PMR revision

probabilistic reasoning with *sum rule* & *product rule*.

Representation

What reasonably weak assumptions can we make to efficiently represent $p(\mathbf{x}, \mathbf{y}, \mathbf{z})$?

independence assumption & **parametric family assumption**

independence assumption:

Graph concepts: directed graph, DAG, parent, child, path / trail, directed path, ancestors, (non-) descendants

topological ordering $(x_1, \ldots, x_d)$: For all x_i, x_j connected by a directed edge $x_i \to x_j$, x_i should appear before x_j in the ordering.

- **directed graphical model (DGM)**: a distribution $p(x_1, \ldots, x_d)$ satisfies

  > - factorization property $F(G)$ (generative), if it factorizes over the graph G, $p(\mathbf{x}) = \Pi_{i=1}^{d} k(x_i|pa_i)$, where $k(x_i|pa_i)$ is called kernels / factors, factors equal to conditionals: $k(x_i|\mathrm{pa}_i) = p(x_i|\mathrm{pa}_i)$). (**ancestral sampling**)
  > - ordered (directed) Markov property, $M_0(G)$ (independence, filtering), if $\forall x_i \exists \pi_i$ s.t. $x_i \perp\!\!\!\perp (\mathrm{pre}_i \backslash \pi_i)|\pi_i$, i.e. $p(x_i|\mathrm{pa}_i) = p(x_i|\mathrm{pre}_i)$ where:
  >   - pre_i is the set of nodes before x_i in a topological ordering.
  >   - π_i is a minimal subsets of pre_i (parents of x_i in graphs).
  > - directed global Markov property, $M_g(G)$, if all independencies asserted by d-separation hold for $p(x_1, \ldots, x_d)$.
  > - directed local Markov property, $M_l(G)$, if $x_i \perp\!\!\!\perp (\mathrm{nondesc}(x_i) \backslash \mathrm{pa}_i)|\mathrm{pa}_i$ holds for all i, i.e. $p(x_i|\mathrm{nondesc}(x_i)) = p(x_i|\mathrm{pa}_i)$ for all i.

  - **independence**: $x \perp\!\!\!\perp y|z \Leftrightarrow p(x, y|z) = p(x|z)p(y|z) \quad \text{or} \quad p(x|y, z) = p(x|z)$.
  - factorization with chain rule, apply to any ordering: $p(\mathbf{x}) = \Pi_{i=1}^{d} p(x_i|\mathbf{pre}_i)$. given conditional independencies $x_i \perp\!\!\!\perp (\mathrm{pre}_i \backslash \pi_i)|\pi_i$, factorization simplifies as $p(\mathbf{x}) = \Pi_{i=1}^{d} p(x_i|\pi_i)$.
  - **D-separation**: $X \perp\!\!\!\perp Y|Z$ if every trail from $\forall x \in X$ to $\forall y \in Y$ is blocked by Z (serial or diverging: b in Z, converging: b and $desc(b)$ not in Z). Note that if not d-separated, then $X \not\perp\!\!\!\perp Y|Z$ in *some* probability distributions that factorize over the DAG.

    D-separation does generally not reveal all independencies in all probability distributions that factorize over the graph (graph criteria do not operate on the numerical values of the factor).
  - **Markov Blanket**: the minimal set of variables $\mathrm{MB}(x_i) = parents(x_i) \cup children(x_i) \cup \{parents(children(x_i)) \backslash x_i\}$ that makes x_i independent from all other variables. $x_i \perp\!\!\!\perp X \backslash \{x_i \cup \mathrm{MB}(x_i)\}|\mathrm{MB}(x_i)$.
- **undirected graphical models (UGM)**: all variables are associated with one node, each set of variables $\mathcal{X}_c$ for a factor ϕ_c are maximally connected with edges (forms a maximal *clique*). A distribution $p(x_1, \ldots, x_d)$ satisfies (assume $p(\mathbf{x} > 0)$ for all $\mathbf{x}$ in its domain, excludes deterministic relationships, gives equivalences of the four properties)

> - factorization property $F(H)$, if if it factorizes over the graph H, $p(\mathbf{x}) = \frac{1}{Z}\Pi_c\phi_c(\mathcal{X}_c)$, where $\phi_c(\mathcal{X}_c) \geq 0$, $\mathcal{X}_c$ corresponds to the **maximal cliques** in the graph.
> - global Markov property, $M_g(H)$, if all independencies asserted by graph separation hold for p.
> - local Markov property, $M_l(H)$, if $x \perp\!\!\!\perp X\backslash(x \cup \mathrm{ne}(x))|\mathrm{ne}(x)$ for $\forall x \in X$ holds for p.
> - pairwise Markov property, $M_p(H)$, if $x_i \perp\!\!\!\perp x_j|X\backslash(x_i, x_j)$, for $\forall x_i, x_j \in X$, s.t. $x_i \notin \mathrm{ne}(x_j)$ holds for p.

