \,|\, y) + \log_2 P(y)$$

Relation to *Occams's Razor*

Occam's Razor:

Choose the simplest explanation for the observed data

- Information theoretic perspective Occam's razor corresponds to choosing the explanation requiring the fewest bits to represent.
- The optimal representation requires $-\log_2 p(y \,|\, \mathbf{x})$ bits to store. (Remember: the Shannon information content)
- Minimum description length principle: Choose hypothesis

$$\begin{aligned} y_{\text{MDL}} &= \arg\min_{y \in \mathcal{Y}} \; -\log_2 P(y \,|\, \mathbf{x}) \\ &= \arg\min_{y \in \mathcal{Y}} \; -\log_2 P(\mathbf{x} \,|\, y) - \log_2 P(y) \end{aligned}$$

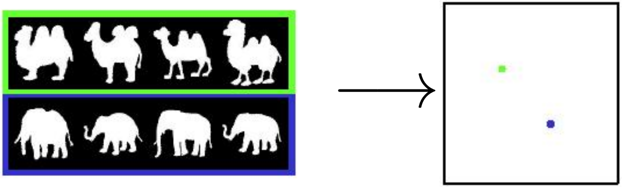
- The MDL estimate is equal to the MAP estimate

$$y_{\text{MAP}} = \arg\max_{y \in \mathcal{Y}} \; \log_2 P(\mathbf{x} \,|\, y) + \log_2 P(y)$$

Naïve Bayes Classifier

Feature Space

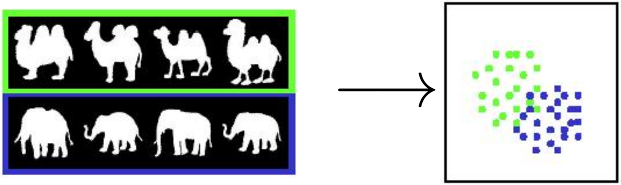
- Sensors give *measurements* which can be converted to *features*.
- Ideally a feature value is identical for all *samples* in one *class*.



Samples **Feature space**

Feature Space

- Sensors give *measurements* which can be converted to *features*.
- However in the real world



Samples **Feature space**

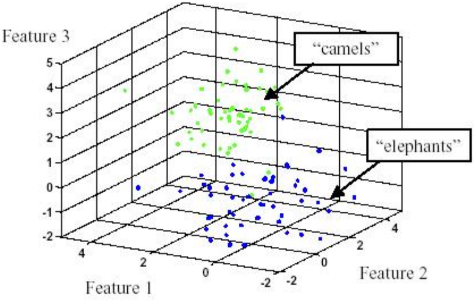
because of

- ✓ Measurement noise
- ✓ Intra-class variation
- ✓ Poor choice of features

Feature Space

End result: a $K-$dimensional space

- in which each dimension is a **feature**
- containing n labelled **samples** (objects)



Problem: Large Feature Space

- Size of feature space exponential in number of features.
- More features $\implies$ potential for better description of the objects but...

  More features $\implies$ more difficult to model $P(\mathbf{x} \mid y)$.
- **Extreme Solution:** Naïve Bayes classifier
  - ✓ All features (dimensions) regarded as independent.
  - ✓ Model k one-dimensional distributions instead of one k-dimensional distribution.

Problem: Large Feature Space

- Size of feature space exponential in number of features.
- More features $\Longrightarrow$ potential for better description of the objects but...

  More features $\Longrightarrow$ more difficult to model $P(\mathbf{x} \mid y)$.
- **Extreme Solution:** Naïve Bayes classifier
  - ✓ All features (dimensions) regarded as independent.
  - ✓ Model k one-dimensional distributions instead of one k-dimensional distribution.

