CS5340 Uncertainty Modelling in AI

AY2022/23 Semester 2 · Prepared by Tian Xiao *@snoidetx*

CS5340 is about how to ***represent*** *and* ***reason*** *with* ***uncertainty*** *in a computer.*

1 Probability Basics

Probability Space: A *probability space* (Ω, E, P) models a process consisting of outcomes that occur randomly. It consists of three parts:

- Outcome or sample space, Ω (e.g. $\{1,2,3,4,5,6\}$);
- Event space, $E \subseteq 2^\Omega$ (e.g. $\{\emptyset, \{1,3,5\}, \{2,4,6\}, \Omega\}$);
  - ▶ Event space must contain $\emptyset$ and Ω.
  - ▶ Event space is closed under union ($\alpha, \beta \in E \rightarrow \alpha \cup \beta \in E$).
  - ▶ Event space is closed under complement ($\alpha \in E \rightarrow \Omega - \alpha \in E$).
- Probability function, $P : E \rightarrow [0,1]$.

Probability Distribution: A *probability distribution* P over (Ω, E) is a mapping from events in E to real values that satisfies the following:

- Non-negativity: $\forall \alpha \in E\,[P(\alpha) > 0]$.
- Probability of all outcomes sum to 1, i.e. $P(\Omega) = 1$.
- Mutually disjoint events: $\alpha \cap \beta = \emptyset \Rightarrow P(\alpha \cup \beta) = P(\alpha) + P(\beta)$.

Random Variable: A *random variable*, $X : \Omega \rightarrow S$, is a function that maps a set of possible outcomes Ω to a space S.

- *Indicator random variable* maps every outcome to either 0 or 1.
- The set of values that X can take is denoted as $\text{Val}(X)$.
- A lower-case letter x is a generic value/realisation of X.
- $p(x)$ denotes $P(X = x)$. x^i denotes a specific value of X.
- For any *discrete* probability distribution, $\sum_{i=1}^{K} p(x^i) = 1$.
- For any *continuous* probability distribution, $\int_{-\infty}^{\infty} p(x^i) = 1$.

Joint Probability: $p(x, y) = P(X = x \text{ and } Y = y)$.

- Sum rule: $p(x) = \begin{cases} \sum_y p(x,y), & \text{when } Y \text{ is discrete;} \\ \int p(x,y)\,dy, & \text{when } Y \text{ is continuous.} \end{cases}$
  - ▶ Sum rule is also known as *marginalization*.
- Product rule: $p(x, y) = p(x|y)p(y)$.
  - ▶ Product rule is also known as *chain rule*.

Conditional Probability: $p(x|y^*)$ denotes the probability of $X = x$ given $Y = y^*$.

- $p(x|y) = \frac{p(x,y)}{p(y)} = \frac{p(x,y)}{\int p(x,y)\,dx}$.

Bayes Rule:

$$p(y|x) = \frac{p(x|y)p(y)}{\int p(x|y)p(y)\,dy}$$

Likelihood: Propensity for observing $X = x$ given $Y = y$. Prior: What we know about Y before seeing X. Posterior: What we know about Y after seeing X. Evidence

Independence: X and Y are *independent* if every conditional probability distribution is the same (i.e. $p(x|y) = p(x)$; $p(y|x) = p(y)$).

- If X and Y are independent, $p(x, y) = p(x)p(y)$.

Expectation: *Expectation* is the expected or average value of some function $f(x)$ taking into account the distribution of X.

- $\mathbb{E}[f(x)] = \begin{cases} \sum_x f(x)p(x), & \text{when } X \text{ is discrete;} \\ \int f(x)p(x)\,dx, & \text{when } X \text{ is continuous.} \end{cases}$
- $\mathbb{E}(c) = c$.
- $\mathbb{E}(cf(x)) = c\,\mathbb{E}(f(x))$.
- $\mathbb{E}(f(x) + g(x)) = \mathbb{E}(f(x)) + \mathbb{E}(g(x))$.
- $\mathbb{E}(f(x)g(y)) = \mathbb{E}(f(x))\mathbb{E}(g(y))$, if X and Y are independent.