  - **independence**: $x \perp\!\!\!\perp y|z \Leftrightarrow p(x, y, z = \frac{1}{Z}\phi_A(x, z)\phi_B(y, z))$, $Z = \int_{x,y,z}\phi_A(x, z)\phi_B(y, z)dxdydz$ (ϕ called factors / potential functions).
  - **Gibbs distribution**: a class of pdfs / pmfs that factorize into factors of sets of variables, $p(x_1, \ldots, x_d) = \frac{1}{Z}\Pi_c\phi_c(\mathcal{X}_c)$, $\mathcal{X} \subseteq \{x_1, \ldots, x_d\}$. When $\phi_c(\mathcal{X}_c) = \exp\left(-E_c(\mathcal{X}_c)\right)$, forms energy-based model $p(x_1, \ldots, x_d) = \frac{1}{Z}\exp[-\Sigma_c E_c(\mathcal{X}_c)]$ (useful with log-space).
  - **Graph separation**: Two sets of variables X and Y are separated by Z, i.e. $X \perp\!\!\!\perp Y|Z$ if, after removing the Z-nodes, there is no path between any variable $x \in X$ and $y \in Y$. Note that separation criterion "not complete", similar to d-separation.
  - **Markov Blanket**: the minimal set of variables $\mathrm{MB}(x_i) = \mathrm{ne}(x_i)$ that makes x_i independent from all other variables. $x_i \perp\!\!\!\perp X\backslash\{x_i \cup \mathrm{MB}(x_i)\}|\mathrm{MB}(x_i)$.
- **Causal Inferences**:
  - **Structural Causal Model (SCM)**: an SCM M is given by the set of assignments $X_i := f_i(Pa_i, U_i)$, one for each variable in the network, where f_is are deterministic functions, and U_is are jointly independent noise variables.
  - **Intervention and the do-operator**: model with intervention $M' = M[X := x]$, eliminates all incoming edges into X. probabilities for an event E under the intervention is $p_{M[X:=x]}(E)$ or $p(E|do(X := x))$, fundamentally different from conditioning.
  - **confounding**: the difference between interventional statements and conditional statements.
  - **adjustment formula**: $p(Y = y|do(X := x)) = \Sigma_z p(Y + y|X = x, Pa = z)p(Pa = z)$, but we should not adjust to everything.
  - **propensity score**: $p(X = x|Pa = z)$; **inverse probability weighting**
  - **identifiable**: intervention distribution can be computed from the observational data & graph structure.
  - **randomized trials**: remove influences of any other variable on X, no hidden common cause.
  - **Counterfactual** $p(Y = y|E = e, do(X := x))$: general recipe
    1. Abduction: Condition the joint distribution of the exogenous variables $U = (U_1, \ldots, U_d)$ on the event $E = e$ to obtain $p(U|E = e)$.
    2. Action: Perform the do-intervention $X := x$ in M resulting in the model $M' = M[X := x]$ and the modified graph.
    3. Prediction: Compute the target counterfactual using the noise distribution $p(U|E = e)$ in M'.
- **I-map (Independency map)**: The sets of independencies that a graph K asserts is denoted $\mathcal{I}(K)$. K is said to be an I-map for a set of independencies $\mathcal{U}$ if $\mathcal{I}(K) \subseteq \mathcal{U}$.

  A complete graph is an I-map with no assertions made, not necessarily useful.

$\mathcal{U}$ can be specified as $\mathcal{U} = \{x_1 \perp\!\!\!\perp x_2\}$, $\mathcal{U} = \mathcal{K}_0$, or $\mathcal{U} = \mathcal{I}(p)$. Note that the incompleteness of d-separation can be expressed as $\mathcal{I}(K) \subseteq \mathcal{I}(p)$, where p is a factorization over K.

- **Minimal I-map** (stronger, not unique, different minI-maps not I-equivalent): sparsified I-map: a graph K such that if any edge is removed, $\mathcal{I}(K) \nsubseteq \mathcal{U}$. Constructions:
  - Undirected graphs: $\forall x_i \in N$, determine $\mathrm{MB}(x_i)$ and connect x_i to all variables in $\mathrm{MB}(x_i)$.
  - Directed graphs: Assume an ordering $x = (x_1, \ldots, x_d)$, then $\forall x_i \in \mathbf{x}$ set pa_i to π_i, where π_i is a minimal subset of the pre_i such that $x_i \perp\!\!\!\perp \{\mathrm{pre}_i \backslash \pi_i\} | \pi_i$.
- **P-map (Perfect I-map)** (even stronger, sometimes not unique / do not exists): K is a perfect I-map for $\mathcal{U}$ if $\mathcal{I}(K) = \mathcal{U}$.

  Note that related to the incompleteness of d-separation: $\mathcal{I}(K) = [\cap_{p \in \mathcal{P}_K} \mathcal{I}(p)]$.

  - Collider does not have an undirected P-map;
  - Diamond does not have a directed P-map.
- **I-equivalence**: whether two graphs make the same independence assertions
  - Directed graphs: $\mathcal{I}(G_1)$ and $\mathcal{I}(G_2)$ are I-equivalent *iff* they have the same skeleton and set of immoralities.
  - Undirected graphs: $\mathcal{I}(H_1)$ and $\mathcal{I}(H_2)$ are I-equivalent *iff* they have the same skeleton.
  - Undirected and directed graphs: $\mathcal{I}(H)$ and $\mathcal{I}(G)$ are I-equivalent *iff* they have the same skeleton and the DAG G does not have immoralities.

- **Factor graph**: a factor graph represents the factorization of an arbitrary function in terms of factors and their connections with variables. For example, a factor graph can represent a distribution written as a Gibbs distribution $p(\mathbf{x}) = \frac{1}{Z} \Pi_c \phi_c(\mathcal{X}_c)$ where
  - variables $x_i \in \mathbf{x}$ are represented with variable nodes (circles);
  - potentials ϕ_c are represented with factor nodes (squares).

  Edges connect each factor node ϕ_c to all its variable nodes $x_i \in \mathcal{X}_c$.

  Mapping from Gibbs distribution to factor graph is one-to-one, to undirected graph is many to one.

  conditionals: directed factor graphs; mixed: mixed graphs.

Exact inference

Can we further exploit the assumptions on $p(\mathbf{x}, \mathbf{y}, \mathbf{z})$ to efficiently compute the posterior probability or derived quantities?

- marginal inference: $p(\mathbf{x}|\mathbf{y}_o)$;
- inference of most probable states: $\arg\max_x p(\mathbf{x}|\mathbf{y}_o)$;
- posterior expectations: $\mathbb{E}[g(\mathbf{x})|\mathbf{y}_o]$.

Variable Elimination

Given $p(\mathbf{x}) \propto \Pi_c \phi_c(\mathcal{X}_c)$, we compute the marginal $p(\mathcal{X} \backslash x^*)$ via the sum rule by exploiting the factorization by means of the distributive law.

