Bayesian Learning II

Data Science Specialization
Spring 2025

Jens Classen

Roskilde University

26.03.2025

Smoothing

$P(fr)$	$=$	0.3	$P(\neg fr)$	$=$	0.7
$P(\mathrm{CH} = \textit{'none'} \mid fr)$	$=$	0.1666	$P(\mathrm{CH} = \textit{'none'} \mid \neg fr)$	$=$	0
$P(\mathrm{CH} = \textit{'paid'} \mid fr)$	$=$	0.1666	$P(\mathrm{CH} = \textit{'paid'} \mid \neg fr)$	$=$	0.2857
$P(\mathrm{CH} = \textit{'current'} \mid fr)$	$=$	0.5	$P(\mathrm{CH} = \textit{'current'} \mid \neg fr)$	$=$	0.2857
$P(\mathrm{CH} = \textit{'arrears'} \mid fr)$	$=$	0.1666	$P(\mathrm{CH} = \textit{'arrears'} \mid \neg fr)$	$=$	0.4286
$P(\mathrm{GC} = \textit{'none'} \mid fr)$	$=$	0.8334	$P(\mathrm{GC} = \textit{'none'} \mid \neg fr)$	$=$	0.8571
$P(\mathrm{GC} = \textit{'guarantor'} \mid fr)$	$=$	0.1666	$P(\mathrm{GC} = \textit{'guarantor'} \mid \neg fr)$	$=$	0
$P(\mathrm{GC} = \textit{'coapplicant'} \mid fr)$	$=$	0	$P(\mathrm{GC} = \textit{'coapplicant'} \mid \neg fr)$	$=$	0.1429
$P(\mathrm{ACC} = \textit{'own'} \mid fr)$	$=$	0.6666	$P(\mathrm{ACC} = \textit{'own'} \mid \neg fr)$	$=$	0.7857
$P(\mathrm{ACC} = \textit{'rent'} \mid fr)$	$=$	0.3333	$P(\mathrm{ACC} = \textit{'rent'} \mid \neg fr)$	$=$	0.1429
$P(\mathrm{ACC} = \textit{'free'} \mid fr)$	$=$	0	$P(\mathrm{ACC} = \textit{'free'} \mid \neg fr)$	$=$	0.0714

Credit History	Guarantor/CoApplicant	Accommodation	Fraudulent
paid	guarantor	free	?

$$\begin{array}{rclrcl}
P(fr) & = & 0.3 & P(\neg fr) & = & 0.7 \\
P(CH = paid \mid fr) & = & 0.1666 & P(CH = paid \mid \neg fr) & = & 0.2857 \\
P(GC = guarantor \mid fr) & = & 0.1666 & P(GC = guarantor \mid \neg fr) & = & 0 \\
P(ACC = free \mid fr) & = & 0 & P(ACC = free \mid \neg fr) & = & 0.0714
\end{array}$$

$$\left(\prod_{k=1}^{m} P(\mathbf{q}[k] \mid fr)\right) \times P(fr) = 0.0$$

$$\left(\prod_{k=1}^{m} P(\mathbf{q}[k] \mid \neg fr)\right) \times P(\neg fr) = 0.0$$

Credit History	Guarantor/CoApplicant	Accommodation	Fraudulent
paid	guarantor	free	?

- The standard way to avoid this issue is to use **smoothing**.
- Smoothing takes some of the probability from the events with lots of the probability share and gives it to the other probabilities in the set.

- There are several different ways to smooth probabilities, we will use **Laplacian smoothing**.

Laplacian Smoothing (conditional probabilities)

$$P(f = v|t) \quad = \quad \frac{count(f = v|t) + k}{count(f|t) + (k \times |Domain(f)|)}$$

Raw	$P(GC = none \mid \neg fr)$	$=$	0.8571
Probabilities	$P(GC = guarantor \mid \neg fr)$	$=$	0
	$P(GC = coapplicant \mid \neg fr)$	$=$	0.1429
Smoothing	k	$=$	3
Parameters	$count(GC \mid \neg fr)$	$=$	14
	$count(GC = none \mid \neg fr)$	$=$	12
	$count(GC = guarantor \mid \neg fr)$	$=$	0
	$count(GC = coapplicant \mid \neg fr)$	$=$	2
	$\lvert Domain(GC) \rvert$	$=$	3
Smoothed	$P(GC = none \mid \neg fr) = \frac{12+3}{14+(3\times 3)}$	$=$	0.6522
Probabilities	$P(GC = guarantor \mid \neg fr) = \frac{0+3}{14+(3\times 3)}$	$=$	0.1304
	$P(GC = coapplicant \mid \neg fr) = \frac{2+3}{14+(3\times 3)}$	$=$	0.2174

Table 1: Smoothing the posterior probabilities for the GUARANTOR/COAPPLICANT feature conditioned on FRAUDULENT being False.

$P(fr)$	$=$	0.3	$P(\neg fr)$	$=$	0.7
$P(CH = none \mid fr)$	$=$	0.2222	$P(CH = none \mid \neg fr)$	$=$	0.1154
$P(CH = paid \mid fr)$	$=$	0.2222	$P(CH = paid \mid \neg fr)$	$=$	0.2692
$P(CH = current \mid fr)$	$=$	0.3333	$P(CH = current \mid \neg fr)$	$=$	0.2692
$P(CH = arrears \mid fr)$	$=$	0.2222	$P(CH = arrears \mid \neg fr)$	$=$	0.3462
$P(GC = none \mid fr)$	$=$	0.5333	$P(GC = none \mid \neg fr)$	$=$	0.6522
$P(GC = guarantor \mid fr)$	$=$	0.2667	$P(GC = guarantor \mid \neg fr)$	$=$	0.1304
$P(GC = coapplicant \mid fr)$	$=$	0.2	$P(GC = coapplicant \mid \neg fr)$	$=$	0.2174
$P(ACC = own \mid fr)$	$=$	0.4667	$P(ACC = own \mid \neg fr)$	$=$	0.6087
$P(ACC = rent \mid fr)$	$=$	0.3333	$P(ACC = rent \mid \neg fr)$	$=$	0.2174
$P(ACC = Free \mid fr)$	$=$	0.2	$P(ACC = Free \mid \neg fr)$	$=$	0.1739

Table 2: The Laplacian smoothed, with $k = 3$, probabilities needed by a Naive Bayes prediction model calculated from the fraud detection dataset. Notation key: FR=FRAUDULENT, CH=CREDIT HISTORY, GC = GUARANTOR/COAPPLICANT, ACC = ACCOMODATION, T='*True*', F='*False*'.

