

Artificial Intelligence

Probabilistic reasoning:
supervised learning

Marco Piastra

Machine Learning

Types of machine learning problems

Consider a number of observations (i.e. a dataset) made by an agent

$$\{D^{(1)}, D^{(2)}, \dots, D^{(N)}\}$$

- **Supervised learning**

Learning from complete observations: each of the observations $\{D^{(1)}, D^{(2)}, \dots, D^{(N)}\}$ include values for all the random variables in the model

The objective is learning a *distribution* P

- **Unsupervised learning**

Learning from incomplete observations: observations $\{D^{(1)}, D^{(2)}, \dots, D^{(N)}\}$ do not necessarily include values for all the random variables in the model

The objective is learning a *distribution* P

- **Reinforcement learning**

The observations $\{D^{(1)}, D^{(2)}, \dots, D^{(N)}\}$ are *states* or *situations*, at each state X_i the agent must perform an **action** a_i that produces a **result** r_i .

The objective is learning a *distribution* π over possible actions in each state which describes a *policy* that the agent will follow

Such policy should maximize the expected value of a *reward function* $v(\langle r_1, r_2, \dots, r_n \rangle)$ of the sequence of *results*

Observations and Independence

Each observation could be the outcome of an experiment or a test

The outcome of a particular experiment can be represented by a set of *random variables*

For example, if the model makes use of the two random variables $\{X, Y\}$, the N outcomes of the experiments are $D^{(1)} = (X^{(1)}, Y^{(1)}), \dots, D^{(N)} = (X^{(N)}, Y^{(N)})$

That is, a *dataset*

$$D := \{(X^{(i)}, Y^{(i)})\}_{i=1}^N$$

■ Independent observations, same probability distribution

Independent and Identically Distributed (IID) random variables

Definition

A sequence or a set of random variables $\{X_1, X_2, \dots, X_n\}$ is *Independent and Identically Distributed (IID)* iff:

- 1) $\langle X_i \perp X_j \rangle, \forall i \neq j$ (independence)
- 2) $P(X_i = x) = P(X_j = x), \forall i \neq j, \forall x$ (identical distribution)

CAUTION: *Being IID is not an obvious property of observations*

*e.g. different measurements on different patients may be IID,
but different measurements over time on the same patient are not IID*

ML = Representation + Evaluation + Optimization

Assume that an I.I.D. dataset D is available

■ Representation

The objective is learning a specific distribution

$$P(\{X_r\}; \theta)$$

where $\{X_r\}$ are all the random variables of interest and θ is a set of parameters

Which kind of distribution (i.e. the *model* or also the *learner*) do we select?

Example: assume we select the anti-spam filter (i.e. Naïve Bayesian Classifier) as the model the parameters in such case are the numerical probabilities in the CPTs

■ Evaluation

Given a dataset D , how well does a specific set of parameter values $\hat{\theta}$ make the distribution P fit the dataset?

An estimator, i.e. a scoring function of some sort, must be selected

■ Optimization

How can we find the optimal set of parameter values θ^* with respect to the *estimator* of choice?

In general, this is an optimization problem

Maximum Likelihood Estimator (MLE)

Likelihood

A probabilistic model $P(X)$, with parameters θ

θ is a set of values that characterizes $P(X)$ *completely*: once θ is defined, $P(X)$ is also defined.

A set of IID observations (data items) $D = \{D^{(1)}, \dots, D^{(N)}\}$

■ Likelihood function (or conditional probability)

A function, or a conditional probability, derived from the model $P(X)$

$$L(\theta | D) = P(D | \theta) = P(D^{(1)}, \dots, D^{(N)} | \theta)$$

where $P(D | \theta)$ is the conditional probability that the parameter θ , considered as a random variables, could generate the observations D

When the observations $\{D^{(1)}, \dots, D^{(N)}\}$ are IID:

$$P(D | \theta) = P(D^{(1)} | \theta) \dots P(D^{(N)} | \theta) = \prod_m P(D^{(m)} | \theta)$$

Note the 'trick':
likelihood of the dataset given the parameters

Maximum Likelihood Estimator (MLE)

A probabilistic model $P(X)$, with parameters θ

θ is a set of values that characterizes $P(X)$ *completely*: once θ is defined, $P(X)$ is also defined.