We sum out the variable x^* by first finding all factors $\phi_i(\mathcal{X}_i)$ such that $x^* \in \mathcal{X}_i$, and forming the compound factor $\phi^*(\mathcal{X}^*) = \Pi_{i:x^* \in \mathcal{X}_i} \phi_i(\mathcal{X}_i)$, with $\mathcal{X}^* = \cup_{i:x^* \in \mathcal{X}_i} \mathcal{X}_i$. Summing out x^* then produce a new factor $\tilde{\phi}^*(\tilde{\mathcal{X}}^*) = \Sigma_{x^*} \phi^*(\mathcal{X}^*)$ that does not depend on x^*, i.e. $\tilde{\mathcal{X}}^* = \mathcal{X}^* \backslash x^*$. This is possible as products are commutative, and a sum can be distributed within a product as long as all terms depending on the variable(s) being summed come to the right of the sum.

$$\begin{aligned} p(\mathcal{X}\backslash x^*) \propto \Sigma_{x^*}\Pi_c\phi_c(\mathcal{X}_c) &\propto [\Pi_{i:x^*\notin\mathcal{X}_i}\phi_i(\mathcal{X}_i)][\Sigma_{x^*}\Pi_{i:x^*\in\mathcal{X}_i}\phi_i(\mathcal{X}_i)] \\ &\propto [\Pi_{i:x^*\notin\mathcal{X}_i}\phi_i(\mathcal{X}_i)]\tilde{\phi}^*(\tilde{\mathcal{X}}^*) \end{aligned}$$

for conditionals, we need the new conditional factor graph on the non-evidential variables and apply the same process.

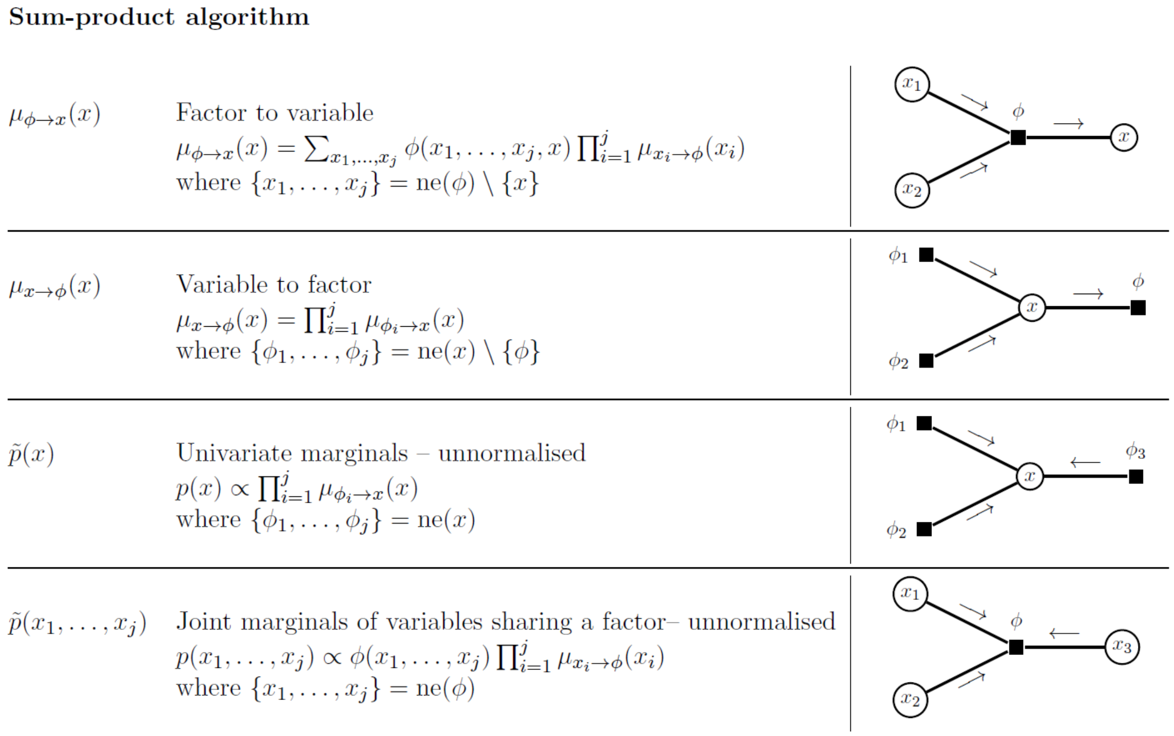
order: order of elimination matters. However, optimal choice of elimination order is difficult. Picking variables greedily is a common heuristic, where the "best" x^* is the one that fewest factors ϕ_c depend upon.

Sum-product algorithm

Variable elimination for factor trees reformulated with "messages" which allows for re-use of computations already done.

can only be used for factor trees, otherwise turn into trees with conditional / grouping variables / use variable elimination.

can compute **marginals** for a single variable / variables connected to the same factor node (neighbors).



Sum-product algorithm

$\mu_{\phi\to x}(x)$	Factor to variable $\mu_{\phi\to x}(x) = \sum_{x_1,\dots,x_j}\phi(x_1,\dots,x_j,x)\prod_{i=1}^{j}\mu_{x_i\to\phi}(x_i)$ where $\{x_1,\dots,x_j\} = \mathrm{ne}(\phi)\setminus\{x\}$	
$\mu_{x\to\phi}(x)$	Variable to factor $\mu_{x\to\phi}(x) = \prod_{i=1}^{j}\mu_{\phi_i\to x}(x)$ where $\{\phi_1,\dots,\phi_j\} = \mathrm{ne}(x)\setminus\{\phi\}$	
$\tilde{p}(x)$	Univariate marginals – unnormalised $p(x) \propto \prod_{i=1}^{j}\mu_{\phi_i\to x}(x)$ where $\{\phi_1,\dots,\phi_j\} = \mathrm{ne}(x)$	
$\tilde{p}(x_1,\dots,x_j)$	Joint marginals of variables sharing a factor– unnormalised $p(x_1,\dots,x_j) \propto \phi(x_1,\dots,x_j)\prod_{i=1}^{j}\mu_{x_i\to\phi}(x_i)$ where $\{x_1,\dots,x_j\} = \mathrm{ne}(\phi)$	

for conditionals, we need the new conditional factor graph on the non-evidential variables and apply the same process, or keep the original graph, but for factor to variable messagges, the sum only taken over non-evidential variables.

numerics: working in log domain with log-sum-exp trick for factor-to-variable messages. withdraw $\exp(\omega^*(x))$ outside to solve underflow issues.

cost: linear in the number of variables d, exponential in maximal number of variables attached to a factor node M, $O(2dK^M) = O(dK^M)$.