Credit History	Guarantor/CoApplicant	Accommodation	Fraudulent
paid	guarantor	free	?

$P(fr)$	$=$	0.3	$P(\neg fr)$	$=$	0.7
$P(CH = paid \mid fr)$	$=$	0.2222	$P(CH = paid \mid \neg fr)$	$=$	0.2692
$P(GC = guarantor \mid fr)$	$=$	0.2667	$P(GC = guarantor \mid \neg fr)$	$=$	0.1304
$P(ACC = Free \mid fr)$	$=$	0.2	$P(ACC = Free \mid \neg fr)$	$=$	0.1739
$\left(\prod_{k=1}^{m} P(\mathbf{q}[m] \mid fr)\right) \times P(fr) = 0.0036$					
$\left(\prod_{k=1}^{m} P(\mathbf{q}[m] \mid \neg fr)\right) \times P(\neg fr) = 0.0043$					

Table 3: The relevant smoothed probabilities, from Table 2, needed by the Naive Bayes prediction model in order to classify the query from the previous slide and the calculation of the scores for each candidate classification.

Continuous Features

Binning

- Two of the best known binning techniques: **equal-width** and **equal-frequency**.
- We can use these techniques to *bin* continuous features into categorical features.
- In general we recommend **equal-frequency binning**.

☞ We can also represent continuous features directly using *probability density functions*!

Probability Density Functions

- A **probability density function** (PDF) represents the probability distribution of a continuous feature using a mathematical function, such as the normal distribution.

$$N(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

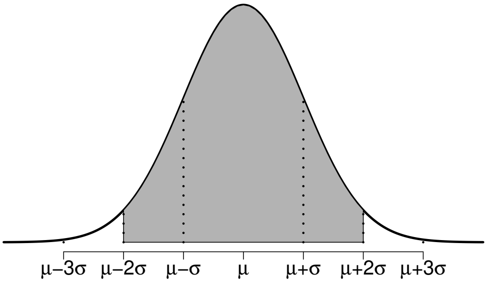
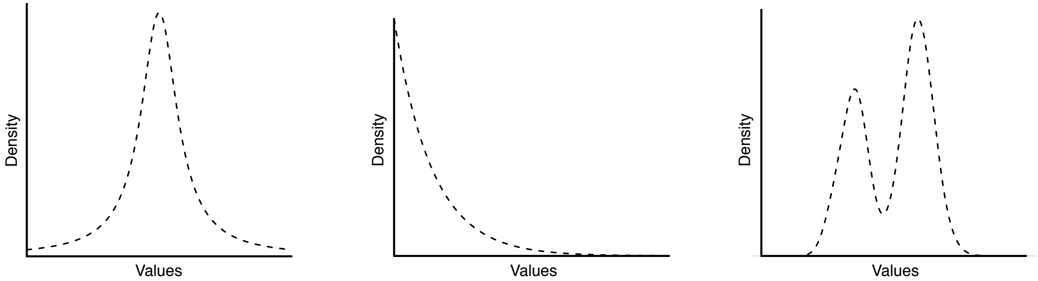


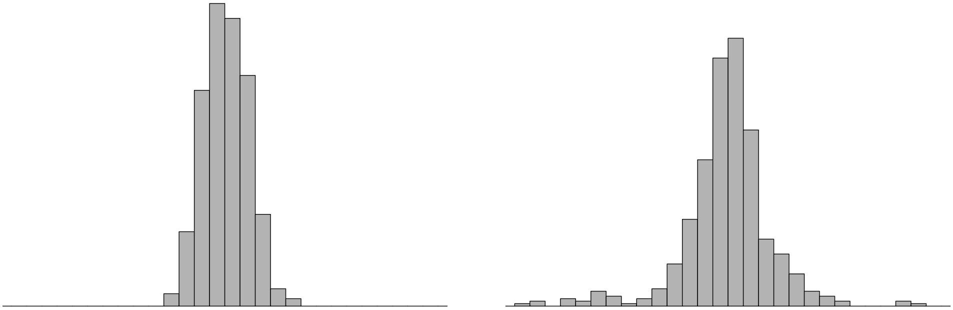
Table 4: Definitions of some standard probability distributions.

Normal $x \in \mathbb{R}$ $\mu \in \mathbb{R}$ $\sigma \in \mathbb{R}_{>0}$	$N(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
Student-t $x \in \mathbb{R}$ $\phi \in \mathbb{R}$ $\rho \in \mathbb{R}_{>0}$ $\kappa \in \mathbb{R}_{>0}$ $z = \frac{x-\phi}{\rho}$	$\tau(x, \phi, \rho, \kappa) = \frac{\Gamma(\frac{\kappa+1}{2})}{\Gamma(\frac{\kappa}{2}) \times \sqrt{\pi\kappa} \times \rho} \times \left(1 + \left(\frac{1}{\kappa} \times z^2\right)\right)^{-\frac{\kappa+1}{2}}$
Exponential $x \in \mathbb{R}$ $\lambda \in \mathbb{R}_{>0}$	$E(x, \lambda) = \begin{cases} \lambda e^{-\lambda x} & \text{for } x > 0 \\ 0 & \text{otherwise} \end{cases}$
Mixture of n Gaussians $x \in \mathbb{R}$ $\{\mu_1, \ldots, \mu_n \mid \mu_i \in \mathbb{R}\}$ $\{\sigma_1, \ldots, \sigma_n \mid \sigma_i \in \mathbb{R}_{>0}\}$ $\{\omega_1, \ldots, \omega_n \mid \omega_i \in \mathbb{R}_{>0}\}$ $\sum_{i=1}^{n} \omega_i = 0$	$N(x, \mu_1, \sigma_1, \omega_1, \ldots, \mu_n, \sigma_n, \omega_n) = \sum_{i=1}^{n} \frac{\omega_i}{\sigma_i\sqrt{2\pi}} e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}}$



(a) Normal/Student-t (b) Exponential (c) Mixture of Gaussians

Figure 1: Plots of some well known probability distributions.