Distribution	Parameter(s)	Domain	Probability Density/Mass Function
Bernoulli	$\lambda \in [0,1]$	$x \in \{0,1\}$ (binary)	$p(x) = \text{Bern}_x[\lambda] = \lambda^x(1-\lambda)^{1-x}$
Binomial	$n > 0$ and $\lambda \in [0,1]$	$x \in \{0, \cdots, n\}$ (discrete)	$p(x) = \text{Bin}_x[n,\lambda] = \binom{n}{x}\lambda^x(1-\lambda)^{n-x}$
Categorial	$\boldsymbol{\lambda} = [\lambda_1, \cdots, \lambda_K]^\top$ $\lambda_k \geq 0,\ \sum_{k=1}^{K} \lambda_k = 1$	K-dimensional one-hot vector	$p(\mathbf{x}) = \text{Cat}_{\mathbf{x}}[\boldsymbol{\lambda}] = \lambda_k$
Univariate Normal (Gaussian)	$\mu \in \mathbb{R}$ (mean) $\sigma^2 > 0$ (variance)	$x \in \mathbb{R}$ (continuous)	$p(x) = \text{Norm}_x[\mu, \sigma^2] = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
Multivariable Normal	$\boldsymbol{\mu} \in \mathbb{R}^D$ (mean) $\boldsymbol{\Sigma} \in \mathbb{R}^+_{D\times D}$ (cov)	$\mathbf{x} \in \mathbb{R}^D$ (continuous)	$p(\mathbf{x}) = \text{Norm}_{\mathbf{x}}[\boldsymbol{\mu}, \boldsymbol{\Sigma}] = \frac{1}{(2\pi)^{\frac{D}{2}}\lvert\boldsymbol{\Sigma}\rvert^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$

Conjugate Distribution: *Conjugate distributions* are used to model the parameters of probability distributions.

- Product of a probability distribution and its conjugate has the same form as the conjugate times a constant.
- Parameters of conjugate distributions are called *hyperparameters*.

Distribution	Conjugate	Hyperparameter(s)	Probability Density/Mass Function of Conjugate
Bernoulli Binomial	Beta	$\alpha, \beta > 0$	$p(\lambda) = \text{Beta}_\lambda[\alpha,\beta] = \frac{1}{B(\alpha,\beta)}\lambda^{\alpha-1}(1-\lambda)^{\beta-1}$ where $B(\alpha,\beta) = \frac{\Gamma[\alpha]\Gamma[\beta]}{\Gamma[\alpha+\beta]} = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\,dt$
Categorial	Dirichlet	$\alpha_1, \cdots, \alpha_k > 0$	$p(\boldsymbol{\lambda}) = \text{Dir}_{\boldsymbol{\lambda}}[\boldsymbol{\alpha}] = \frac{\Gamma\left[\sum_{k=1}^{K}\alpha_k\right]}{\prod_{k=1}^{K}\Gamma[\alpha_k]}\prod_{k=1}^{K}\lambda_k^{\alpha_k-1}$
Univariate Normal (Gaussian)	Normal Inverse Gamma	$\alpha, \beta, \gamma > 0$ $\delta \in \mathbb{R}$	$p(\mu,\sigma^2) = \text{NormInvGam}_{\mu,\sigma^2}[\alpha,\beta,\gamma,\delta] = \frac{\sqrt{\gamma}}{\sigma\sqrt{2\pi}}\frac{\beta^\alpha}{\Gamma[\alpha]}\left(\frac{1}{\sigma^2}\right)^{\alpha+1} e^{-\frac{2\beta+\gamma(\delta-\mu)^2}{2\sigma^2}}$
Multivariable Normal	Normal Inverse Wishart	$\alpha, \gamma > 0$ $\boldsymbol{\delta} \in \mathbb{R}^D$ +ve definite $\Psi \in \mathbb{R}^{D\times D}_+$	$p(\boldsymbol{\mu},\boldsymbol{\Sigma}) = \text{NorIWis}_{\boldsymbol{\mu},\boldsymbol{\Sigma}}[\alpha,\boldsymbol{\Phi},\gamma,\boldsymbol{\delta}] = \frac{\gamma^{\frac{D}{2}}\lvert\boldsymbol{\Psi}\rvert^{\frac{\alpha}{2}} e^{-\frac{1}{2}\left(\text{Tr}[\boldsymbol{\Phi}\boldsymbol{\Sigma}^{-1}]+\gamma(\boldsymbol{\mu}-\boldsymbol{\delta})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{\delta})\right)}}{2^{\frac{\alpha D}{2}}(2\pi)^{\frac{D}{2}}\lvert\boldsymbol{\Sigma}\rvert^{\frac{\alpha+D+2}{2}}\Gamma_D\left[\frac{\alpha}{2}\right]}$

2 Simple Probabilistic Models

Goal: To learn the unknown parameter(s) θ from a set of given data $\mathcal{D} = \{x[1], \cdots, x[N]\}$, and use those parameter(s) to make predictions.