A set of IID observations (data items) $D = \{D^{(1)}, \dots, D^{(N)}\}$

■ Maximum Likelihood Estimation

$$\theta_{ML}^* := \operatorname{argmax}_{\theta} L(\theta|D)$$

Since the observations are IID, using *log-Likelihood* could ease computations:

$$\ell(\theta | D) = \log L(\theta|D) = \log \prod_m P(D^{(m)}|\theta) = \sum_m \log P(D^{(m)}|\theta)$$

$$\theta_{ML}^* = \operatorname{argmax}_{\theta} \ell(\theta|D)$$

true because log is monotonically increasing

Example: coin tossing (*Bernoulli Trials*)

Experiment: tossing a coin X , not necessarily *fair* ($X = 1$ head, $X = 0$ tail)

Parameters: $\theta := \{ \pi \} \Leftrightarrow P(X = 1) = \pi, P(X = 0) = 1 - \pi$

Observations: a sequence of experimental outcomes

$$D = \{D_1 = \{X^{(1)} = x^{(1)}\}, D_2 = \{X^{(2)} = x^{(2)}\}, \dots, D_N = \{X^{(N)} = x^{(N)}\}\}$$

■ Binomial distribution

$$\begin{aligned} P(D|\theta) &= \binom{N}{N_{X=1}} \prod_i P(X^{(i)}|\theta) = \binom{N}{N_{X=1}} P(X = 1|\theta)^{N_{X=1}} P(X = 0|\theta)^{N_{X=0}} \\ &= \binom{N}{N_{X=1}} \pi^{N_{X=1}} (1 - \pi)^{N_{X=0}} \end{aligned}$$

$\binom{N}{k} := \frac{N!}{k!(N-k)!}$ *binomial coefficient*

$N_{X=1}$ is the number of $X=1$ (i.e. heads) in a sequence of N trials

It is the probability of obtaining $N_{X=1}$ times 'head' in a sequence of N trials

In this case, it is assumed to be the likelihood of $\{D^{(1)}, \dots, D^{(N)}\}$ given the parameters θ

MLE as optimization

Example: coin tossing (*Bernoulli Trials*)

■ (Log-)Likelihood Function

$$\ell(\theta|D) = \log P(D|\theta) = \log P(\{X^{(i)}\}|\theta) = \log \binom{N}{N_{X=1}} \prod_i P(X^{(i)}|\theta) = \log \binom{N}{N_{X=1}} + \sum_i \log P(X^{(i)}|\theta)$$

Rewrite $P(X|\theta)$ as:

$$P(X|\theta) = \pi^{[X=1]}(1-\pi)^{[X=0]} \quad \text{where:} \quad [X^{(i)}=v] := \begin{cases} 1 & \text{if } X^{(i)}=v \\ 0 & \text{if } X^{(i)} \neq v \end{cases} \quad \begin{array}{l} \text{Also called} \\ \text{indicator function} \end{array}$$

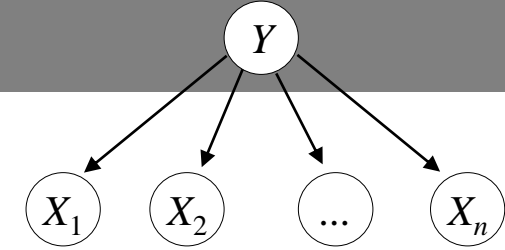
$$\begin{aligned} \ell(\theta|D) &= \log \binom{N}{N_{X=1}} + \sum_i \log \left(\pi^{[X^{(i)}=1]} (1-\pi)^{[X^{(i)}=0]} \right) = \\ &= \log \binom{N}{N_{X=1}} + \log \pi \sum_i [X^{(i)}=1] + \log(1-\pi) \sum_i [X^{(i)}=0] \\ &= \log \binom{N}{N_{X=1}} + N_{X=1} \log \pi + N_{X=0} \log(1-\pi) \end{aligned}$$

■ Maximum Likelihood Estimation

$$\frac{\partial \ell}{\partial \theta} = \frac{\partial \ell}{\partial \pi} = \frac{N_{X=1}}{\pi} - \frac{N_{X=0}}{(1-\pi)} \quad \frac{\partial \ell}{\partial \theta} = 0 \quad \Rightarrow \quad \theta_{ML}^* = \frac{N_{X=1}}{N_{X=1} + N_{X=0}} = \frac{N_{X=1}}{N}$$

Naïve Bayesian Classifier

$$P(Y, X_1, \dots, X_n) = P(Y) \prod_{i=1}^n P(X_i | Y)$$



Parameters: the *conditional probability tables* in the graphical model

$$\theta := \{\pi_k, \pi_{ijk}\} \quad , \quad P(Y = k) =: \pi_k \quad P(X_i = j | Y = k) =: \pi_{ijk}$$

Observations: a set of messages *with classification*

$$D = \{D^{(1)} = \{Y^{(1)} = 1, X_1^{(1)} = 1, \dots, X_n^{(1)} = 0\},$$

$\dots,$

$$D^{(N)} = \{Y_2^{(N)} = y^{(N)}, X_1^{(N)} = x_1^{(N)}, \dots, X_n^{(N)} = x_n^{(N)}\}$$

■ Likelihood Function

$$L(\theta|D) = P(D|\theta) = P(\{D^{(m)}\}|\{\pi_k, \pi_{ijk}\}) = \prod_m P(D^{(m)}|\{\pi_k, \pi_{ijk}\}) \quad (\text{data items are IID})$$

$$\stackrel{\text{(factorization)}}{=} \prod_m P(\{Y^{(m)} = y^{(m)}, X_i^{(m)} = x_i^{(m)}\}|\{\pi_k, \pi_{ijk}\})$$