(a)

(b)

Figure 2: Histograms of two unimodal datasets: (a) the distribution has light tails; (b) the distribution has fat tails.

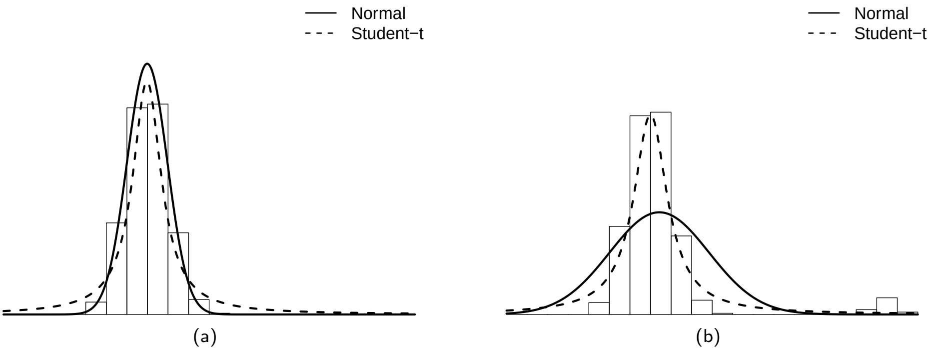


Figure 3: Illustration of the robustness of the student-t distribution to outliers: (a) a density histogram of a unimodal dataset overlaid with the density curves of a normal and a student-t distribution that have been fitted to the data; (b) a density histogram of the same dataset with outliers added, overlaid with the density curves of a normal and a student-t distribution that have been fitted to the data. The student-t distribution is less affected by the introduction of outliers. (This figure is inspired by Figure 2.16 in (Bishop, 2006).)

Fitting Probability Distributions

Choosing a PDF to Fit

1. Draw density histogram.
2. Compare shape to standard distributions.
3. Select the one matching best.

Fitting a PDF to Data

- Gaussian: compute mean μ and standard deviation σ from data
- Exponential: set λ ("drop off rate") to 1 divided by mean
- Student-t, Mixture of Gaussians: no closed form, requires guided search (like gradient descent)

☞ Fitting PDFs is supported by many data analytics packages and APIs!

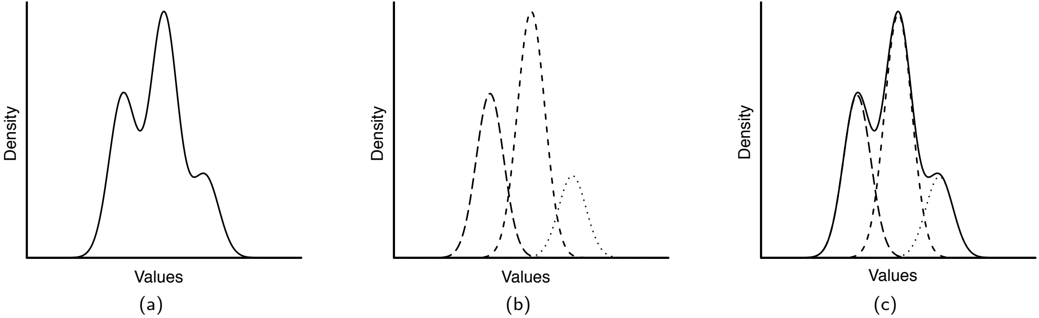


Figure 4: Illustration of how a mixture of Gaussians model is composed of a number of normal distributions. The curve plotted using a solid line is the mixture of Gaussians density curve, created using an appropriately weighted summation of the three normal curves, plotted using dashed and dotted lines.

- A PDF is an abstraction over a density histogram and consequently PDF represents probabilities in terms of area under the curve.
- To use a PDF to calculate a probability we need to think in terms of the area under an interval of the PDF curve.
- We can calculate the area under a PDF by looking this up in a probability table or to use integration to calculate the area under the curve within the bounds of the interval.

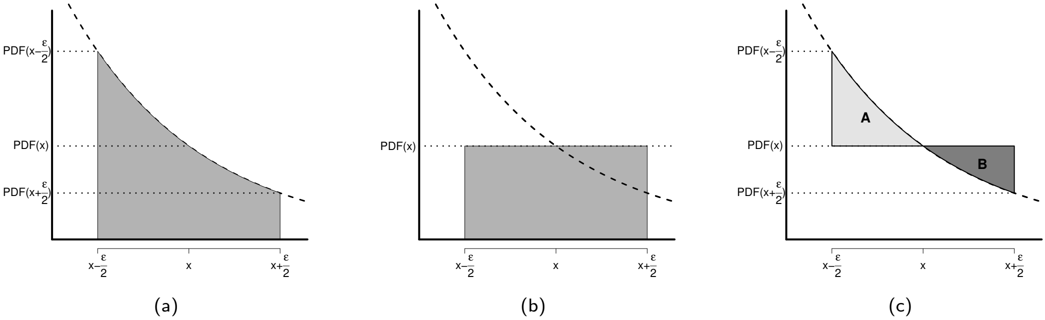


Figure 5: (a) The area under a density curve between the limits $x - \frac{\epsilon}{2}$ and $x + \frac{\epsilon}{2}$; (b) the approximation of this area computed by $PDF(x) \times \epsilon$; and (c) the error in the approximation is equal to the difference between area A, the area under the curve omitted from the approximation, and area B, the area above the curve erroneously included in the approximation. Both of these areas will get smaller as the width of the interval gets smaller, resulting in a smaller error in the approximation.

Idea

As an **approximation**, we can multiply the value at x by the size of the interval!
For small intervals, the error is **negligable**!