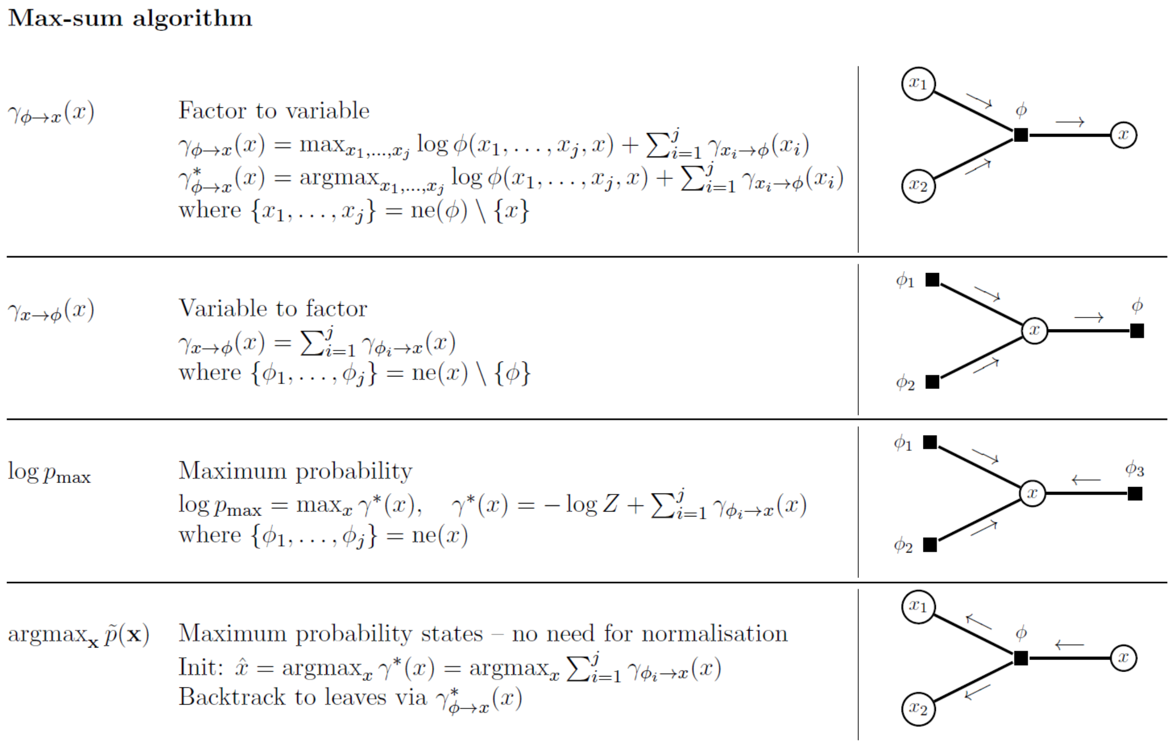
Max-product algorithm

Same as the sum-product algorithm, but $\max$ replaces Σ.

can compute **MAP** estimates.

Max-sum algorithm

Max-product algorithm in the log-domain.



Max-sum algorithm

$\gamma_{\phi\to x}(x)$	Factor to variable $\gamma_{\phi\to x}(x) = \max_{x_1,\ldots,x_j} \log\phi(x_1,\ldots,x_j,x) + \sum_{i=1}^{j}\gamma_{x_i\to\phi}(x_i)$ $\gamma^*_{\phi\to x}(x) = \mathrm{argmax}_{x_1,\ldots,x_j} \log\phi(x_1,\ldots,x_j,x) + \sum_{i=1}^{j}\gamma_{x_i\to\phi}(x_i)$ where $\{x_1,\ldots,x_j\} = \mathrm{ne}(\phi)\setminus\{x\}$
$\gamma_{x\to\phi}(x)$	Variable to factor $\gamma_{x\to\phi}(x) = \sum_{i=1}^{j}\gamma_{\phi_i\to x}(x)$ where $\{\phi_1,\ldots,\phi_j\} = \mathrm{ne}(x)\setminus\{\phi\}$
$\log p_{\max}$	Maximum probability $\log p_{\max} = \max_x \gamma^*(x), \quad \gamma^*(x) = -\log Z + \sum_{i=1}^{j}\gamma_{\phi_i\to x}(x)$ where $\{\phi_1,\ldots,\phi_j\} = \mathrm{ne}(x)$
$\mathrm{argmax}_{\mathbf{x}}\,\tilde{p}(\mathbf{x})$	Maximum probability states – no need for normalisation Init: $\hat{x} = \mathrm{argmax}_x \gamma^*(x) = \mathrm{argmax}_x \sum_{i=1}^{j}\gamma_{\phi_i\to x}(x)$ Backtrack to leaves via $\gamma^*_{\phi\to x}(x)$

Learning

How can we learn the numbers from data?

Likelihood $L(\theta)$: The chance that the model generates data like the observed one when using parameter configuration θ. For *iid* data $\mathcal{D} = \{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$, the likelihood of the parameters θ is $L(\theta) = p(\mathcal{D};\theta) = \Pi_{i=1}^{n} p(\mathbf{x}_i;\theta)$.

Prior $p(\theta)$: Beliefs about the plausibility of parameter values before we see any data.

Posterior $p(\theta|\mathcal{D})$: Beliefs about the parameters after having seen the data. This is proportional to the likelihood function $L(\theta)$ weighted by our prior beliefs about the parameters $p(\theta)$, $p(\theta|\mathcal{D}) \propto L(\theta)p(\theta)$.