Naïve Bayes Classifier

- One of the most common learning methods.
- **When to use:**
  - ✓ Moderate or large training set available.
  - ✓ Features x_i of a data instance $\mathbf{x}$ are conditionally independent given classification (or at least reasonably independent, still works with a little dependence).
- **Successful applications:**
  - ✓ Medical diagnoses (symptoms independent)
  - ✓ Classification of text documents (words independent)

Naïve Bayes Classifier

- $\mathbf{x}$ is a vector $(x_1, \ldots, x_K)$ of attribute or feature values.
- Let $\mathcal{Y} = \{1, 2, \ldots, Y\}$ be the set of possible classes.
- The MAP estimate of y is

$$\begin{aligned} y_{\text{MAP}} &= \arg\max_{y \in \mathcal{Y}} P(y \,|\, x_1, \ldots, x_K) \\ &= \arg\max_{y \in \mathcal{Y}} \frac{P(x_1, \ldots, x_K \,|\, y)\, P(y)}{P(x_1, \ldots, x_K)} \\ &= \arg\max_{y \in \mathcal{Y}} P(x_1, \ldots, x_K \,|\, y)\, P(y) \end{aligned}$$

- Naïve Bayes assumption: $P(x_1, \ldots, x_K \,|\, y) = \prod_{k=1}^{K} P(x_k \,|\, y)$
- This give the *Naïve Bayes classifier*:

$$y_{\text{MAP}} = \arg\max_{y \in \mathcal{Y}} P(y) \prod_{k=1}^{K} P(x_k \,|\, y)$$

Naïve Bayes Classifier

- $\mathbf{x}$ is a vector $(x_1, \ldots, x_K)$ of attribute or feature values.
- Let $\mathcal{Y} = \{1, 2, \ldots, Y\}$ be the set of possible classes.
- The MAP estimate of y is

$$
\begin{aligned}
y_{\text{MAP}} &= \arg\max_{y \in \mathcal{Y}} P(y \,|\, x_1, \ldots, x_K) \\
&= \arg\max_{y \in \mathcal{Y}} \frac{P(x_1, \ldots, x_K \,|\, y)\, P(y)}{P(x_1, \ldots, x_K)} \\
&= \arg\max_{y \in \mathcal{Y}} P(x_1, \ldots, x_K \,|\, y)\, P(y)
\end{aligned}
$$

- Naïve Bayes assumption: $P(x_1, \ldots, x_K \,|\, y) = \prod_{k=1}^{K} P(x_k \,|\, y)$
- This give the *Naïve Bayes classifier*:

$$
y_{\text{MAP}} = \arg\max_{y \in \mathcal{Y}} P(y) \prod_{k=1}^{K} P(x_k \,|\, y)
$$

Example: Play Tennis?

Question: Will I go and play tennis given the forecast?

My measurements:

1. **forecast** $\in$ {sunny, overcast, rainy},
2. **temperature** $\in$ {hot, mild, cool},
3. **humidity** $\in$ {high, normal},
4. **windy** $\in$ {false, true}.

Possible decisions:

$y \in \{\text{yes}, \text{no}\}$

Example: Play Tennis?

What I did in the past:

outlook	temp.	humidity	windy	play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes

outlook	temp.	humidity	windy	play
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

Example: Play Tennis?

Counts of when I played tennis (did not play)

Outlook			**Temperature**			**Humidity**		**Windy**	
sunny	overcast	rain	hot	mild	cool	high	normal	false	true
2 (3)	4 (0)	3 (2)	2 (2)	4 (2)	3 (1)	3 (4)	6 (1)	6 (2)	3 (3)

Prior of whether I played tennis or not

Counts:

Play	
yes	no
9	5

Prior Probabilities:

Play	
yes	no
$\frac{9}{14}$	$\frac{5}{14}$

Likelihood of attribute when tennis played $P(x_i \mid \text{y=yes})(P(x_i \mid \text{y=no}))$

Outlook			**Temperature**			**Humidity**		**Windy**	
sunny	overcast	rain	hot	mild	cool	high	normal	false	true
$\frac{2}{9}$ $(\frac{3}{5})$	$\frac{4}{9}$ $(\frac{0}{5})$	$\frac{3}{9}$ $(\frac{2}{5})$	$\frac{2}{9}$ $(\frac{2}{5})$	$\frac{4}{9}$ $(\frac{2}{5})$	$\frac{3}{9}$ $(\frac{1}{5})$	$\frac{3}{9}$ $(\frac{4}{5})$	$\frac{6}{9}$ $(\frac{1}{5})$	$\frac{6}{9}$ $(\frac{2}{5})$	$\frac{3}{9}$ $(\frac{3}{5})$

Example: Play Tennis?