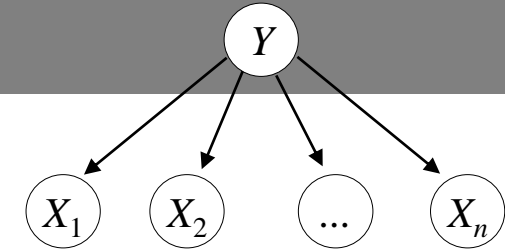
$$\stackrel{\text{(cond. independence)}}{=} \prod_m P(Y^{(m)} = y^{(m)}|\{\pi_k, \pi_{ijk}\}) P(\{X_i^{(m)} = x_i^{(m)}\}|Y^{(m)} = y^{(m)}, \{\pi_k, \pi_{ijk}\})$$

$$\stackrel{\langle X_i \perp X_j | Y \rangle}{=} \prod_m P(Y^{(m)} = y^{(m)}|\{\pi_k\}) P(\{X_i^{(m)} = x_i^{(m)}\}|Y^{(m)} = y^{(m)}, \{\pi_{ijk}\})$$

$$\stackrel{\langle X_i \perp X_j | Y \rangle}{=} \prod_m P(Y^{(m)} = y^{(m)}|\{\pi_k\}) \prod_i P(X_i^{(m)} = x_i^{(m)}|Y^{(m)} = y^{(m)}, \{\pi_{ijk}\})$$

Naïve Bayesian Classifier

$$P(Y, X_1, \dots, X_n) = P(Y) \prod_{i=1}^n P(X_i | Y)$$



■ Log-Likelihood Function

$$\ell(\{\pi_k, \pi_{ijk}\} | D) = \sum_m \log P(Y^{(m)} = y^{(m)} | \{\pi_k\}) + \sum_m \sum_i \log P(X_i^{(m)} = x_i^{(m)} | Y^{(m)} = y^{(m)}, \{\pi_{ijk}\})$$

Alternative form for P : (i.e. rewritten using indicator functions)

$$P(Y = k | \{\pi_k\}) = \prod_k \pi_k^{[Y=k]}$$

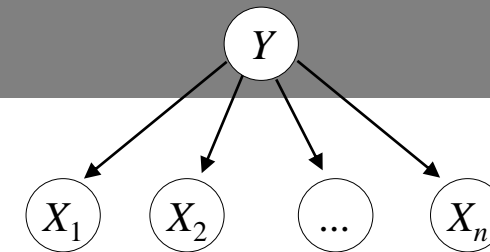
$$P(X_i = j | Y = k, \{\pi_{ijk}\}) = \prod_j \prod_k \pi_{i,j,k}^{[X_i=j][Y=k]}$$

$$\ell(\{\pi_k, \pi_{ijk}\} | D) = \sum_m \sum_k [Y^{(m)} = k] \log \pi_k + \sum_m \sum_i \sum_j \sum_k [X_i^{(m)} = j][Y^{(m)} = k] \log \pi_{ijk}$$

Being both positive and depending on different variables,
the two terms above can be optimized separately

Anti-spam filter

$$P(Y, X_1, \dots, X_n) = P(Y) \prod_{i=1}^n P(X_i | Y)$$



■ Maximum Likelihood Estimation

$$\ell(\{\pi_k, \pi_{ijk}\} | D) = \sum_m \sum_k [Y^{(m)} = k] \log \pi_k + \sum_m \sum_i \sum_j \sum_k [X_i^{(m)} = j][Y^{(m)} = k] \log \pi_{ijk}$$

Optimizing first term:

$$\ell^*(\{\pi_k\} | D) = \sum_m \sum_k [Y^{(m)} = k] \log \pi_k + \lambda \left(1 - \sum_k \pi_k\right)$$

Lagrange multiplier

$$\frac{\partial \ell^*}{\partial \pi_k} = \frac{\sum [Y^{(m)} = k]}{\pi_k} - \lambda$$

number of messages in D classified as k

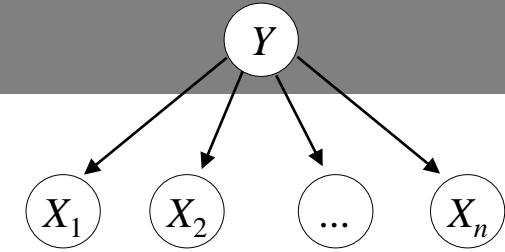
$$\frac{\partial \ell^*}{\partial \pi_k} = 0 \Rightarrow \pi_k = \frac{N_{Y=k}}{\lambda}$$

$$\sum_k \pi_k = 1 \Rightarrow \sum_k \frac{N_{Y=k}}{\lambda} = 1 \Rightarrow \lambda = \sum_k N_{Y=k} = N$$

$$\pi_k^* = \frac{N_{Y=k}}{N} \text{ (Maximum Likelihood Estimator of } \pi_k \text{)}$$