- There is no hard and fast rule for deciding on interval size - instead, this decision is done on a case by case basis and is dependent on the precision required in answering a question.
- Sometimes, an interval size is given by the problem:
  financial application $\rightarrow$ 1 cent, temperatures $\rightarrow$ 1 degree
- To illustrate how PDFs can be used in Naive Bayes models we will extend our loan application fraud detection query to have an ACCOUNT BALANCE feature

Table 5: The dataset from the loan application fraud detection domain with a new continuous descriptive features added: ACCOUNT BALANCE

ID	CREDIT HISTORY	GUARANTOR/ COAPPLICANT	ACCOMMODATION	ACCOUNT BALANCE	FRAUD
1	current	none	own	56.75	true
2	current	none	own	1,800.11	false
3	current	none	own	1,341.03	false
4	paid	guarantor	rent	749.50	true
5	arrears	none	own	1,150.00	false
6	arrears	none	own	928.30	true
7	current	none	own	250.90	false
8	arrears	none	own	806.15	false
9	current	none	rent	1,209.02	false
10	none	none	own	405.72	true
11	current	coapplicant	own	550.00	false
12	current	none	free	223.89	true
13	current	none	rent	103.23	true
14	paid	none	own	758.22	false
15	arrears	none	own	430.79	false
16	current	none	own	675.11	false
17	arrears	coapplicant	rent	1,657.20	false
18	arrears	none	free	1,405.18	false
19	arrears	none	own	760.51	false
20	current	none	own	985.41	false

- We need to define two PDFs for the new ACCOUNT BALANCE (AB) feature with each PDF conditioned on a different value in the domain or the target:
  - $P(AB = X|fr) = PDF_1(AB = X|fr)$
  - $P(AB = X|\neg fr) = PDF_2(AB = X|\neg fr)$
- Note that these two PDFs do not have to be defined using the same statistical distribution.

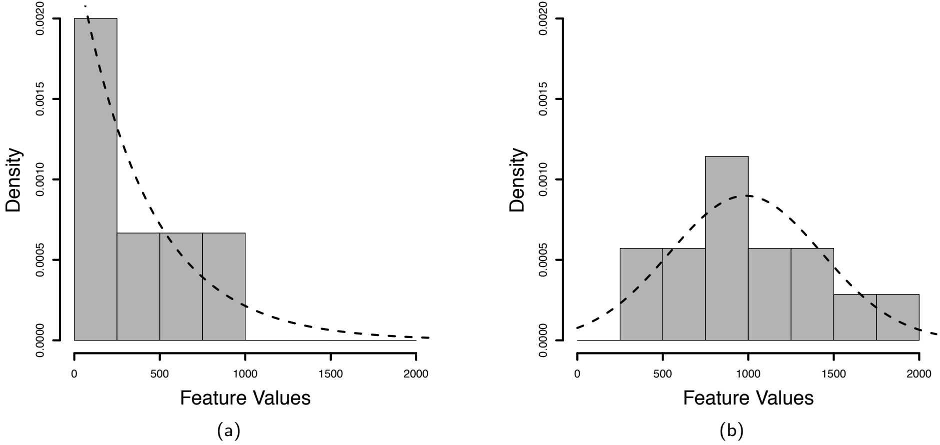


Figure 6: Histograms, using a bin size of 250 units, and density curves for the ACCOUNT BALANCE feature: (a) the fraudulent instances overlaid with a fitted exponential distribution; (b) the non-fraudulent instances overlaid with a fitted normal distribution.

- From the shape of these histograms it appears that
  - the distribution of values taken by the ACCOUNT BALANCE feature in the set of instances where the target feature FRAUDULENT=*'True'* follows an exponential distribution
  - the distributions of values taken by the ACCOUNT BALANCE feature in the set of instances where the target feature FRAUDULENT=*'False'* is similar to a normal distribution.
- Once we have selected the distributions the next step is to fit the distributions to the data.

- To fit the exponential distribution we simply compute the sample mean, $\bar{x}$, of the ACCOUNT BALANCE feature in the set of instances where FRAUDULENT=*'True'* and set the λ parameter equal to one divided by $\bar{x}$.
- To fit the normal distribution to the set of instances where FRAUDULENT=*'False'* we simply compute the sample mean and sample standard deviation, s, for the ACCOUNT BALANCE feature for this set of instances and set the parameters of the normal distribution to these values.

Table 6: Partitioning the dataset based on the value of the target feature and fitting the parameters of a statistical distribution to model the ACCOUNT BALANCE feature in each partition.

ID	...	ACCOUNT BALANCE	FRAUD
1		56.75	true
4		749.50	true
6		928.30	true
10	...	405.72	true
12		223.89	true
13		103.23	true
$\overline{\mathrm{AB}}$		411.22	
$\lambda = {}^{1}!/_{\overline{\mathrm{AB}}}$		0.0024	

ID	...	ACCOUNT BALANCE	FRAUD
2		1 800.11	false
3		1 341.03	false
5		1 150.00	false
7		250.90	false
8		806.15	false
9		1 209.02	false
11		550.00	false
14		758.22	false
15		430.79	false
16		675.11	false
17		1 657.20	false
18		1 405.18	false
19		760.51	false
20		985.41	false
$\overline{\mathrm{AB}}$		984.26	
$sd(\mathrm{AB})$		460.94	

Table 7: The Laplace smoothed (with $k = 3$) probabilities needed by a naive Bayes prediction model calculated from the dataset in Table 5, extended to include the conditional probabilities for the new ACCOUNT BALANCE feature, which are defined in terms of PDFs.