Counts of when I played tennis (did not play)

Outlook			Temperature			Humidity		Windy	
sunny	overcast	rain	hot	mild	cool	high	normal	false	true
2 (3)	4 (0)	3 (2)	2 (2)	4 (2)	3 (1)	3 (4)	6 (1)	6 (2)	3 (3)

Prior of whether I played tennis or not

Counts:

Play	
yes	no
9	5

Prior Probabilities:

Play	
yes	no
$\frac{9}{14}$	$\frac{5}{14}$

Likelihood of attribute when tennis played $P(x_i \mid \text{y=yes})(P(x_i \mid \text{y=no}))$

Outlook			Temperature			Humidity		Windy	
sunny	overcast	rain	hot	mild	cool	high	normal	false	true
$\frac{2}{9}$ $(\frac{3}{5})$	$\frac{4}{9}$ $(\frac{0}{5})$	$\frac{3}{9}$ $(\frac{2}{5})$	$\frac{2}{9}$ $(\frac{2}{5})$	$\frac{4}{9}$ $(\frac{2}{5})$	$\frac{3}{9}$ $(\frac{1}{5})$	$\frac{3}{9}$ $(\frac{4}{5})$	$\frac{6}{9}$ $(\frac{1}{5})$	$\frac{6}{9}$ $(\frac{2}{5})$	$\frac{3}{9}$ $(\frac{3}{5})$

Example: Play Tennis?

Counts of when I played tennis (did not play)

Outlook			**Temperature**			**Humidity**		**Windy**	
sunny	overcast	rain	hot	mild	cool	high	normal	false	true
2 (3)	4 (0)	3 (2)	2 (2)	4 (2)	3 (1)	3 (4)	6 (1)	6 (2)	3 (3)

Prior of whether I played tennis or not

Counts:

Play	
yes	no
9	5

Prior Probabilities:

Play	
yes	no
$\frac{9}{14}$	$\frac{5}{14}$

Likelihood of attribute when tennis played $P(x_i \mid \text{y=yes})(P(x_i \mid \text{y=no}))$

Outlook			**Temperature**			**Humidity**		**Windy**	
sunny	overcast	rain	hot	mild	cool	high	normal	false	true
$\frac{2}{9}$ $(\frac{3}{5})$	$\frac{4}{9}$ $(\frac{0}{5})$	$\frac{3}{9}$ $(\frac{2}{5})$	$\frac{2}{9}$ $(\frac{2}{5})$	$\frac{4}{9}$ $(\frac{2}{5})$	$\frac{3}{9}$ $(\frac{1}{5})$	$\frac{3}{9}$ $(\frac{4}{5})$	$\frac{6}{9}$ $(\frac{1}{5})$	$\frac{6}{9}$ $(\frac{2}{5})$	$\frac{3}{9}$ $(\frac{3}{5})$

Example: Play Tennis?

Inference: Use the learnt model to classify a new instance.