Maximum Likelihood Estimate (MLE): $\hat{\theta}_{\text{MLE}} = \arg\max_{\theta\in\Theta}[p(\mathcal{D}|\theta)]$.

- i.i.d. assumption: $p(\mathcal{D}|\theta) = \prod_{i=1}^{N} p(x[i]|\theta)$.
- Log likelihood: $\hat{\theta}_{\text{MLE}} = \arg\max_{\theta\in\Theta}\left[\sum_{i=1}^{N}\log[p(x[i]|\theta)]\right]$.
- Maximizer: Partial derivative equates to 0.
- Pros:
  - ▶ MLE is easy and fast to compute.
  - ▶ MLE is consistent, i.e. $\hat{\theta}_{\text{MLE}} \rightarrow \theta^*$ as $N \rightarrow \infty$.
  - ▶ MLE is efficient, i.e. there is no consistent estimator that has lower MSE than $\hat{\theta}_{\text{MLE}}$.
  - ▶ MLE us functionally invariant, i.e. MLE for $g(\theta^*)$ is $g(\hat{\theta}_{\text{MLE}})$.
- Cons:
  - ▶ MLE is a point estimate which does not represent uncertainty.
  - ▶ MLE may overfit.
  - ▶ MLE does not incorporate prior information.
  - ▶ Asymptotic results are for the limit and assume model is correct.
  - ▶ MLE may not exist or may not be unique.
- Prediction for new data point x^*: Evaluate $p(x^*|\hat{\theta}_{\text{MLE}})$.

Bayesian Inference: $p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{\int p(\mathcal{D}|\theta)p(\theta)d\theta} = \frac{\prod_{i=1}^{N} p(x[i]|\theta)p(\theta)}{\int \prod_{i=1}^{N} p(x[i]|\theta)p(\theta)d\theta}$.

- Bayesian inference computes posterior distribution over all possible parameter values, hence it models uncertainty over parameters.
- Pros:
  - ▶ Bayesian inference incorporates prior information.
  - ▶ We can derive quantities of interest from the result.
  - ▶ Bayesian inference allows us to perform model selection.
- Cons:
  - ▶ Prior belief may not be conjugate to likelihood, hence it is computationally intractable.
- Prediction for x^*: Calculate $p(x^*|\mathcal{D}) = \int p(x^*|\theta)p(\theta|\mathcal{D})\,d\theta$.

Maximum a Posteriori Estimate (MAP): $\hat{\theta}_{\text{MAP}} = \arg\max_{\theta \in \Theta}[p(\theta|\mathcal{D})]$.

- More data points → MAP closer to MLE.
- Fewer data points → MAP closer to prior.
- Pros:
  - ▶ MAP is easy and fast to compute.
  - ▶ MAP incorporates prior information.
  - ▶ MAP avoids overfitting.
  - ▶ As $n \to \infty$, MAP approaches MLE but does not have similar asymptotic properties (consistency & efficiency).
- Cons:
  - ▶ MAP is also a point estimate like MLE.
  - ▶ We are still forced to choose prior.
  - ▶ MAP is not functionally invariant.
- Prediction for new data point x^*: Evaluate $p(x^*|\hat{\theta}_{\text{MAP}})$.