$P(fr)$	$=$	0.3	$P(\neg fr)$	$=$	0.7
$P(CH = none \mid fr)$	$=$	0.2222	$P(CH = none \mid \neg fr)$	$=$	0.1154
$P(CH = paid \mid fr)$	$=$	0.2222	$P(CH = paid \mid \neg fr)$	$=$	0.2692
$P(CH = current \mid fr)$	$=$	0.3333	$P(CH = current \mid \neg fr)$	$=$	0.2692
$P(CH = arrears \mid fr)$	$=$	0.2222	$P(CH = arrears \mid \neg fr)$	$=$	0.3462
$P(GC = none \mid fr)$	$=$	0.5333	$P(GC = none \mid \neg fr)$	$=$	0.6522
$P(GC = guarantor \mid fr)$	$=$	0.2667	$P(GC = guarantor \mid \neg fr)$	$=$	0.1304
$P(GC = coapplicant \mid fr)$	$=$	0.2	$P(GC = coapplicant \mid \neg fr)$	$=$	0.2174
$P(ACC = own \mid fr)$	$=$	0.4667	$P(ACC = own \mid \neg fr)$	$=$	0.6087
$P(ACC = rent \mid fr)$	$=$	0.3333	$P(ACC = rent \mid \neg fr)$	$=$	0.2174
$P(ACC = free \mid fr)$	$=$	0.2	$P(ACC = free \mid \neg fr)$	$=$	0.1739
$P(AB = x \mid fr)$			$P(AB = x \mid \neg fr)$		
	$\approx$	$E\left(\begin{array}{c} x, \\ \lambda = 0.0024 \end{array}\right)$		$\approx$	$N\left(\begin{array}{c} x, \\ \mu = 984.26, \\ \sigma = 460.94 \end{array}\right)$

Table 8: A query loan application from the fraud detection domain.

Credit History	Guarantor/ CoApplicant	Accomodation	Account Balance	Fraudulent
paid	guarantor	free	759.07	?

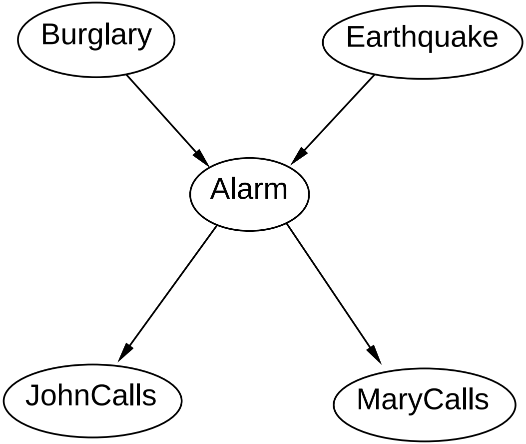
Table 9: The probabilities, from Table 7, needed by the naive Bayes prediction model to make a prediction for the query $\langle \mathrm{CH} = \textit{'paid'}, \mathrm{GC} = \textit{'guarantor'}, \mathrm{ACC} = \textit{'free'}, \mathrm{AB} = 759.07 \rangle$ and the calculation of the scores for each candidate prediction.

$P(fr)$	$=$	0.3	$P(\neg fr)$	$=$	0.7
$P(CH = paid \mid fr)$	$=$	0.2222	$P(CH = paid \mid \neg fr)$	$=$	0.2692
$P(GC = guarantor \mid fr)$	$=$	0.2667	$P(GC = guarantor \mid \neg fr)$	$=$	0.1304
$P(ACC = free \mid fr)$	$=$	0.2	$P(ACC = free \mid \neg fr)$	$=$	0.1739
$P(AB = 759.07 \mid fr)$ $\approx E\begin{pmatrix} 759.07, \\ \lambda = 0.0024 \end{pmatrix}$	$=$	0.00039	$P(AB = 759.07 \mid \neg fr)$ $\approx N\begin{pmatrix} 759.07, \\ \mu = 984.26, \\ \sigma = 460.94 \end{pmatrix}$	$=$	0.00077

$$\left(\prod_{k=1}^{m} P(\mathbf{q}[k] \mid fr)\right) \times P(fr) = 0.0000014$$
$$\left(\prod_{k=1}^{m} P(\mathbf{q}[k] \mid \neg fr)\right) \times P(\neg fr) = 0.0000033$$

Bayesian Nets

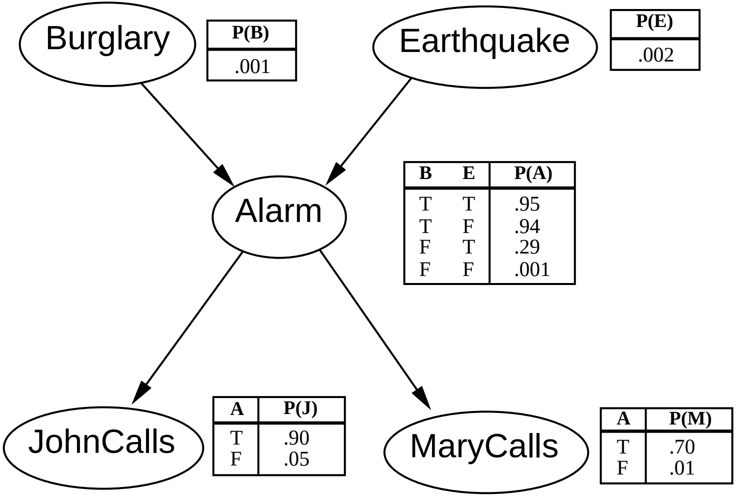
Belief Networks (1)



Idea: Only represent causal connections. Surprisingly simple in many applications!

Belief Networks (2)

Same Example with labelled nodes $\mathbf{P}(X \mid Parents(X))$:



P(B)
.001

P(E)
.002

B	E	P(A)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J)
T	.90
F	.05

A	P(M)
T	.70
F	.01

Belief Networks in General

A belief network is an acyclic graph where

- the nodes represent random variables;
- each node X is labelled with the conditional probabilities

$$\mathbf{P}(X \mid Parents(X)),$$

  where Y is in $Parents(X)$ if there is an edge from Y to X.
  (The label is called a Conditional Probability Table (CPT).)

The tolopology of the network should be chosen in such a way that for each edge from Y to X, the parent node Y has direct causal influence on X.

Belief Networks and Joint Distributions

Let $X_1, \ldots, X_n$ be random variables. We abbreviate $P(X_1 = x_1, \cdots, X_n = x_n)$ as $P(x_1, \ldots, x_n)$.