New instance:

$$\mathbf{x} = (\text{sunny, cool, high, true})$$

Apply Naïve Bayes Classifier:

$$y_{\text{MAP}} = \arg \max_{y \,\in\, \{\text{yes, no}\}} P(y) \prod_{i=1}^{4} P(x_i \,|\, y)$$

$$P(\text{yes})\, P(\text{sunny} \,|\, \text{yes})\, P(\text{cool} \,|\, \text{yes})\, P(\text{high} \,|\, \text{yes})\, P(\text{true} \,|\, \text{yes}) = \frac{9}{14} \times \frac{2}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} = .005$$

$$P(\text{no})\, P(\text{sunny} \,|\, \text{no})\, P(\text{cool} \,|\, \text{no})\, P(\text{high} \,|\, \text{no})\, P(\text{true} \,|\, \text{no}) = \frac{5}{14} \times \frac{3}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{5} = .021$$

$$\Longrightarrow \; y_{\text{MAP}} = \text{no}$$

Naïve Bayes: Independence Violation

- Conditional independence assumption:

$$P(x_1, x_2, \ldots, x_K \mid y) = \prod_{k=1}^{K} P(x_k \mid y)$$

often violated - but it works surprisingly well anyway!

- **Note:** Do not need the posterior probabilities $P(y \mid \mathbf{x})$ to be correct. Only need y_{MAP} to be correct.

- Since dependencies ignored, naïve Bayes posteriors often unrealistically close to 0 or 1.
  Different attributes say the same thing to a higher degree than we expect as they are correlated in reality.

Naïve Bayes: Independence Violation

- Conditional independence assumption:

$$P(x_1, x_2, \ldots, x_K \,|\, y) = \prod_{k=1}^{K} P(x_k \,|\, y)$$

  often violated - but it works surprisingly well anyway!

- **Note:** Do not need the posterior probabilities $P(y \,|\, \mathbf{x})$ to be correct. Only need y_{MAP} to be correct.

- Since dependencies ignored, naïve Bayes posteriors often unrealistically close to 0 or 1.
  Different attributes say the same thing to a higher degree than we expect as they are correlated in reality.

Naïve Bayes: Independence Violation

- Conditional independence assumption:

$$P(x_1, x_2, \ldots, x_K \mid y) = \prod_{k=1}^{K} P(x_k \mid y)$$

  often violated - but it works surprisingly well anyway!

- **Note:** Do not need the posterior probabilities $P(y \mid \mathbf{x})$ to be correct. Only need y_{MAP} to be correct.

- Since dependencies ignored, naïve Bayes posteriors often unrealistically close to 0 or 1.
  Different attributes say the same thing to a higher degree than we expect as they are correlated in reality.

Naïve Bayes: Estimating Probabilities

- **Problem:** What if none of the training instances with target value y have attribute x_i? Then

$$P(x_i \mid y) = 0 \quad \Longrightarrow \quad P(y)\prod_{i=1}^{K} P(x_i \mid y) = 0$$

- **Solution:** Add as prior knowledge that $P(x_i \mid y)$ must be larger than 0:

$$P(x_i \mid y) = \frac{n_y + mp}{n + m}$$

where

$$\begin{aligned} n &= \text{number of training samples with label } y \\ n_y &= \text{number of training samples with label } y \text{ and value } x_i \\ p &= \text{prior estimate of } P(x_i \mid y) \\ m &= \text{weight given to prior estimate (in relation to data)} \end{aligned}$$

Naïve Bayes: Estimating Probabilities

- **Problem:** What if none of the training instances with target value y have attribute x_i? Then

$$P(x_i \mid y) = 0 \quad \Longrightarrow \quad P(y) \prod_{i=1}^{K} P(x_i \mid y) = 0$$

- **Solution:** Add as prior knowledge that $P(x_i \mid y)$ must be larger than 0:

$$P(x_i \mid y) = \frac{n_y + mp}{n + m}$$

where

$$\begin{aligned} n &= \text{number of training samples with label } y \\ n_y &= \text{number of training samples with label } y \text{ and value } x_i \\ p &= \text{prior estimate of } P(x_i \mid y) \\ m &= \text{weight given to prior estimate (in relation to data)} \end{aligned}$$