Exponential Family (ExpFam): An *exponential family* is a set of probabilistic distributions $\{p_\theta : \theta \in \Theta\}$ with the form

$$p_\theta(x) = \frac{h(x)e^{\eta(\theta)^\top s(x)}}{Z(\theta)},$$

where

- $\theta \in \Theta \subseteq \mathbb{R}^k$, $x \in \mathbb{R}^d$;
- Natural parameters, $\eta(\theta) : \Theta \to \mathbb{R}^m$;
- Sufficient statistics, $s(x) : \mathbb{R}^d \to \mathbb{R}^m$;
- Base measure, $h(x) : \mathbb{R}^d \to [0, \infty)$;
- Partition function, $Z(\theta) : \Theta \to [0, \infty)$.
- An exponential family is in its *natural/canonical* form if it is parametrized by its natural parameters:

$$p_\eta(x) = \frac{h(x)e^{\eta^\top s(x)}}{Z(\eta)},$$

where $Z(\eta) = \int h(x)e^{\eta^\top s(x)}\,dx$ is called *normalizer*.
- Log partition function: $p_\eta(x) = h(x)e^{\left[\eta^\top s(x) - A(\eta)\right]}$.
Here $A(\eta)$ is the log of partition function, i.e. $A(\eta) = \log Z(\eta)$.
  - ▶ $\mathbb{E}(s(x)) = \nabla \log Z(\eta) = \nabla A(\eta)$.
  - ▶ If $s(x) = x$, we can find moments of x by differentiation.
- $\nabla A(\eta_{\text{MLE}}) = \frac{1}{N}\sum_{n=1}^{N} s(x_n)$. The MLE only depends on $s(x)$.

3 Bayesian Networks

Conditional Independence: Two random variables X_A and X_C are *conditionally independent* given X_B (i.e. $X_A \perp X_C \mid X_B$) if and only if $p(x_A, x_C|x_B) = p(x_A|x_B)p(x_C|x_B)$, or alternatively $p(x_A|x_B, x_C) = p(x_A|x_B)$.

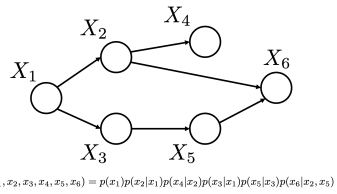
- Consider $p(x|\theta) = p(x_1|\theta)p(x_2|x_1, \theta_2)p(x_3|x_1, \theta_3)p(x_4|x_2, x_3, \theta_4)$:

$$\begin{aligned}
\text{MLE: } \arg\max_{\theta_1} p(x|\theta) &= \arg\max_{\theta_1} \log p(x|\theta) \\
&= \arg\max_{\theta_1} \{\log p(x_1|\theta_1) + \cancel{\log p(x_2|x_1, \theta_2)} + \\
&\qquad \cancel{\log p(x_3|x_1, \theta_3)} + \cancel{\log p(x_4|x_2, x_3, \theta_4)}\} \\
&= \arg\max_{\theta_1} \log p(x_1|\theta_1)
\end{aligned}$$

$$\begin{aligned}
\text{MAP: } \arg\max_{\theta_1} p(\theta|x) &= \arg\max_{\theta_1} \log p(\theta|x) \\
&= \arg\max_{\theta_1} \log p(x|\theta)p(\theta) \\
&= \arg\max_{\theta_1} \log p(x|\theta_1) + \log p(\theta_1)
\end{aligned}$$

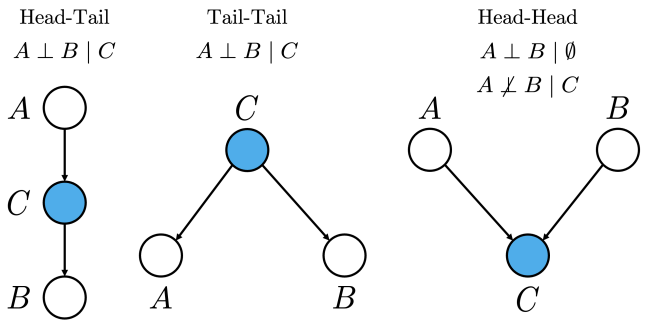
Bayesian Networks: A *Bayesian network* is a tuple $B = (G, P)$ where P factorizes according to G and where P is specified as a set of conditional probability distributions associated with G's nodes.