We can rewrite the joint distribution in the following way:

$$P(x_1, \ldots, x_n) = P(x_n \mid x_{n-1}, \ldots, x_1) \times P(x_{n-1}, \ldots, x_1)$$

Applying this rewriting recursively we get

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid x_{i-1}, \ldots, x_1)$$

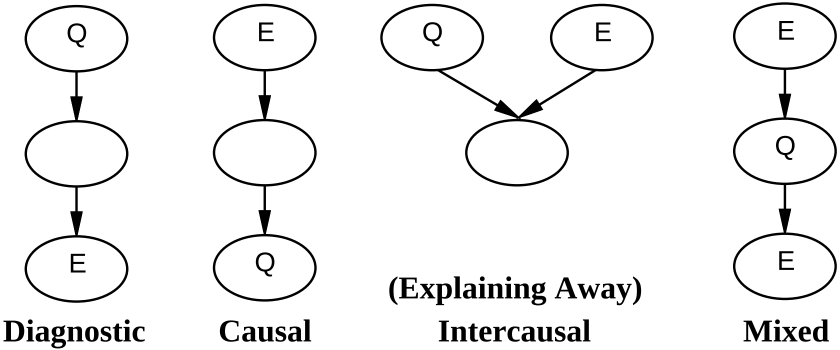
A Belief network is a correct representation of a joint distribution if

$$\mathbf{P}(X_i \mid X_{i-1}, \ldots, X_1) = \mathbf{P}(X_i \mid Parents(X_i)) \text{ and } Parents(X_i) \subseteq \{X_{i-1}, \ldots, X_1\}.$$

In other words, each node must be conditionally independent of its predecessors given its parents.

- **Bayesian networks** use a graph-based representation to encode the structural relationships—such as direct influence and conditional independence—between subsets of features in a domain.
- Consequently, a Bayesian network representation is generally more compact than a full joint distribution, yet is not forced to assert global conditional independence between all descriptive features.

Kinds of Inferences in Belief Networks



Query Evaluation in Bayesian Nets

Computing conditional probabilities if some values are missing is more complex as we need to resort to **summing out** again.

In general, if X is a single query variable, $E = \{E_1, \ldots, E_m\}$ evidence variables, $Y = \{Y_1, \ldots, Y_n\}$ other unmentioned variables:

Want: $\mathbf{P}(X \mid e)$, where e stands for values for $E_1, \ldots, E_m$.

$$\mathbf{P}(X \mid e) = \alpha \mathbf{P}(X, e) = \alpha \sum_{y} \mathbf{P}(X, e, y)$$

The joint probabilities can then be determined from the CPTs in the Bayes net.

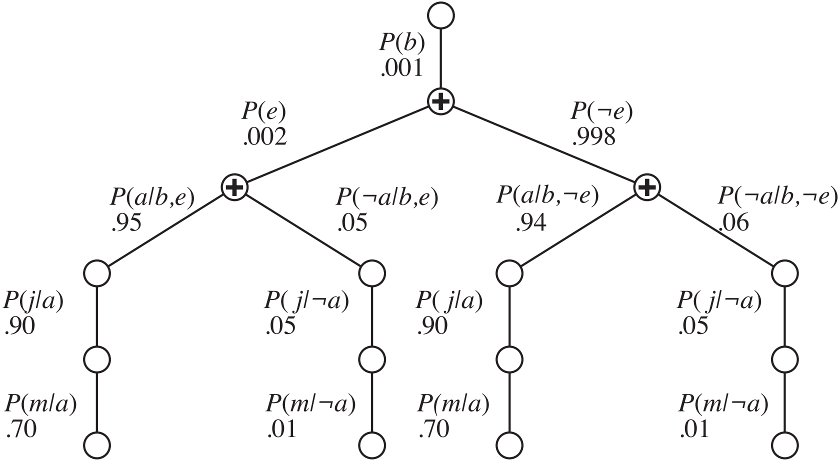
Query Evaluation in Bayesian Nets

Example

$$
\begin{aligned}
P(b \mid j, m) = \alpha P(b, j, m) &= \alpha \sum_{e} \sum_{a} P(m, j, e, a, b) \\
&= \alpha \sum_{e} \sum_{a} P(m \mid j, e, a, b) P(j \mid e, a, b) P(a \mid e, b) P(e \mid b) P(b) \\
&= \alpha \sum_{e} \sum_{a} P(m \mid a) P(j \mid a) P(a \mid e, b) P(e) P(b) \\
&= \alpha P(b) \sum_{e} P(e) \sum_{a} P(m \mid a) P(j \mid a) P(a) \\
&= \alpha \cdot 0.00059224 \\
&= 0.284
\end{aligned}
$$

(To obtain α determine $P(\neg b \mid j, m) = \alpha \cdot 0.00014919$ and use that $P(b \mid j, m) + P(\neg b \mid j, m) = 1$.)

Example Computation



☞ In general, **caching** can help to avoid repeated calculations.

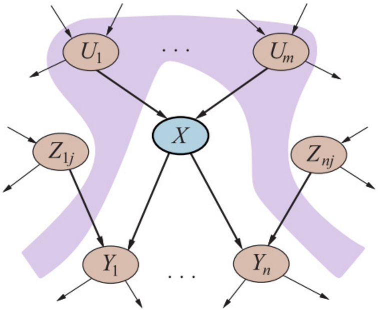
Computational Complexity

The problem is NP-hard for multiply connected networks.

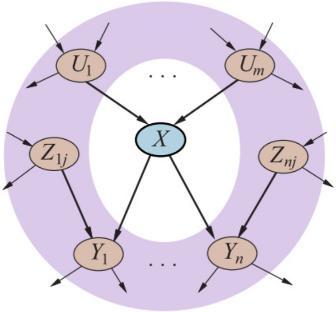
(Actually, the problem is at least as hard as enumerating all satisfying assignments of a propositional formula (#P-hard), which is strictly harder than NP-completeness.)

It is linear in the case of singly connected networks.