Naïve Bayesian Classifier

$$P(Y, X_1, \dots, X_n) = P(Y) \prod_{i=1}^n P(X_i | Y)$$



Maximum Likelihood Estimation

$$\ell(\{\pi_k, \pi_{ijk}\} | D) = \sum_m \sum_k [Y^{(m)} = k] \log \pi_k + \sum_m \sum_i \sum_j \sum_k [X_i^{(m)} = j][Y^{(m)} = k] \log \pi_{ijk}$$

Optimizing second term:

$$\ell^*(\{\pi_{ijk}\} | D) = \sum_m \sum_i \sum_j \sum_k [X_i^{(m)} = j][Y^{(m)} = k] \log \pi_{ijk} + \sum_i \sum_k \lambda_{ik} (1 - \sum_j \pi_{ijk})$$

Lagrange multipliers

$$\frac{\partial \ell^*}{\partial \pi_{ijk}} = \frac{\sum_m [X_i^{(m)} = j][Y^{(m)} = k]}{\pi_{ijk}} - \lambda_{ik}$$

$$\frac{\partial \ell^*}{\partial \pi_{ijk}} = 0 \Rightarrow \pi_{ijk} = \frac{N_{X_i=j, Y=k}}{\lambda_{ik}}$$

$$\sum_j \pi_{ijk} = 1 \Rightarrow \sum_j \frac{N_{X_i=j, Y=k}}{\lambda_{ik}} = 1 \Rightarrow \lambda_{ik} = \sum_j N_{X_i=j, Y=k} = N_{Y=k}$$

$$\pi_{ijk}^* = \frac{N_{X_i=j, Y=k}}{N_{Y=k}} \quad (\text{Maximum Likelihood Estimator of } \pi_{ijk})$$

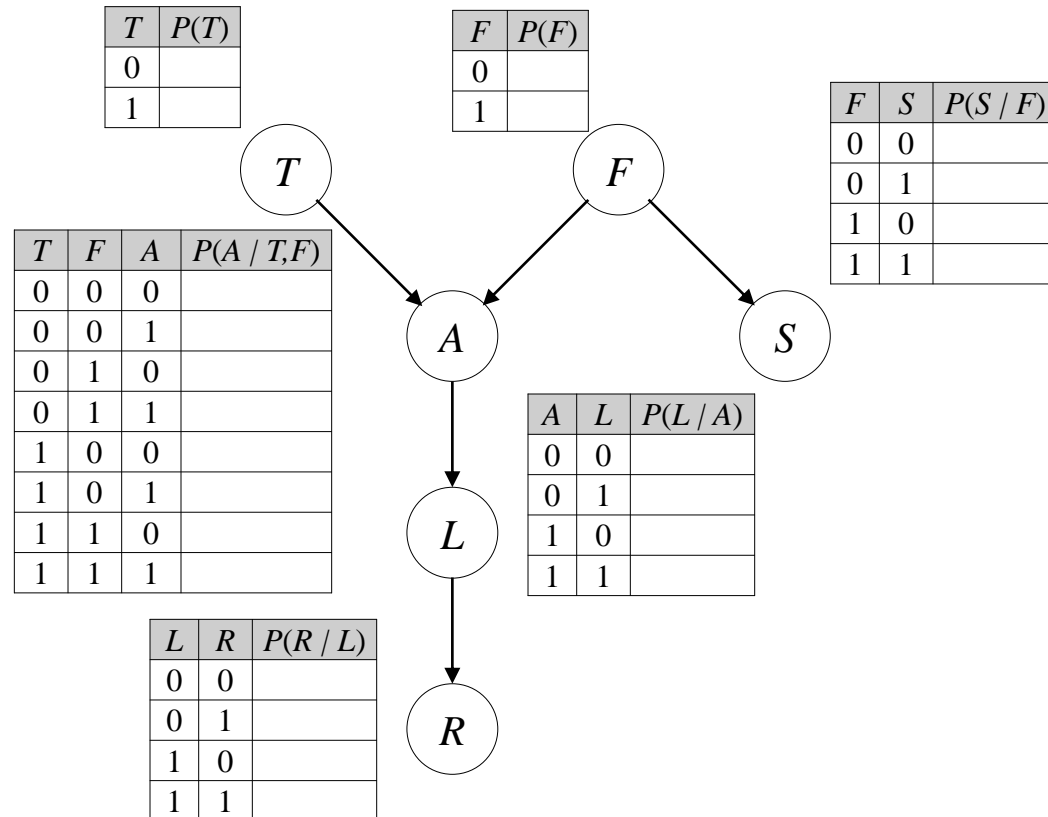
MLE for Graphical Models: A Practical Rule