Example: Spam detection

- **Aim**: Build a classifier to identify spam e-mails.
- **How**:
  Training
  - ✓ Create dictionary of words and tokens $\mathcal{W} = \{w_1, \ldots, w_L\}$.
    These words should be those which are specific to spam or non-spam e-mails.
  - ✓ E-mail is a concatenation, in order, of its words and tokens: $\mathbf{e} = (e_1, e_2, \ldots, e_K)$ with $e_i \in \mathcal{W}$.
  - ✓ Must model and learn
    $P(e_1, e_2, \ldots, e_K \mid \text{spam})$ **and** $P(e_1, e_2, \ldots, e_K \mid \text{not spam})$

Example: Spam detection

- **Aim**: Build a classifier to identify spam e-mails.
- **How**:
  Training
  - ✓ Create dictionary of words and tokens $\mathcal{W} = \{w_1, \ldots, w_L\}$.
    These words should be those which are specific to spam or non-spam e-mails.
  - ✓ E-mail is a concatenation, in order, of its words and tokens: $\mathbf{e} = (e_1, e_2, \ldots, e_K)$ with $e_i \in \mathcal{W}$.
  - ✓ Must model and learn
    $P(e_1, e_2, \ldots, e_K \mid \text{spam})$ **and** $P(e_1, e_2, \ldots, e_K \mid \text{not spam})$

Example: Spam detection

- **Aim**: Build a classifier to identify spam e-mails.

- **How**:
  Training
  - ✓ Create dictionary of words and tokens $\mathcal{W} = \{w_1, \ldots, w_L\}$.

    These words should be those which are specific to spam or non-spam e-mails.
  - ✓ E-mail is a concatenation, in order, of its words and tokens: $\mathbf{e} = (e_1, e_2, \ldots, e_K)$ with $e_i \in \mathcal{W}$.
  - ✓ Must model and learn
    $P(e_1, e_2, \ldots, e_K \mid \text{spam})$ **and** $P(e_1, e_2, \ldots, e_K \mid \text{not spam})$

Email: E

Dear customer,
A fully licensed Online Pharmacy is offering pharmaceuticals:
- brought to you directly from abroad
-produced by the same multinational corporations selling through the major US pharmacies
-priced up to 5 times cheaper as compared to major US pharmacies.
Enjoy the US dollar purchasing power on http://pharmacy-buyonline.com.ua/

Vector: e

('dear', 'customer', ',', 'a', 'fully', 'licensed', ... ,'/')

Concatenate words from e-mail into a vector

Example: Spam detection

- **Aim**: Build a classifier to identify spam e-mails.

- **How**:
  Training
  - ✓ Create dictionary of words and tokens $\mathcal{W} = \{w_1, \ldots, w_L\}$.

    These words should be those which are specific to spam or non-spam e-mails.
  - ✓ E-mail is a concatenation, in order, of its words and tokens: $\mathbf{e} = (e_1, e_2, \ldots, e_K)$ with $e_i \in \mathcal{W}$.
  - ✓ Must model and learn
    $P(e_1, e_2, \ldots, e_K \,|\, \text{spam})$ **and** $P(e_1, e_2, \ldots, e_K \,|\, \text{not spam})$

  Inference
  - ✓ Given an e-mail, E, compute $\mathbf{e} = (e_1, \ldots, e_K)$.
  - ✓ Use Bayes' rule to compute

$$P(\text{spam} \,|\, e_1, \ldots, e_K) \propto P(e_1, \ldots, e_K \,|\, \text{spam})\, P(\text{spam})$$

Example: Spam detection

- How is the joint probability distribution modelled?

$$P(e_1, \dots, e_K \mid \text{spam})$$

Remember K will be very large and vary from e-mail to e-mail..

- Make conditional independence assumption:

$$P(e_1, \dots, e_K \mid \text{spam}) = \prod_{k=1}^{K} P(e_k \mid \text{spam})$$

Similarly

$$P(e_1, \dots, e_K \mid \text{not spam}) = \prod_{k=1}^{K} P(e_k \mid \text{not spam})$$

- Have assumed the position of word is not important.