Parametric statistical model: A set of pdfs / pmfs indexed by parameters θ, $\{p(\mathbf{x};\theta)\}_\theta$.

- **Parametric estimation**: Using $\mathcal{D}$ to pick the "best" parameter value $\hat{\theta}$ among the possible θ, i.e. pick the "best" pdf / pmf $p(\mathbf{x};\hat{\theta})$ from the set of pdfs / pmfs $\{p(\mathbf{x};\theta)\}_\theta$.
- **Partition function $Z(\theta)$ for unnormalized models**: $Z(\theta) = \int \tilde{p}(\mathbf{x};\theta)d\mathbf{x}$, $p(\mathbf{x};\theta) = \frac{\tilde{p}(\mathbf{x};\theta)}{Z(\theta)}$.

Bayesian model: Considers $p(\mathbf{x};\theta)$ to be conditional $p(\mathbf{x}|\theta)$. Models the distribution of the parameters θ, as well as the random variable x: $p(\mathbf{x},\theta) = p(\mathbf{x}|\theta)p(\theta)$.

- **Bayesian inference**: Determine the plausibility of all possible θ in light of the observed data, i.e. compute the posterior $p(\theta|\mathcal{D})$.
- **Predictive distribution**: $p(\mathbf{x}|\mathcal{D}) = \int p(\mathbf{x}|\theta)p(\theta|\mathcal{D})d\theta$.

Maximum likelihood: The parameters $\hat{\theta}$ that give the largest likelihood (or log-likelihood), $\hat{\theta} = \arg\max_\theta l(\theta) = \arg\max_\theta L(\theta)$. Sometimes this can be computed directly (as in the tutorials). However, numerical methods are often needed for this optimization problem, swhich leads to local optima.

- reparameterization
- **moment matching**: with $p(\mathbf{x};\theta) = \frac{\tilde{p}(\mathbf{x};\theta)}{Z(\theta)}$, then $\mathbb{E}_{p(\mathbf{x};\hat{\theta})}[\mathbf{m}(\mathbf{x};\hat{\theta})] = \frac{1}{n}\Sigma_{i=1}^n \mathbf{m}(\mathbf{x}_i;\hat{\theta})$, where moments $\mathbf{m}(\mathbf{x};\theta) = \nabla_\theta \log \tilde{p}(\mathbf{x};\theta)$. can *ignore partition function*!

Approximate inference and learning

Intractable likelihoods

1. due to **unobserved variables**: $l(\theta) = \log \int_{\mathbf{u}} p(\mathbf{u}, \mathcal{D}; \theta) d\mathbf{u}$.

   The likelihood is often maximized by gradient ascent, so we need the gradient, by solving an inference problem: $\nabla_\theta l(\theta) = \mathbb{E}_{\mathbf{u}\sim p(\mathbf{u}|\mathcal{D};\theta)}[\nabla_\theta \log p(\mathbf{u}, \mathcal{D}; \theta)]$ (a weighted average of gradients for filled-in data where the weight indicates the plausibility of the values that are used to fill-in the missing data)

2. due to **intractable partition functions**: $l(\theta) = \Sigma_{i=1}^n \log \tilde{p}(\mathbf{x}_i;\theta) - n \log Z(\theta)$, where $Z(\theta) = \int \tilde{p}(\mathbf{x};\theta) d\mathbf{x}$.

   The likelihood is often maximized by gradient ascent, so we need the gradient, by solving an inference problem: $\nabla_\theta l(\theta) = \Sigma_{i=1}^n \mathbf{m}(\mathbf{x}_i;\theta) - n\mathbb{E}_{p(\mathbf{x};\theta)}[\mathbf{m}(\mathbf{x};\theta)]$ (moment matching, turns to solving the moment expectation).

3. due to **both**: $l(\theta) = \log\left[\int_{\mathbf{u}} \tilde{p}(\mathbf{u}, \mathcal{D}; \theta) d\mathbf{u}\right] - \log\left[\int_{\mathbf{u},\mathbf{v}} \tilde{p}(\mathbf{u}, \mathbf{v}; \theta) d\mathbf{u} d\mathbf{v}\right]$.

   $\nabla_\theta l(\theta) = \mathbb{E}_{p(\mathbf{u}|\mathcal{D};\theta)}[\mathbf{m}(\mathbf{u}, \mathcal{D}; \theta)] - \mathbb{E}_{p(\mathbf{u},\mathbf{v};\theta)}[\mathbf{m}(\mathbf{u}, \mathbf{v}; \theta); \theta]$, where $\mathbf{m}(\mathbf{u}, \mathbf{v}; \theta) = \nabla_\theta \log \tilde{p}(\mathbf{u}, \mathbf{v}; \theta)$ (turns into computing expectations)

Variational Inference and Learning

Framing the problem into an *optimization* problem.

KL divergence: The Kullback-Leibler divergence measures the "distance" between p and q: $\mathrm{KL}(p||q) = \mathbb{E}_{p(\mathbf{x})}[\log \frac{p(\mathbf{x})}{q(\mathbf{x})}]$. It satisfies:

- $\mathrm{KL}(p||q) = 0 \Leftrightarrow p = q$;
- $\mathrm{KL}(p||q) \neq \mathrm{KL}(q||p)$;
- $\mathrm{KL}(p||q) \geq 0$.

Optimizing with respect to the first argument when the second is fixed leads to mode seeking (local fit). Optimizing with respect to the second argument when the first is fixed produces global fits (moment-matching, minimizing KL equivalent to MLE in iid case).