Learning CPTs for a graphical model via MLE

Model: random variables plus the graph of dependencies

Observations: dataset of values, from completely observed outcomes

Parameters (to be determined): all conditional probabilities (*i.e.* all CPTs)

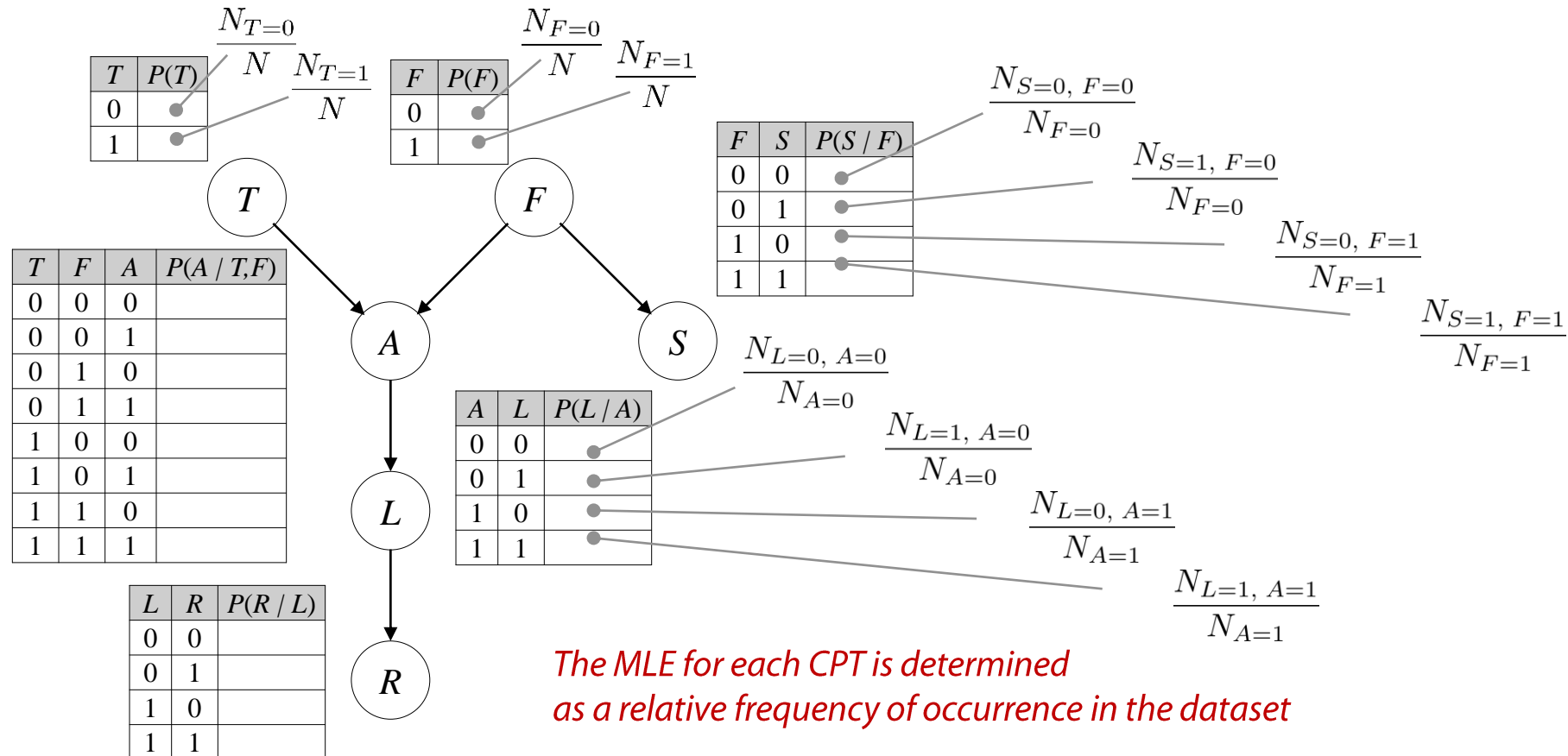


Learning CPTs for a graphical model via MLE

Model: random variables plus the graph of dependencies

Observations: dataset of values, from completely observed outcomes

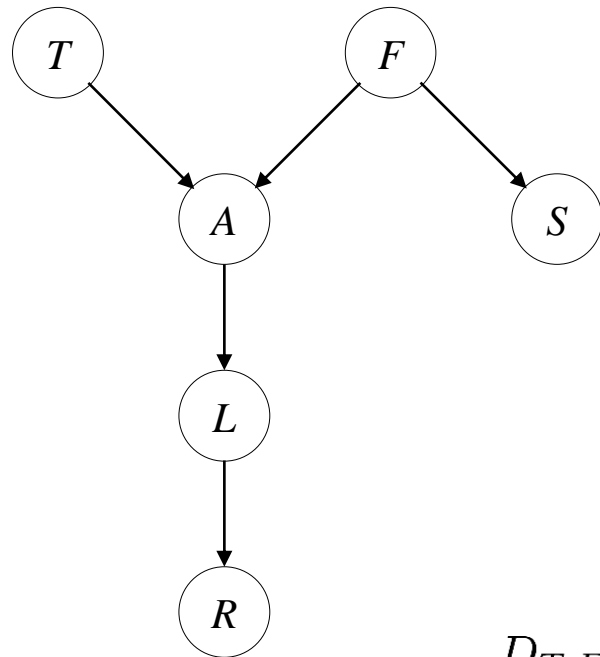
Parameters (to be determined): all conditional probabilities (i.e. all CPTs)



Learning CPTs for a graphical model via MLE

More in general:

The MLE of a (directed) graphical model is the MLE of each node
(in each corresponding observation subset)



$$\theta_{ML}^* := \operatorname{argmax}_{\theta} P(D \mid \theta)$$

$$\theta = \{\pi_T, \pi_F, \pi_{S|F}, \pi_{A|S,F}, \pi_{L|A}, \pi_{R|L}\}$$

$$\pi_T^* := \operatorname{argmax}_{\pi_T} P(D \mid \pi_T)$$

$$\pi_F^* := \operatorname{argmax}_{\pi_F} P(D \mid \pi_F)$$

$$\pi_{S|F}^* := \operatorname{argmax}_{\pi_{S|F}} P(D_F \mid \pi_{S|F})$$

$$\pi_{A|T,F}^* := \operatorname{argmax}_{\pi_{A|T,F}} P(D_{T,F} \mid \pi_{A|T,F})$$

$$\pi_{L|A}^* := \operatorname{argmax}_{\pi_{L|A}} P(D_A \mid \pi_{L|A})$$