Two Types of Conditional Independence in Bayesian Nets



(a)



(b)

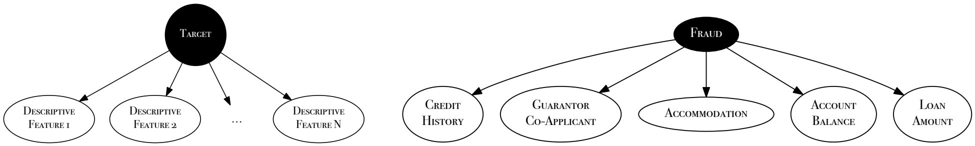
(a) A node X is conditionally independent of *its non-descendants* (the Zs) given its parents (Us).

(b) A node X is conditionally independent of *all other nodes* given its **Markov blanket** (grey area).

Conditional independence of node x_i in graph with n nodes (Markov Blanket)

$$P(x_i|x_1,\dots,x_{i-1},x_{i+1},\dots,x_n) = P(x_i|Parents(x_i)) \prod_{j \in Children(x_i)} P(x_j|Parents(x_j)) \tag{1}$$

A naive Bayes classifier is a Bayesian network with a specific topological structure.

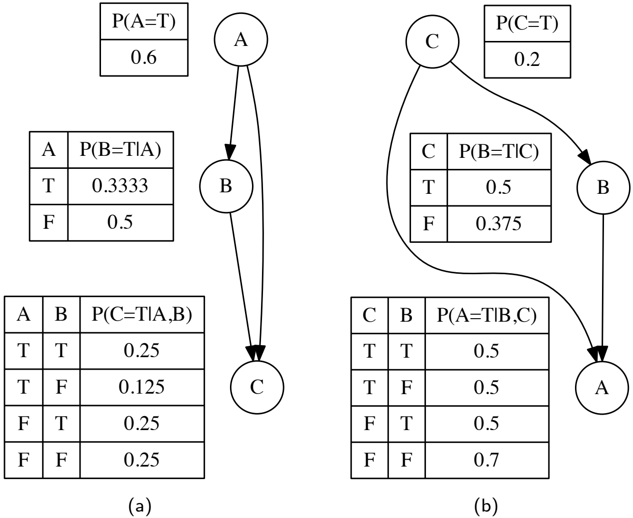


Computing a conditional probability for a target feature using a naive Bayes model:

$$P(t|\mathbf{d}[1], \ldots, \mathbf{d}[n]) = P(t) \prod_{j \in Children(t)} P(\mathbf{d}[j]|t)$$

☞ Special case of Equation (1)!

- This example illustrates the power of Bayesian networks.
    - When complete knowledge of the state of all the nodes in the network is not available, we clamp the values of nodes that we do have knowledge of and sum out the unknown nodes.



P(A=T)
0.6

A	P(B=T|A)
T	0.3333
F	0.5

A	B	P(C=T|A,B)
T	T	0.25
T	F	0.125
F	T	0.25
F	F	0.25

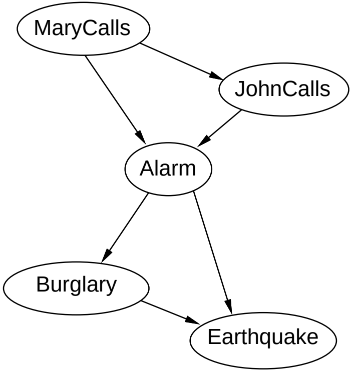
P(C=T)
0.2

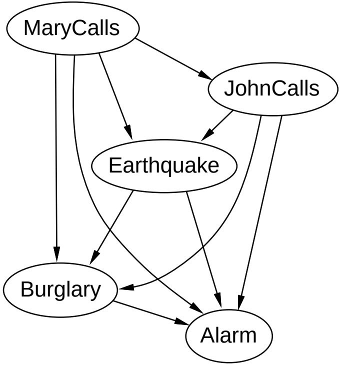
C	P(B=T|C)
T	0.5
F	0.375

C	B	P(A=T|B,C)
T	T	0.5
T	F	0.5
F	T	0.5
F	F	0.7

Figure 7: Two different Bayesian networks, each defining the same full joint probability distribution.

The Ordering of Nodes Matters





A bad choice in the ordering of the variables leads to large networks.

Orderings in the examples:

Left: MaryCalls, JohnCalls, Alarm, Burglary, Earthquake.

Right: MaryCalls, JohnCalls, Earthquake, Burglary, Alarm.

Constructing Bayesian Networks

Learning the **structure** of a Bayesian Net is very difficult. It is usally done by means of some form of guided local search.

The simpler way to construct a Bayesian network is to use a hybrid approach where:

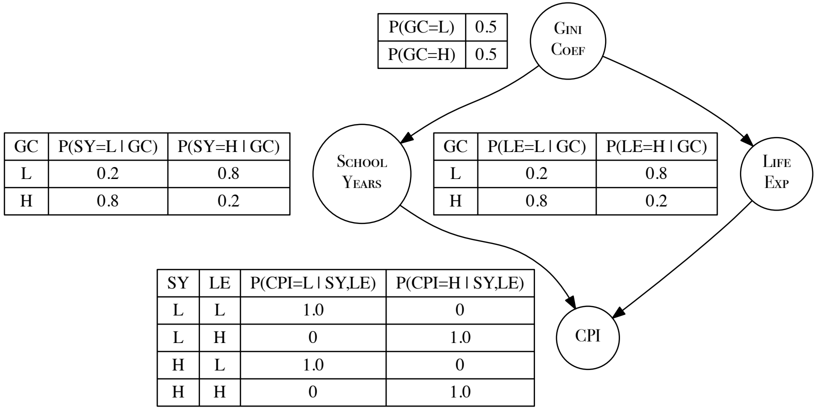
1. the topology of the network is given to the learning algorithm,
2. and the learning task involves inducing the CPT from the data.

☞ This is called **parameter learning** and makes use of expert knowledge.

Table 10: (a) Some socio-economic data for a set of countries; (b) a binned version of the data listed in (a).