ELBO: For a joint model $p(\mathbf{x}, \mathbf{y})$, the evidence lower bound (ELBO) is $\mathcal{L}_{\mathbf{x}}(q) = \mathbb{E}_{q(\mathbf{y}|\mathbf{x})}[\log \frac{p(\mathbf{x},\mathbf{y})}{q(\mathbf{y}|\mathbf{x})}]$, where $q(\mathbf{y}|\mathbf{x})$ is the variational distribution. It can be rewritten as:

$$\begin{aligned}\mathcal{L}_{\mathbf{x}}(q) &= \log p(\mathbf{x}) - \mathrm{KL}(q(\mathbf{y}|\mathbf{x})||p(\mathbf{y}|\mathbf{x})) \\ &= \mathbb{E}_{q(\mathbf{y}|\mathbf{x})} \log p(\mathbf{x}|\mathbf{y}) - \mathrm{KL}(q(\mathbf{y}|\mathbf{x})||p(\mathbf{y})) \\ &= \mathbb{E}_{q(\mathbf{y}|\mathbf{x})} \log p(\mathbf{x}, \mathbf{y}) + \mathcal{H}(q)\end{aligned}$$

where $\mathcal{H}(q) = -\mathbb{E}_{q(\mathbf{y}|\mathbf{x})}[\log q(\mathbf{y}|\mathbf{x})]$ is the entropy of q. The ELBO is a lower bound on $\log p(\mathbf{x})$. It is maximized when $q(\mathbf{y}|\mathbf{x}) = p(\mathbf{y}|\mathbf{x})$ which makes the bound tight:

$$\log p(\mathbf{x}) = \max_q \mathcal{L}_{\mathbf{x}}(q)$$
$$p(\mathbf{y}|\mathbf{x}) = \arg\max_q \mathcal{L}_{\mathbf{x}}(q)$$

Solving the optimization problem:

- independence assumptions $q(\mathbf{y}|\mathbf{x}) = \Pi_i q(y_i|\mathbf{x})$.
- parametric assumptions e.g. $q(y_i|\mathbf{x}) = \mathcal{N}(y_i; \mu_i(\mathbf{x}), \sigma_i^2(\mathbf{x}))$.

Application to inference: In posterior inference tasks (given $\mathbf{x} = \mathbf{x}_o$ and $p(\mathbf{y}, \mathbf{x})$, compute $p(\mathbf{y}|\mathbf{x}_o)$): eq. 2 means posterior is a compromise between prior and fit; eq. 3 means posterior is a compromise between variable and likely imputations.

Application to learning:

- for Bayesian learning, a inference task as mentioned.
- learning in presence of unobserved variables: for model $p(\mathbf{v}, \mathbf{h}; \theta)$, estimate the parameters θ from data $\mathcal{D}$ on the visibles $\mathbf{v}$ only ($\mathbf{h}$ is unobserved): MLE = maximize the ELBO $\mathcal{L}_{\mathcal{D}}(\theta, q)$ w.r.t. θ and q.

EM algorithm: The expectation maximization (EM) algorithm can be used to learn the parameters θ of a statistical model $p(\mathbf{v}, \mathbf{h}; \theta)$ with latent (unobserved) variables $\mathbf{h}$ and visible (observed) variables $\mathbf{v}$ for which we have data $\mathcal{D}$, by maximizing $\mathcal{L}_{\mathcal{D}}(\theta, q)$. It updates the parameters θ by iterating between the expectation (E) and the maximization (M) step:

- **E-step**: compute $J(\theta) = \mathbb{E}_{p(\mathbf{h}|\mathcal{D};\theta_{\text{old}})}[\log p(\mathcal{D}, \mathbf{h}; \theta)]$
- **M-step**: $\theta_{\text{new}} \leftarrow \arg\max_\theta J(\theta)$.

The update rule produces a sequence of parameters for which the log-likelihood is guaranteed to never decrease, i.e. $l(\theta_{\text{new}}) \geq l(\theta_{\text{old}})$.

Scalable generic variational learning for latent variable models

since $q(\mathbf{h}_1, \ldots, \mathbf{h}_n|\mathbf{v}_1, \ldots, \mathbf{v}_n) = \Pi_{i=1}^n q(\mathbf{h}_i|\mathbf{v}_i)$ (iid data), ELBO $\mathcal{L}_{\mathcal{D}}$ becomes a sum of per data-point ELBOs $\mathcal{L}_i(\theta, q) = \mathbb{E}_{q(\mathbf{h}|\mathbf{v}_i)}[\log \frac{p(\mathbf{v}_i, \mathbf{h}; \theta)}{q(\mathbf{h}|\mathbf{v}_i)}] = \log p(\mathbf{v}_i; \theta) - \text{KL}(q(\mathbf{h}|\mathbf{v}_i)||p(\mathbf{h}|\mathbf{v}_i; \theta))$, $\mathcal{L}_{\mathcal{D}}(\theta, q) = l(\theta) - \Sigma_{i=1}^n \text{KL}(q(\mathbf{h}|\mathbf{v}_i)||p(\mathbf{h}|\mathbf{v}_i; \theta))$, $\max_\theta l(\theta) = \max_{\theta,q} \mathcal{L}_{\mathcal{D}}(\theta, q)$, MLE=maximization of ELBO.

two problem:

- learning the conditional variation distribution: amortisation;
- maximization: reparameterization.

VAE: Gaussianity assumption on $q_\phi(\mathbf{h}|\mathbf{v})$ and $p(\mathbf{h})$.

- variational distribution $q_\phi(\mathbf{h}|\mathbf{v})$: encoder. assumed factorized Gaussian, mapping $\mathbf{v}$ to $(\mu_1, \ldots, \mu_H, \sigma_1^2, \ldots, \sigma_H^2)$.
- model $p(\mathbf{v}|\mathbf{h}; \theta)$: decoder.

two independence assumptions:

- conditional independence assumption for $p(\mathbf{v}_i|\mathbf{h}; \theta)$;
- independence assumption for $q_\phi(\mathbf{h}|\mathbf{v})$.

Gaussian VAE is a nonlinear generalization of FA.