- A Bayesian network is a DAG.
  - ▶ Each node is associated with a random variable X_i.
  - ▶ Shaded node refers to an observed variable.
  - ▶ Topological ordering: $(X_i \to X_j) \Rightarrow (i < j)$ (not unique).
  - ▶ Path: A walk following the direction of →.
  - ▶ Trail: A walk following the direction, or anti-direction, of →.
- Local Markov assumption: Each random variable X_i is independent of its non-descendants $X_{\text{nonDesc}(X_i)}$ given its parents X_{π_i}.
  - ▶ Locality of the parent-child relationship is used to construct economical representations of the joint distribution.
  - ▶ The parent-child (X_{π_i}, X_i) represents the conditional independence $p(x_i|x_{\pi_i})$.
- Joint probability: $p(x_1, \cdots, x_N) = \prod_{i=1}^{N} p(x_i|x_{\pi_i})$.



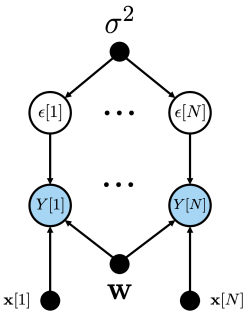
- Independence set (I-Set): Let P be a distribution over $\mathcal{X}$. Define $\mathcal{J}(P)$ as the set of independence assertions of the form $(X \perp Y \mid Z)$ that hold in P.
- Independence map (I-Map): Let G be associated with independence assertions $\mathcal{J}(G)$. G is an *independence map* for P if $\mathcal{J}(G) \subseteq \mathcal{J}(P)$.
- Pros:
  - ▶ Reduces the number of parameters needed to model the joint distribution.
  - ▶ Visualizes the structure of the probablistic model.
  - ▶ Provides insights into the conditional independence properties.
  - ▶ Expresses complex calculations as graphical manipulations.
- Misconceptions:
  - ✖ The arrows always indicate dependence.
  - ✖ Every network represents a unique probability distribution.
  - ✖ Observations always result in independence between random variables.

Graph Separation: A set of nodes A is said to be *d-separated* from B by C if all trails from nodes in set A are "blocked" from nodes in set B when all nodes from set C are observed, such that $A \perp B \mid C$.



Linear Regression: $Y[i] = \mathbf{w}^\top \mathbf{x}[i] + \epsilon[i]$, where

- $\mathbf{x}[i]$ is a D-dimensional observed input vector;
- $\mathbf{w}$ is a coefficient vector;
- $\epsilon[i] \sim N(0, \sigma^2)$ is iid zero-mean Gaussian noise.



- Circles are random variables.
- Shaded circles are observed random variables.
- Unshaded circles are unobserved/latent/hidden.
- Filled circles are deterministic parameters.

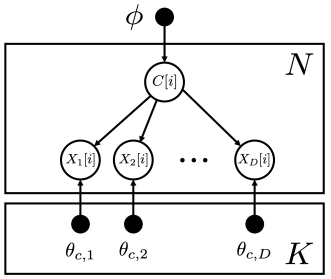
The independence assertions here are $Y[i] \perp Y[i+1] \mid \mathbf{x}[i], \mathbf{w}, \sigma_n^2$, hence we have the factorization $p(y[1], \cdots, y[N]) = \prod_{i=1}^{N} p(y[i]|\mathbf{w}^\top \mathbf{x}[i], \sigma_n^2)$. Assume we know σ_n^2, we want to learn $\mathbf{w}$.

$$\begin{aligned}
\mathbf{w}_{\text{MLE}} &= \arg\max_{\mathbf{w}} \log p(\mathcal{D}|\theta) \\
&= \arg\max_{\mathbf{w}} \log \prod_{i=1}^{N} p\left(y[i] \middle| \mathbf{w}^\top \mathbf{x}[i], \sigma_n^2\right) \\
&= \arg\max_{\mathbf{w}} \sum_{i=1}^{N} \log \left[N\left(y[i] \middle| \mathbf{w}^\top \mathbf{x}[i], \sigma_n^2\right)\right] \\
&= \arg\max_{\mathbf{w}} -\sum_{i=1}^{N} \frac{\left(y[i] - \mathbf{w}^\top \mathbf{x}[i]\right)^2}{2\sigma_n^2} \\
&= \arg\min_{\mathbf{w}} \underbrace{\frac{1}{2} \sum_{i=1}^{N} \left(y[i] - \mathbf{w}^\top \mathbf{x}[i]\right)^2}_{\mathcal{L}(\mathbf{w})}
\end{aligned}$$

By letting $\nabla \mathcal{L}(\mathbf{w}) = 0$, we get $\mathbf{w}_{\text{MLE}} = (\mathbf{X}^\top \mathbf{X})^{-1}(\mathbf{X}^\top \mathbf{y})$.