$$\pi_{R|L}^* := \operatorname{argmax}_{\pi_{R|L}} P(D_L \mid \pi_{R|L})$$

$D_{T,F}$ denotes the subset of complete observation in which
the random variables T, F have the corresponding values

Bayesian Learning:
Maximum a Posteriori (MAP)
estimator

Bayesian learning

■ *Maximum a Posteriori Estimation (MAP)*

Instead of a *likelihood function*, the a posteriori probability is maximized

$$P(\theta|D) = \frac{P(D|\theta) P(\theta)}{P(D)} = \frac{P(D|\theta) P(\theta)}{\sum_{\theta} P(D|\theta) P(\theta)}$$

Which is equivalent to optimize, w.r.t. θ :

$$P(D|\theta) P(\theta)$$

$$\theta_{MAP}^* := \operatorname{argmax}_{\theta} P(D|\theta) P(\theta)$$

Advantages:

- Regularization: not all possible combinations of values might be present in D
- A formula for incremental learning:
a priori terms could represent what was known *before* observations D

Problem:

- Which *prior* distribution? $P(\theta)$

Beta distribution

Gamma function (n integer > 0)

$$\Gamma(n) := (n - 1)!$$

Beta function (α and β integers > 0)

$$B(\alpha, \beta) := \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha + \beta)} = \frac{(\alpha - 1)!(\beta - 1)!}{(\alpha + \beta - 1)!}$$

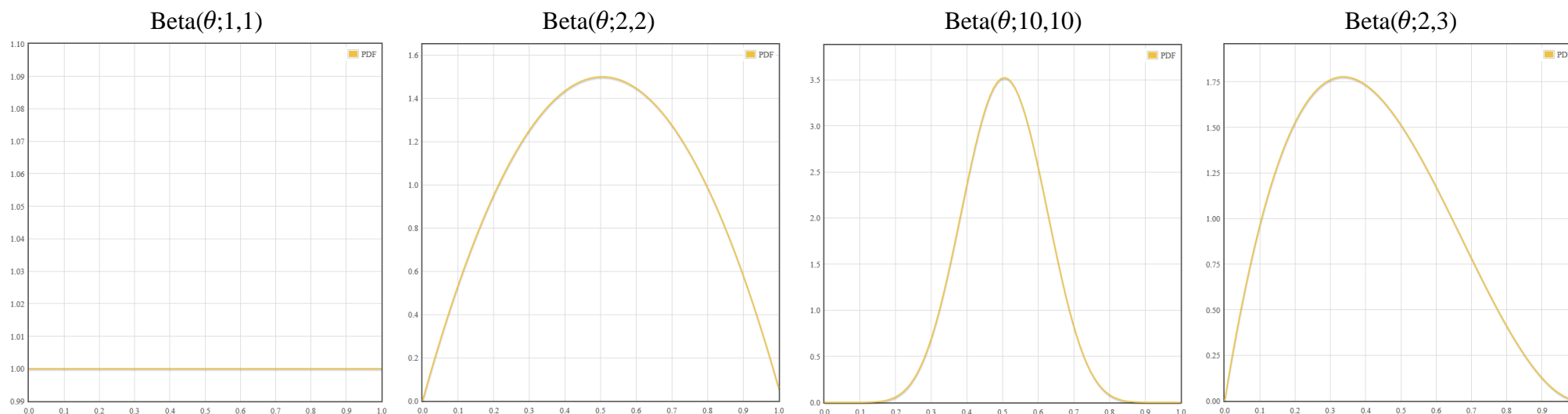
The definition is more complex when α and β are not integers (see Wikipedia)