Approximate with sampling

Monte Carlo integration: We approximate an expectation via a sample average, $\mathbb{E}[g(\mathbf{x})] = \int g(\mathbf{x})p(\mathbf{x})d\mathbf{x} \approx \frac{1}{n}\Sigma_{i=1}^n g(\mathbf{x}_i), \quad \mathbf{x}_i \overset{iid}{\sim} p(\mathbf{x})$. In importance sampling, we approximate the expected value via $\mathbb{E}[g(\mathbf{x})] = \int g(\mathbf{x})\frac{p(\mathbf{x})}{q(\mathbf{x})}q(\mathbf{x})d\mathbf{x} \approx \frac{1}{n}\Sigma_{i=1}^n g(\mathbf{x}_i)\frac{p(\mathbf{x}_i)}{q(\mathbf{x}_i)}, \quad \mathbf{x}_i \overset{iid}{\sim} q(\mathbf{x})$, where $q(\mathbf{x})$ is the importance distribution. To avoid division by small values, $q(\mathbf{x})$ needs to be large when $g(\mathbf{x})p(\mathbf{x})$ is large.

Inverse transform sampling: Given we have a cdf $F_x(\alpha)$ which is invertible, we can generate samples $x^{(i)}$ from our distribution $p_x(x)$ using uniform samples $y^{(i)} \sim \mathcal{U}(0,1)$, $F_x(\alpha) = \mathbb{P}(x \leq \alpha) = \int_{-\infty}^{\alpha} p_x(y)dy$. Using the inverse cdf $F_x^{-1}(y)$, a sample $x^{(i)} \sim p_x(x)$ can be generated using $x^{(i)} = F_x^{-1}(y^{(i)}) \quad y^{(i)} \sim \mathcal{U}(0,1)$.

Rejection sampling: If we sample $\mathbf{x}_i \sim q(\mathbf{x})$ and only keep $\mathbf{x}_i$ with probability $f(\mathbf{x}_i) \in [0,1]$, the retained samples follow a pdf / pmf proportional to $q(\mathbf{x})f(\mathbf{x})$. The normalising constant equals the acceptance probability $\int q(\mathbf{x})f(\mathbf{x})d\mathbf{x}$. The samples follow $p(\mathbf{x})$ if $f(\mathbf{x})$ is chosen as: $f(\mathbf{x}) = \frac{1}{M}\frac{p(\mathbf{x})}{q(\mathbf{x})} \quad M = \max_{\mathbf{x}} \frac{p(\mathbf{x})}{q(\mathbf{x})}$.

The acceptance probability then equals $\frac{1}{M}$.

Gibbs sampling: Given a multivariate pdf $p(\mathbf{x})$ and an initial state $\mathbf{x}^{(1)} = (x_1^{(1)}, \ldots, x_d^{(1)})$, we obtain multivariate samples $\mathbf{x}^{(k)}$ by sampling from a univariate distribution $p(x_i|\mathbf{x}_{\backslash i})$, and updating individual variables many times.

$$\begin{aligned}
\mathbf{x}^{(2)} &= (x_1^{(1)}, \ldots, x_{i-1}^{(1)}, x_i^{(2)}, x_{i+1}^{(1)}, \ldots, x_d^{(1)}) \quad i \sim \{0, \ldots, d\} \\
&\vdots \\
\mathbf{x}^{(n)} &= (x_1^{(n-1)}, \ldots, x_{j-1}^{(n-1)}, x_j^{(n)}, x_{j+1}^{(n-1)}, \ldots, x_d^{(n-1)}) \quad j \sim \{0, \ldots, d\}
\end{aligned}$$

In the multidimensional space of $\mathbf{x}$, the iterative Gibbs sampling process will appear as a path in orthogonal axes. Like other MCMC methods, Gibbs sampling typically exhibits a warm-up period, where the samples are not representative of the distribution $p(\mathbf{x})$ and the samples are not independent from each other. For multi-modal distributions Gibbs sampling may fail to sample from one or more modes, especially if the modes do not overlap when projected onto any of axes.

Markov models

Markov chains

L-th order Markov chain: A distribution factorized such that each variable x_i depends on L previous (contiguous) nodes $\{x_{i-L}, \ldots, x_{i-1}\}$, $p(x_1, \ldots, x_d) = \Pi_{i=1}^d p(x_i|x_{i-L}, \ldots, x_{i-1})$.

1-st order Markov chain: $p(x_1, \ldots, x_d) = \Pi_{i=1}^d p(x_i|x_{i-1})$.

The **transition distribution** $p(x_i|x_{i-1})$ (transition matrix $\mathbf{A}^{(i)}$) gives the probability of transitioning to different states. However, if this does not depend on i, then the Markov chain is said to be **homogeneous**.

Hidden Markov Model (HMM)

A 1st-order Markov chain on latent variables h_i (hiddens), with an additional set of visible variables v_i that represent observations. An emission distribution $p(v_i|h_i)$ gives the probabilities of the observations v_i (visibles) taking different values, if the observations are real-valued then $p(v_i|h_i)$ will be a probability density function.

$$p(h_{1:d}, v_{1:d}) = p(v_1|h_1)p(h_1)\Pi_{i=2}^d p(v_i|h_i)p(h_i|h_{i-1})$$

An HMM is said to be **stationary** if its transition and emission distributions don't depend on i. Turns into a *Linear Dynamic System (LDS)* if transition & emission model is linear.

Inference

1. **Filtering**, $p(h_t|v_{1:t})$: **alpha-recursion**, forward. Conditioning reduces the factor graph to a chain.

   A recursive process that propagates information forwards, from h_s to h_{s-1}:

   $$\begin{aligned} p(h_s|v_{1:s}) \propto \alpha(h_s) &= p(v_s|h_s)\Sigma_{h_{s-1}}p(h_s|h_{s-1})\alpha(h_{s-1}) = p(h_s, v_{1:s}) \\ p(h_1|v_1) \propto \alpha(h_1) &= p(h_1)p(v_1|h_1) = p(h_1, v_1) \end{aligned}$$

   Note that also, $p(v_{1:s}) = \Sigma_{h_s}\alpha(h_s) = Z$, likelihood is the normalizer.