Bayesian Linear Regression: We want to model uncertainty over $\mathbf{w}$.

- The coefficient vector $\mathbf{w}$ is now a random variable with a prior $p(\mathbf{w}|v) = N(\mathbf{0}, v\mathbf{I})$.
- Factorization: $p(y[1], \cdots, y[N], \mathbf{w}) = p(\mathbf{w}|v) \prod_{i=1}^{N} p(y[i]|\mathbf{w}^\top \mathbf{x}[i], \sigma_N^2)$.



N data points

K classes

D feature dimensions

Naïve Bayes: *Naïve Bayes* is a model for class $c \in \{1, \cdots K\}$ given input features $\mathbf{x}$: $p(\mathbf{x}, c) = p(\mathbf{x}|c)p(c)$. It can be used to classify new data via Bayes rule: $p(c|\mathbf{x}) = \frac{p(\mathbf{x}|c)p(c)}{\sum_k p(\mathbf{x}|k)p(k)}$ and returns c which maximizes $p(c|\mathbf{x})$.

- Assumption: All features are independent given class $C[i]$:

$$p(\mathbf{x}|c) = \prod_j p(x_j|c).$$

- Given training examples, we can learn each $\theta_{c,j}$ separately:

$$p(\mathbf{x}, c|\phi, \boldsymbol{\theta}) = p(c|\phi) \prod_j p(x_j|c, \theta_{c,j}).$$

Theorem 4.1: Given a graph G over a set of random variables $\mathcal{X} = \{X_1, \cdots, X_N\}$ and P be a joint distribution over the same space. If G is an I-Map for P, then P factorizes according to G.

Theorem 4.2: Let P be a joint distribution over $\mathcal{X}$ and G be a Bayesian network structure over $\mathcal{X}$. If P factorizes according to G, then the local dependence assertions $\mathcal{J}_l(G) \subseteq \mathcal{J}(P)$.

- The *local Markov dependencies* $\mathcal{J}_l(G)$ is the set of all basic conditional independence assertions of the form:

$$\{X_i \perp (X_{\text{nonDesc}(x_i)} \setminus X_{\pi_i}) | X_{\pi_i}\}.$$

Global Markov Independencies: The set of all independencies that correspond to d-separation in graph G is the set of *global Markov independencies*:

$$\mathcal{J}(G) = \{(X \perp Y|Z) : \text{dsep}_G(X; Y|Z)\}.$$

Theorem 4.3 (Soundness): If a distribution P factorizes according to G, then $\mathcal{J}(G) \subseteq \mathcal{J}(P)$. If two nodes are found to be d-separated given Z, they are in fact conditionally independent given Z in P.

Faithful: P is *faithful* to G if for any conditional independence $(X \perp Y|Z) \in \mathcal{J}(P)$ then $\text{dsep}_G(X; Y|Z)$. In other words, any independence in P is reflected as d-separation in the graph G.

Perfect Map: A graph G is a perfect map for a probability distribution P if $\mathcal{J}(P) = \mathcal{J}(G)$.

Theorem 4.4 (Weak Completeness): If $(X \perp Y|Z)$ in all distributions P that factorize over G, then $\text{dsep}_G(X; Y|Z)$.

Theorem 4.5 (Almost Completeness): For almost all distributions P that factorize over G, we have $\mathcal{J}(P) = \mathcal{J}(G)$.

Bayesian networks are sound and almost complete, but they cannot exactly represent all conditional independencies for a given distribution.

4 Markov Random Fields

Generative Models: Approaches that explicitly or implicitly model the distributions of input and outputs (e.g. hidden Markov model, chain structure MRF).

Discriminative Models: Approaches that model the posterior probablities directly (e.g. chain structure CRF).