Example: Spam detection

- How is the joint probability distribution modelled?

$$P(e_1, \ldots, e_K \mid \text{spam})$$

Remember K will be very large and vary from e-mail to e-mail..

- Make conditional independence assumption:

$$P(e_1, \ldots, e_K \mid \text{spam}) = \prod_{k=1}^{K} P(e_k \mid \text{spam})$$

Similarly

$$P(e_1, \ldots, e_K \mid \text{not spam}) = \prod_{k=1}^{K} P(e_k \mid \text{not spam})$$

- Have assumed the position of word is not important.

Example: Spam detection

- How is the joint probability distribution modelled?

$$P(e_1, \ldots, e_K \mid \text{spam})$$

Remember K will be very large and vary from e-mail to e-mail..

- Make conditional independence assumption:

$$P(e_1, \ldots, e_K \mid \text{spam}) = \prod_{k=1}^{K} P(e_k \mid \text{spam})$$

Similarly

$$P(e_1, \ldots, e_K \mid \text{not spam}) = \prod_{k=1}^{K} P(e_k \mid \text{not spam})$$

- Have assumed the position of word is not important.

Example: Spam detection

Learning:

Assume one has n training e-mails and their labels - spam /non-spam

$$\mathcal{S} = \{(\mathbf{e}_1, y_1), \dots, (\mathbf{e}_n, y_n)\}$$

Note: $\mathbf{e}_i = (e_{i1}, \dots, e_{iK_i})$.

Example: Spam detection

Learning:
Assume one has n training e-mails and their labels - spam /non-spam

$$\mathcal{S} = \{(\mathbf{e}_1, y_1), \ldots, (\mathbf{e}_n, y_n)\}$$

Note: $\mathbf{e}_i = (e_{i1}, \ldots, e_{iK_i})$.

Create dictionary

1. Make a union of all the distinctive words and tokens in $\mathbf{e}_1, \ldots, \mathbf{e}_n$ to create $\mathcal{W} = \{w_1, \ldots, w_L\}$. (**Note:** words such as *and*, *the*, ... omitted)

Example: Spam detection

Learning:

Assume one has n training e-mails and their labels - spam /non-spam

$$\mathcal{S} = \{(\mathbf{e}_1, y_1), \ldots, (\mathbf{e}_n, y_n)\}$$

Note: $\mathbf{e}_i = (e_{i1}, \ldots, e_{iK_i})$.

Learn probabilities

For $y \in \{\text{spam}, \text{not spam}\}$

1. Set $P(y) = \frac{\sum_{i=1}^{n} \text{Ind}(y_i = y)}{n}$ ←proportion of e-mails from class y.
2. $n_y = \sum_{i=1}^{n} K_i \times \text{Ind}\,(y_i = y)$ ← total # of words in the class y e-mails.
3. For each word w_l compute
   $n_{yl} = \sum_{i=1}^{n} \text{Ind}\,(y_i = y) \times \left(\sum_{k=1}^{K_i} \text{Ind}\,(e_{ik} = w_l)\right)$ ← # of occurrences of word w_l in the class y e-mails.
4. $P(w_l \,|\, y) = \frac{n_{yl}+1}{n_y+|\mathcal{W}|}$ ← assume prior value of $P(w_l \,|\, y)$ is $1/|\mathcal{W}|$.

Example: Spam detection

Inference: Classify a new e-mail $\mathbf{e}^* = (e_1^*, \ldots, e_{K^*}^*)$

$$y^* = \arg \max_{y \in \{-1,1\}} P(y) \prod_{k=1}^{K^*} P(e_k^* | y)$$

Summary so far

- **Bayesian theory**: Combines prior knowledge and observed data to find the most probable hypothesis.
- **Naïve Bayes Classifier**: All variables considered independent.