   Intuition of why $\alpha(h_s) = p(h_s, v_{1:s})$ instead of $\alpha(h_s) = p(h_s|v_{1:s})$: based on the original factor graph (without conditioning),

   $$\begin{aligned} \alpha(h_4) &= \mu_{\phi_4 \to h_4} \\ &= \Sigma_{h_3}p(h_4|h_3)p(v_4|h_4)\Sigma_{h_2}p(h_3|h_2)p(v_3|h_3)\Sigma_{h_1}p(h_2|h_1)p(v_2|h_2)p(h_1)p(v_1|h_1) \\ &= \Sigma_{h_{1:3}}p(h_{1:4}, v_{1:4}) \\ &= p(h_4, v_{1:4}) \end{aligned}$$

   not the same as $p(h_4|v_{1:4})$!

2. **Smoothing**, $p(h_t|v_{1:u}), t < u$: **beta-recursion**, backward.

   A recursive process that propagates information backwards, from h_s to h_{s-1}:

   $$\begin{aligned} \beta(h_{s-1}) &= \Sigma_{h_s}p(v_s|h_s)p(h_s|h_{s-1})\beta(h_s) = p(v_{s:u}|h_{s-1}) \\ \beta(h_u) &= 1 \end{aligned}$$

   $p(h_t|v_{1:u}) = \frac{1}{Z_t^u}\alpha(h_t)\beta(h_t)$, $Z_t^u = \Sigma_{h_t}\alpha(h_t)\beta(h_t)$.

3. **Prediction**, **Posterior sampling**, **Most likely hidden path**, ...

FA & ICA

1. **Factor Analysis**: A graphical model where statistical dependencies between the observed variables (visibles $\mathbf{v}$) is modelled through unobserved variables (latents $\mathbf{h}$). In factor analysis, the latents $\mathbf{h}$ are assumed to be independent Gaussians with zero mean and unit variance.

   $$\begin{gathered} p(\mathbf{h}) = \mathcal{N}(\mathbf{h}; \mathbf{0}, \mathbf{I}) \\ p(\mathbf{v}|\mathbf{h}; \theta) = \mathcal{N}(\mathbf{v}; \mathbf{Fh} + \mathbf{c}, \Psi) \\ \mathbf{v} = \mathbf{Fh} + \mathbf{c} + \epsilon = \Sigma_i^H \mathbf{f}_i h_i + \mathbf{c} + \epsilon \quad \epsilon \sim \mathcal{N}(\epsilon; 0, \Psi) \end{gathered}$$

   The covariance matrix Ψ is a diagonal matrix.

   - H **vs** D: number of latents H assumed smaller than number of visibles V;
   - **distribution of v & likelihood**: $\mathbf{v} = \mathcal{N}(\mathbf{v}; \mathbf{c}, \mathbf{FF}^\top + \Psi)$, likelihood given by multivariate Gaussian;
   - $\mathbf{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_H\}$, $\mathbf{f}_i$ called factors;
   - **ambiguity**: with orthogonal matrix $\mathbf{R}$, $\mathbf{v} = (\mathbf{FR})(\mathbf{R}^\top\mathbf{h}) + \mathbf{c} + \epsilon = (\mathbf{FR})\tilde{\mathbf{h}} + \mathbf{c} + \epsilon$, $p(\tilde{\mathbf{h}}) = \mathcal{N}(\tilde{\mathbf{h}}; \mathbf{0}, \mathbf{I})$, estimation of the factor matrix $\mathbf{F}$ is not unique (rotational ambiguity). FA jsust define a subspace of the data space;
   - Probabilistic PCA is a special case of factor analysis, where $\Psi = \sigma^2\mathbf{I}$.

2. **Independent Component Analysis**: The DAG is the same as in factor analysis, but with non-Gaussian latents (one latent may be Gaussian)

$$p(\mathbf{h}) = \Pi_i p(h_i)$$
$$p(\mathbf{v}|\mathbf{h};\theta) = \mathcal{N}(\mathbf{v}; \mathbf{A}\mathbf{h} + \mathbf{c}, \Psi)$$

   - H **vs** D: number of latents H can be larger or smaller than number of visibles V;
   - **distribution of** v **& likelihood**: $p(\mathbf{v}; \mathbf{A}) = [\Pi_{j=1}^{D} p_h(\mathbf{b}_j \mathbf{v})] |\det \mathbf{B}|$, where $\mathbf{B} = \mathbf{A}^{-1}$, $\mathbf{b}_i$ is its i-th row; thus likelihood also given.
   - **ambiguity**: Consider no noise: $\mathbf{v} = \mathbf{A}\mathbf{h} = \Sigma_{i=1}^{D} \mathbf{a}_i h_i = \Sigma_{k=1}^{D} \mathbf{a}_{i_k} h_{i_k} = \Sigma_{i=1}^{D} (\mathbf{a}_i \alpha_i) \frac{1}{\alpha_i} h_i$, ambiguity regarding *ordering of columns of* $\mathbf{A}$ and *scaling*. (if h_i typically assumed as zero mean & unit variance, no scaling ambiguity)
   - Sub- / Super- Gaussian pdf

Decision Theory:

Loss and Risk: An agent has a set of possible actions $\mathcal{A}$ to choose from. Each action has costs / benefits, which depend on the underlying state of nature $h \in \mathcal{H}$. The **loss** function $l(h, a)$ specifies the loss incurred when taking action a when the state of nature is h. Utility is basically same thing as loss, but with the opposite sign, $U(h, a) = -l(h, a)$.

Given observations $\mathbf{x}$, we obtain $p(h|\mathbf{x})$. The **risk** associated with action a is given by $R(a|\mathbf{x}) = \Sigma_h l(h, a) p(h|\mathbf{x})$.

The optimal policy is to choose the action associated with the lowest risk, i.e. $\pi^*(\mathbf{x}) = \arg\min_a R(a|\mathbf{x})$.

L2 loss: optimal action is posterior mean; L1 loss: optimal action is posterior median.

Other basics

conditional gaussian: $\mu_{1|2}^c = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{x}_2 - \mu_2)$, $\Sigma_{1|2}^c = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$, for $\mu = (\mu_1, \mu_2)^\top$, $\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$.