Markov Random Fields: A *Markov random field*, or *undirected graphical model*, is a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where

- $\mathcal{V}$ is a set of nodes that are in one-to-one correspondence with a set of random variables;
- $\mathcal{E}$ is a set of undirected edges.
- Factorization via Gibbs distribution: $p(x_1, \cdots, x_n) = \frac{1}{Z} \prod_{j=1}^{M} \varphi_j(\mathcal{C}_j)$.
  - ▶ A *factor* $\varphi(\mathcal{C})$ is a function that maps a set of random variables $\mathcal{C} = \{X, \cdots, Z\}$ to a non-negative real number.
- Pros:
  - ▶ No edge orientations, hence more natural for problems such as image analysis and spatial statistics.
  - ▶ Discriminative UGMs work better than discriminative DGMs.
- Cons:
  - ▶ The parameters are less interpretable and less modular.
  - ▶ Parameter estimation is more computationally expensive.
- Misconceptions:
  - ✖ Factors always represent marginal/conditional distributions.
  - ✖ UGMs represent more conditional independencies than DGMs.
  - ✖ UGMs specify a unique factorization.

Global Markov Property: Given the sets of nodes A, B and C, $X_A \perp X_B \mid X_C$ if and only if C separates A from B in the graph $\mathcal{G}$. In other words, there are no trails connecting any node in A to any node in B when we remove all nodes in C.

Local Markov Property: The *Markov blanket* of X_s denoted $\text{mb}(X_s)$ is the set of nodes that renders a node X_s conditionally independent of all the other nodes in $\mathcal{G}$: $X_s \perp \mathcal{V} \setminus \{\text{mb}(X_s), X_s\} \mid \text{mb}(X_s)$.

- The Markov blanket in a UGM is the set of immediate neighbours.
- The Markov blanket in a DGM is the set of the node's parents, children and co-parents (other parents of children).

Pairwise Markov Property: Two nodes X_s and X_t are conditionally independent given the rest if there is no direct edge between them: $X_s \perp X_t \mid \mathcal{V} \setminus \{X_s, X_t\}$ where $\mathcal{E}_s t = \emptyset$.
Note that the three properties are interrelated: $G \Rightarrow L \Rightarrow P$, and $P \Rightarrow G$ if we assume positive distributions ($p(\mathbf{x}) > 0$).

Theorem 5.1 (Hammersley-Clifford): A positive distribution $p(\mathbf{y}) > 0$ satisfies the conditional independence properties of an undirected graph $\mathcal{H}$ if and only if p can be represented as a product of factors, one per maximal clique: $p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\boldsymbol{\theta}_c)$, where
- $\mathcal{C}$ is the set of all maximal cliques of $\mathcal{G}$;
- $\psi_c(\cdot)$ is the *factor* or *potential function* of clique c;
- θ is the parameter of the factor $\psi_c(\cdot)$ for $c \in \mathcal{C}$;
- $Z(\theta)$ is the partition function $Z(\theta) \triangleq \sum_{\mathbf{y}} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\boldsymbol{\theta}_c)$.

Log-Linear Form: $p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left(\sum_{c \in \mathcal{C}} \boldsymbol{\theta}_c^\top \boldsymbol{\phi}_c(\mathbf{y})\right)$. In this way,

$$\log(\psi_c|\mathbf{y}_c) = \boldsymbol{\phi}_c(\mathbf{y}_c)^\top \boldsymbol{\theta}_c$$
$$\log p(\mathbf{y}|\boldsymbol{\theta}) = \sum_{c \in \mathcal{C}} \boldsymbol{\phi}_c(\mathbf{y}_c)^\top \boldsymbol{\theta}_c - \log Z(\boldsymbol{\theta}).$$

- Every finite MRF is an exponential family.
- We can also specify $p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left(-\sum_{c \in \mathcal{C}} E\left(\mathbf{y}_c|\boldsymbol{\theta}_c\right)\right)$, where E is the *energy* associated with the variables in clique c.