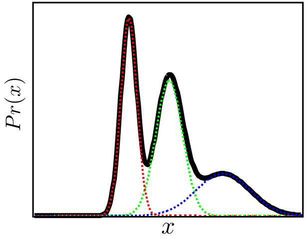
Expectation-Maximization (EM) Algorithm

Mixture of Gaussians

This distribution is a weight sum of K Gaussian distributions

$$P(x) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(x; \mu_k, \sigma_k^2)$$

where $\pi_1 + \cdots + \pi_K = 1$
and $\pi_k > 0$ $(k = 1, \ldots, K)$.



This model can describe **complex multi-modal** probability distributions by combining simpler distributions.

Mixture of Gaussians

$$P(x) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(x; \mu_k, \sigma_k^2)$$

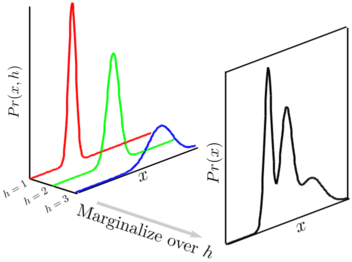
- Learning the parameters of this model from training data $x_1, \ldots, x_n$ is not trivial - using the usual straightforward maximum likelihood approach.
- Instead learn parameters using the **Expectation-Maximization** (EM) algorithm.

Mixture of Gaussians as a marginalization

We can interpret the Mixture of Gaussians model with the introduction of a discrete hidden/latent variable h and $P(x, h)$:

$$P(x) = \sum_{k=1}^{K} P(x, h = k) = \sum_{k=1}^{K} P(x \,|\, h = k) P(h = k)$$

$$= \sum_{k=1}^{K} \pi_k \, \mathcal{N}(x; \mu_k, \sigma_k^2)$$



← **mixture density**

Figures taken from **Computer Vision: models, learning and inference** by Simon Prince.

EM for two Gaussians

Assume: We know the pdf of x has this form:

$$P(x) = \pi_1 \mathcal{N}(x; \mu_1, \sigma_1^2) + \pi_2 \mathcal{N}(x; \mu_2, \sigma_2^2)$$

where $\pi_1 + \pi_2 = 1$ and $\pi_k > 0$ for components $k = 1, 2$.

Unknown: Values of the parameters (Many!)

$$\Theta = (\pi_1, \mu_1, \sigma_1, \mu_2, \sigma_2).$$

Have: Observed n samples $x_1, \ldots, x_n$ drawn from $p(x)$.

Want to: Estimate Θ from $x_1, \ldots, x_n$.

How would it be possible to get them all???

EM for two Gaussians

For each sample x_i introduce a *hidden variable* h_i

$$h_i = \begin{cases} 1 & \text{if sample } x_i \text{ was drawn from } \mathcal{N}(x; \mu_1, \sigma_1^2) \\ 2 & \text{if sample } x_i \text{ was drawn from } \mathcal{N}(x; \mu_2, \sigma_2^2) \end{cases}$$

and come up with initial values

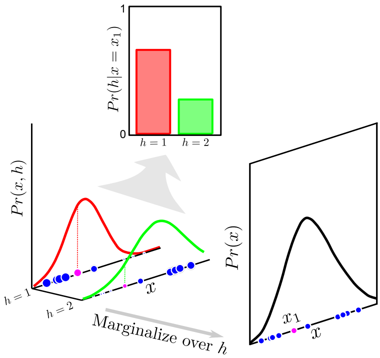
$$\Theta^{(0)} = (\pi_1^{(0)}, \mu_1^{(0)}, \sigma_1^{(0)}, \mu_2^{(0)}, \sigma_2^{(0)})$$

for each of the parameters.

EM is an *iterative algorithm* which updates $\Theta^{(t)}$ using the following two steps...

EM for two Gaussians: E-step

The responsibility of k-th Gaussian for each sample x (indicated by the size of the projected data point)



Look at each sample x along hidden variable h in the E-step

Figure from **Computer Vision: models, learning and inference** by Simon Prince.

EM for two Gaussians: E-step (cont.)