Country ID	Gini Coef	School Years	Life Exp	CPI	Gini Coef	School Years	Life Exp	CPI
Afghanistan	27.82	0.40	59.61	1.52	low	low	low	low
Argentina	44.49	10.10	75.77	3.00	high	low	low	low
Australia	35.19	11.50	82.09	8.84	low	high	high	high
Brazil	54.69	7.20	73.12	3.77	high	low	low	low
Canada	32.56	14.20	80.99	8.67	low	high	high	high
China	42.06	6.40	74.87	3.64	high	low	low	low
Egypt	30.77	5.30	70.48	2.86	low	low	low	low
Germany	28.31	12.00	80.24	8.05	low	high	high	high
Haiti	59.21	3.40	45.00	1.80	high	low	low	low
Ireland	34.28	11.50	80.15	7.54	low	high	high	high
Israel	39.2	12.50	81.30	5.81	low	high	high	high
New Zealand	36.17	12.30	80.67	9.46	low	high	high	high
Nigeria	48.83	4.10	51.30	2.45	high	low	low	low
Russia	40.11	12.90	67.62	2.45	high	high	low	low
Singapore	42.48	6.10	81.788	9.17	high	low	high	high
South Africa	63.14	8.50	54.547	4.08	high	low	low	low
Sweden	25.00	12.80	81.43	9.30	low	high	high	high
U.K.	35.97	13.00	80.09	7.78	low	high	high	high
U.S.A	40.81	13.70	78.51	7.14	high	high	high	high
Zimbabwe	50.10	6.7	53.684	2.23	high	low	low	low

(a) (b)



P(GC=L)	0.5
P(GC=H)	0.5

GC	P(SY=L | GC)	P(SY=H | GC)
L	0.2	0.8
H	0.8	0.2

GC	P(LE=L | GC)	P(LE=H | GC)
L	0.2	0.8
H	0.8	0.2

SY	LE	P(CPI=L | SY,LE)	P(CPI=H | SY,LE)
L	L	1.0	0
L	H	0	1.0
H	L	1.0	0
H	H	0	1.0

Figure 8: A Bayesian network that encodes the causal relationships between the features in the corruption domain. The CPT entries have been calculated using the data from Table 10(b).

Prediction with Bayes Nets

Predicting using a Bayes Nets is as before by means of the maximum a posterio likelihood:

$$\mathbb{M}(\mathbf{q}) = \underset{l \in levels(t)}{\operatorname{argmax}}\, BayesianNetwork(t = l, \mathbf{q}) \tag{2}$$

Example

Predict $\text{CPI} = \textit{'high'}$ if $\text{Gini Coef} = \textit{'high'}$, $\text{School Years} = \textit{'high'}$.

$$P(CPI = H|SY = H, GC = H) = \frac{P(CPI = H, SY = H, GC = H)}{P(SY = H, GC = H)}$$

$$= \frac{\sum_{i \in H,L} P(CPI = H, SY = H, GC = H, LE = i)}{P(SY = H, GC = H)} = \frac{0.02}{0.1} = 0.2$$

$$\sum_{i \in \{H,L\}} P(CPI = H, SY = H, GC = H, LE = i)$$

$$= \sum_{i \in \{H,L\}} P(CPI = H|SY = H, LE = i) \times P(SY = H|GC = H)$$

$$\times P(LE = i|GC = H) \times P(GC = H)$$

$$= (1.0 \times 0.2 \times 0.2 \times 0.5) + (0 \times 0.2 \times 0.8 \times 0.5) = 0.02$$

$$P(SY = H, GC = H) = P(SY = H|GC = H) \times P(GC = H) = 0.2 \times 0.5 = 0.1$$

advantage: can answer queries with missing values, no need to *impute* them etc.

- Because of the calculation complexity that can arise when using Bayesian networks to do exact inference a popular approach is to approximate the required probability distribution using **Markov Chain Monte Carlo** algorithms.
- **Gibbs sampling** is one of the best known MCMC algorithms.

Table 11: Examples of the samples generated using Gibbs sampling.

Sample Number	Gibbs Iteration	Feature Updated	GINI COEF	SCHOOL YEARS	LIFE EXP	CPI
1	37	CPI	high	high	high	low
2	44	LIFE EXP	high	high	high	low
3	51	CPI	high	high	high	low
4	58	LIFE EXP	high	high	low	high
5	65	CPI	high	high	high	low
6	72	LIFE EXP	high	high	high	low
7	79	CPI	high	high	low	high
8	86	LIFE EXP	high	high	low	low
9	93	CPI	high	high	high	low
10	100	LIFE EXP	high	high	high	low
11	107	CPI	high	high	low	high
12	114	LIFE EXP	high	high	high	low
13	121	CPI	high	high	high	low
14	128	LIFE EXP	high	high	high	low
15	135	CPI	high	high	high	low
16	142	LIFE EXP	high	high	low	low
		...				

$$\mathbb{M}(\mathbf{q}) = \underset{l \in levels(t)}{\operatorname{argmax}}\ Gibbs\left(t = l, \mathbf{q}\right) \tag{3}$$

Summary

- Naive Bayes models can suffer from zero probabilities of relatively rare events. **Smoothing** is an easy way to combat this.
- Two ways to handle continuous features in probability-based models are: **Probability density functions** and **Binning**
- Using probability density functions requires that we match the observed data to an existing distribution.
- Although binning results in information loss it is a simple and effective way to handle continuous features in probability-based models.
- Bayesian network representation is generally more compact than a full joint distribution, yet is not forced to assert global conditional independence between all descriptive features.

References

Reading

- Mitchell, T. M. (1997). *Machine Learning* (Vol. 1). McGraw-Hill New York. Chapter 6.
- Russell S. J. & Norvig P. (2020). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson. Chapter 13.
- Kelleher, Mac Namee, B., & D'Arcy, A. (2020). *Fundamentals of Machine Learning for Predictive Data Analytics Algorithms, Worked Examples, and Case Studies.* MIT Press. Chapter 6.

Acknowledgements

In addition to the authors above, some slides are adapted from or inspired by a course by Gerhard Lakemeyer (RWTH Aachen University).