- Beta probability density function (pdf) (α and β integers > 0)

$$\text{Beta}(\theta; \alpha, \beta) := \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha, \beta)}$$

The maximum occurs at:

$$\theta = \frac{\alpha - 1}{\alpha + \beta - 2}$$



Conjugate prior distributions

Coin tossing (i.e. Binomial)

α_D and β_D are the result counts (i.e. heads and tails)

$$P(D|\theta) = \binom{\alpha_D + \beta_D}{\alpha_D} \prod_i P(X_i|\theta) = \binom{\alpha_D + \beta_D}{\alpha_D} \theta^{\alpha_D} (1 - \theta)^{\beta_D}$$

A posteriori probability with Beta prior

α_P and β_P are the **hyperparameters** of the prior

$$\begin{aligned} P(D|\theta)P(\theta) &= \binom{\alpha_D + \beta_D}{\alpha_D} \theta^{\alpha_D} (1 - \theta)^{\beta_D} \cdot \text{Beta}(\theta; \alpha_P, \beta_P) = \binom{\alpha_D + \beta_D}{\alpha_D} \theta^{\alpha_D} (1 - \theta)^{\beta_D} \cdot \frac{\theta^{\alpha_P-1} (1 - \theta)^{\beta_P-1}}{\text{B}(\alpha_P, \beta_P)} \\ &= \binom{\alpha_D + \beta_D}{\alpha_D} \frac{\theta^{\alpha_D + \alpha_P - 1} (1 - \theta)^{\beta_D + \beta_P - 1}}{\text{B}(\alpha_P, \beta_P)} = \binom{\alpha_D + \beta_D}{\alpha_D} \frac{\text{B}(\alpha_D + \alpha_P, \beta_D + \beta_P)}{\text{B}(\alpha_P, \beta_P)} \text{Beta}(\theta; \alpha_D + \alpha_P, \beta_D + \beta_P) \end{aligned}$$

this factor is a positive constant (for θ)

Conjugate prior distributions

Coin tossing (i.e. Binomial)

α_D and β_D are the result counts (i.e. heads and tails)

$$P(D|\theta) = \binom{\alpha_D + \beta_D}{\alpha_D} \prod_i P(X_i|\theta) = \binom{\alpha_D + \beta_D}{\alpha_D} \theta^{\alpha_D} (1 - \theta)^{\beta_D}$$

A posteriori probability with Beta prior

$$P(D|\theta)P(\theta) = \binom{\alpha_D + \beta_D}{\alpha_D} \frac{B(\alpha_D + \alpha_P, \beta_D + \beta_P)}{B(\alpha_P, \beta_P)} \text{Beta}(\theta; \alpha_D + \alpha_P, \beta_D + \beta_P)$$

/ "is proportional to"

$$P(D|\theta)P(\theta) \propto \text{Beta}(\theta; \alpha_D + \alpha_P, \beta_D + \beta_P)$$

Optimization:

$$\theta_{MAP}^* = \operatorname{argmax}_{\theta} \text{Beta}(\theta; \alpha_D + \alpha_P, \beta_D + \beta_P) = \frac{\alpha_D + \alpha_P - 1}{\alpha_D + \alpha_P + \beta_D + \beta_P - 2}$$

which is the same as MLE but with the addition of $\alpha_P + \beta_P$ pseudo-observations

Being a **conjugate prior** $P(\theta)$ of a distribution $P(D|\theta)$ ^{in the above sense} means that the posterior $P(D|\theta)P(\theta)$ is in the same family of $P(\theta)$

Conjugate prior distributions

Coin tossing (i.e. a specific observation i)

$$P(D_i|\theta) = \theta^{[X_i=1]}(1 - \theta)^{[X_i=0]}$$

Likelihood (of a dataset)

$$P(D|\theta) = \binom{N}{N_{X=1}} \prod_i P(D_i|\theta) = \binom{N}{N_{X=1}} \theta^{N_{X=1}} (1 - \theta)^{N_{X=0}}$$

A posteriori probability with Beta prior

"is proportional to"

$$P(D|\theta)P(\theta) \propto \text{Beta}(\theta, N_{X=1} + \alpha_P, N_{X=0} + \beta_P)$$