E-step: Compute the *"posterior probability"* that x_i was generated by component k given the current estimate of the parameters $\Theta^{(t)}$. (responsibilities)

for $i = 1, \ldots n$

for $k = 1, 2$

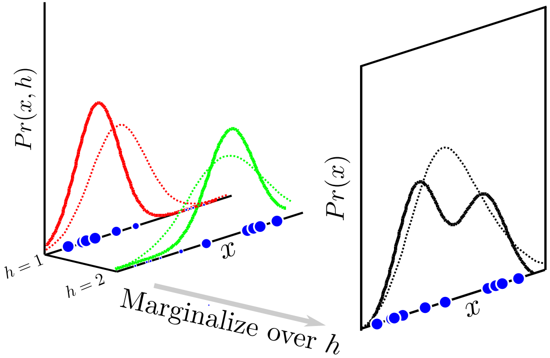
$$\gamma_{ik}^{(t)} = P(h_i = k \mid x_i, \Theta^{(t)})$$

$$= \frac{\pi_k^{(t)} \mathcal{N}(x_i; \mu_k^{(t)}, \sigma_k^{(t)})}{\pi_1^{(t)} \mathcal{N}(x_i; \mu_1^{(t)}, \sigma_1^{(t)}) + \pi_2^{(t)} \mathcal{N}(x_i; \mu_2^{(t)}, \sigma_2^{(t)})}$$

Note: $\gamma_{i1}^{(t)} + \gamma_{i2}^{(t)} = 1$ and $\pi_1 + \pi_2 = 1$

EM for two Gaussians: M-step

Fitting the Gaussian model for each of k-th constinuetnt.
Sample x_i contributes according to the responsibility γ_{ik}.



(dashed and solid lines for fit before and after update)

Look along samples x for each h in the M-step

Figure from **Computer Vision: models, learning and inference** by Simon Prince.

EM for two Gaussians: M-step (cont.)

M-step: Compute the *Maximum Likelihood* of the parameters of the mixture model given out data's membership distribution, the $\gamma_i^{(t)}$'s:

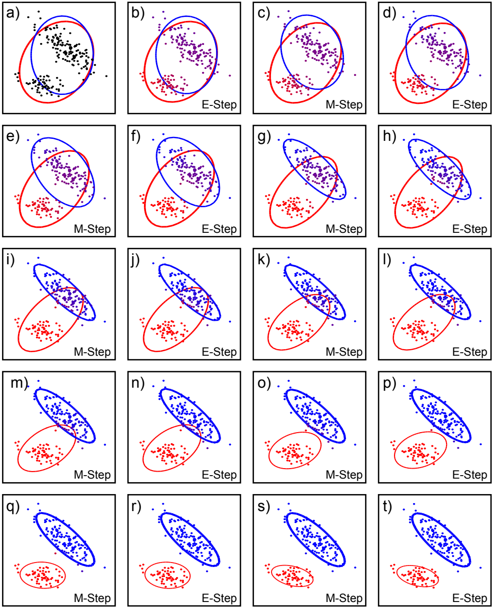
for $k = 1, 2$

$$\mu_k^{(t+1)} = \frac{\sum_{i=1}^{n} \gamma_{ik}^{(t)} x_i}{\sum_{i=1}^{n} \gamma_{ik}^{(t)}},$$

$$\sigma_k^{(t+1)} = \sqrt{\frac{\sum_{i=1}^{n} \gamma_{ik}^{(t)} (x_i - \mu_k^{(t+1)})^2}{\sum_{i=1}^{n} \gamma_{ik}^{(t)}}},$$

$$\pi_k^{(t+1)} = \frac{\sum_{i=1}^{n} \gamma_{ik}^{(t)}}{n}.$$

EM in practice



Summary

- **Bayesian theory**: Combines prior knowledge and observed data to find the most probable hypothesis.
- **Naïve Bayes Classifier**: All variables considered independent.
- **EM algorithm**: Learn probability destribiution (model parameters) in presence of hidden variables.

If you are interested in learning more take a look at:

C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer Verlag 2006.