Parameter Learning via MLE: Consider an MRF in log-linear form, where c indexes the cliques.

$$\text{MLE: } \arg\max_{\boldsymbol{\theta}} p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left(\sum_{c \in \mathcal{C}} \boldsymbol{\theta}_c^\top \boldsymbol{\phi}_c(\mathbf{y})\right).$$

Its scaled log-likelihood is given by

$$l(\boldsymbol{\theta}) \triangleq \frac{1}{N} \sum_{i=1}^N \log p(\mathbf{y}_i|\boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^N \left[\sum_{c \in \mathcal{C}} \boldsymbol{\theta}_c^\top \boldsymbol{\phi}_c(\mathbf{y}_i) - \log Z(\boldsymbol{\theta})\right]$$
$$\frac{\partial l}{\partial \boldsymbol{\theta}_c} = \frac{1}{N} \sum_{i=1}^N \left[\boldsymbol{\phi}_c(\mathbf{y}_i) - \frac{\partial}{\partial \boldsymbol{\theta}_c} \log Z(\boldsymbol{\theta})\right]$$
$$= \frac{1}{N} \sum_{i=1}^N [\boldsymbol{\phi}_c(\mathbf{y}_i) - \mathbb{E}[\phi_c(\mathbf{y})|\boldsymbol{\theta}]] \ (\textit{derivative of log partition})$$
$$= \underbrace{\frac{1}{N} \sum_{i=1}^N \boldsymbol{\phi}_c(\mathbf{y}_i)}_{\text{clamped term}} - \underbrace{\mathbb{E}\left[\phi_c(\mathbf{y})\middle|\boldsymbol{\theta}\right]}_{\text{unclamped/contrastive term}}$$

- l is convex in $\boldsymbol{\theta}$, hence p has unique global maximum.
- Unclamped term requires inference, which makes UGM much slower than DGM.

Parameter Learning via MAP:

$$\text{MAP: } \arg\max_{\boldsymbol{\theta}} \left\{\sum_{i=1}^N \log p(\mathbf{y}_i|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta})\right\}.$$

- We use a Gaussian prior where (μ, Σ) are hyperparameters.

$$p(\boldsymbol{\theta}) = \text{Norm}_{\boldsymbol{\theta}}[\boldsymbol{\mu}, \boldsymbol{\Sigma}]$$
$$= \frac{1}{(2\pi)^{\frac{D}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} e^{-\frac{1}{2}(\boldsymbol{\theta}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}-\boldsymbol{\mu})}$$

Conditional Random Fields: A *conditional random field*, or *discriminative random field*, is an MRF where all the clique potentials are conditioned on input feature X:

$$p(\mathbf{y}|\mathbf{x}, \mathbf{w}) = \frac{1}{Z(\mathbf{x}, \mathbf{w})} \prod_c \psi_c(\mathbf{y}_c|\mathbf{x}, \mathbf{w}).$$

- Log-linear of potentials: $\psi_c(\mathbf{y}_c|\mathbf{x}, \mathbf{w}) = \exp\left(\mathbf{w}_c^\top \boldsymbol{\phi}(\mathbf{x}, \mathbf{y}_c)\right)$.
  - ▶ $\boldsymbol{\phi}(x, y_c)$ is a feature vector derived from the global inputs X and the local set of labels Y_c.

Theorem 5.2 (Soundness): If P is a Gibbs distribution over H, then H is an I-Map for P.

- Hammersley-Clifford states that for positive distributions P is a Gibbs distribution over H if and only if H is an I-Map for P.

Theorem 5.3 (Weak Completeness): If X and Y are not separated in H, then there is some distribution P that factorizes over H where X and Y are dependent.

Markov random fields are sound and almost complete, but they cannot ex actly represent all conditional independencies for a given distribution.

Algorithm 19.1: Stochastic maximum likelihood for fitting an MRF

Initialize weights $\boldsymbol{\theta}$ randomly;
$k = 0$, $\eta = 1$;
for *each epoch* **do**
  for *each minibatch of size* B **do**
    for *each sample* $s = 1 : S$ **do**
      Sample $\mathbf{y}^{s,k} \sim p(\mathbf{y}|\boldsymbol{\theta}_k)$;
    $\hat{E}(\boldsymbol{\phi}(\mathbf{y})) = \frac{1}{S} \sum_{s=1}^S \boldsymbol{\phi}(\mathbf{y}^{s,k})$;
    for *each training case* i *in minibatch* **do**
      $\mathbf{g}_{ik} = \boldsymbol{\phi}(\mathbf{y}_i) - \hat{E}(\boldsymbol{\phi}(\mathbf{y}))$;
    $\mathbf{g}_k = \frac{1}{B} \sum_{i \in B} \mathbf{g}_{ik}$;
    $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \mathbf{g}_k$;
    $k = k + 1$;
    Decrease step size η;