Therefore

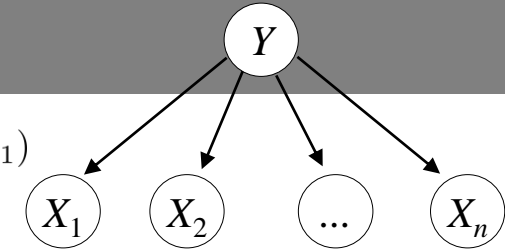
$$\theta_{MAP}^* = \operatorname{argmax}_{\theta} \text{Beta}(\theta, N_{X=1} + \alpha_P, N_{X=0} + \beta_P) = \frac{N_{X=1} + \alpha_P - 1}{N + \alpha_P + \beta_P - 2}$$

which is the same as MLE but with the addition of $\alpha_P + \beta_P$ *pseudo-observations*

Being a **conjugate prior** $P(\theta)$ of a distribution $P(D|\theta)$ *in the above sense* means that the posterior $P(D|\theta)P(\theta)$ is in *the same family* of $P(\theta)$

Anti-spam filter

$$P(Y, X_1, \dots, X_n) = P(Y) \prod_{i=1}^n P(X_i | X_{i-1})$$



■ Maximum a Posteriori (MAP) Estimation

The adapted computations for:

$$\theta_{MAP}^* := \operatorname{argmax}_{\theta} P(D|\theta) P(\theta)$$

yield:

$$\pi_k^* = \frac{\alpha_k + N_{Y=k} - 1}{\alpha_k + \beta_k + N - 2} \quad (\text{MAP Estimator of } \pi_k)$$

$$\pi_{ijk}^* = \frac{\alpha_{ijk} + N_{X_i=j, Y=k} - 1}{\alpha_{ijk} + \beta_{ijk} + N_{Y=k} - 2} \quad (\text{MAP Estimator of } \pi_{ijk})$$

where the

$$\alpha_k, \beta_k, \alpha_{ijk}, \beta_{ijk}$$

are the *hyperparameters* of the prior distribution representing the *pseudo-observations* made *before* the arrival of new, actual observations D

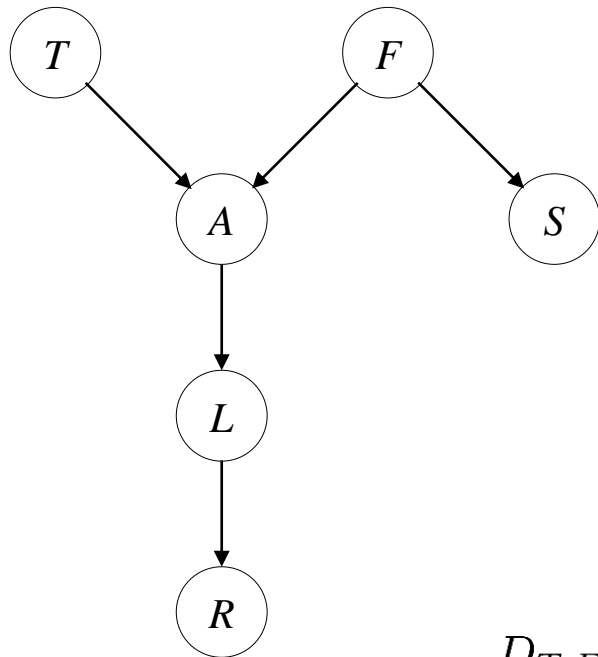
Bayesian Learning:
MAP for Graphical Models

Learning CPTs for a graphical model

As Maximum a Posteriori Estimation

More in general:

The MAP of a (directed) graphical model is the MAP of each node
(in each corresponding observation subset)



$$\theta_{MAP}^* := \operatorname{argmax}_{\theta} P(D | \theta) P(\theta)$$

$$\theta = \{\pi_T, \pi_F, \pi_{S|F}, \pi_{A|S,F}, \pi_{L|A}, \pi_{R|L}\}$$

$$\pi_T^* := \operatorname{argmax}_{\pi_T} P(D | \pi_T) P(\pi_T)$$

$$\pi_F^* := \operatorname{argmax}_{\pi_F} P(D | \pi_F) P(\pi_F)$$

$$\pi_{S|F}^* := \operatorname{argmax}_{\pi_{S|F}} P(D_F | \pi_{S|F}) P(\pi_{S|F})$$

$$\pi_{A|T,F}^* := \operatorname{argmax}_{\pi_{A|T,F}} P(D_{T,F} | \pi_{A|T,F}) P(\pi_{A|T,F})$$

$$\pi_{L|A}^* := \operatorname{argmax}_{\pi_{L|A}} P(D_A | \pi_{L|A}) P(\pi_{L|A})$$

$$\pi_{R|L}^* := \operatorname{argmax}_{\pi_{R|L}} P(D_L | \pi_{R|L}) P(\pi_{R|L})$$

$D_{T,F}$ denotes the subset of complete observation in which the random variables T, F have the corresponding value