Probabilistic Neural Networks

- $X = \{X_i \mid i \in V\}$ – collection of K-valued random variables
- (V, E) – directed acyclic graph (DAG)

Definition 1 X is a Bayesian network (aka belief network) w.r.t. the DAG (V, E) if its joint distribution is

$$p(x) = \prod_{i \in V} p(x_i \mid x_{pa(i)}),$$

where $pa(i) \subset V$ denotes the parents of node $i \in V$ □

Let $de(i)$ denote all descendants of a node $i \in V$. Then X_i conditioned on $X_{pa(i)}$ is independent of all X_j with $j \in V \setminus de(i)$, i.e.

$$\not\exists\ X_i \perp\!\!\!\perp X_{V \setminus de(i)} \mid X_{pa(i)}$$

Remark 1 Notice that BNs do not imply causality. For example, a Markov chain model is both, a Markov model on an undirected graph (chain) and a BN on a directed chain.

4. Probabilistic networks

In probabilistic networks all units are considered random. The units are organised in layers. Each layer $k = 1, \dots, \ell$ is described by a random vector X^k with dimension n_k (number of layer units). The network $\not\in$ is a conditional Bayesian network with pdf

$$p(X^{1 \dots \ell} \mid X^0) = \prod_{k=1}^{\ell} p(X^k \mid X^{k-1})$$

where it is usually assumed that the conditional distributions factorise as

$$p(X^k \mid X^{k-1}) = \prod_{i=1}^{n_k} p(X_i^k \mid X^{k-1})$$

1C. Approximate inference for PNN

Computing $p(x^k|x^0)$ for probabilistic NN is not tractable.

Approximate $p(x^k|x^0) \approx \prod_{i=1}^{n_k} q(x_i^k)$ with some simple distributions q. This factorised approximation can be computed layer by layer

$$q(x_i^k) = \mathbb{E}_{q(x^{k-1})}[p(x^k|x^{k-1})]$$
$$= \int p(x^k|x^{k-1}) \prod_{i=1}^{n_{k-1}} q(x_i^{k-1})\, dx^{k-1}$$

Linear layer $A = \langle w, x \rangle$, where $\mathbb{E}[x] = \mu$, $\mathbb{V}[x] = \sigma^2 \mathbb{I}$

Then $\mu' = \mathbb{E}[A] = \langle w, \mu \rangle$

$$\sigma'^2 \approx \sum_i w_i^2 \sigma_i^2$$

Non-linear (coordinate-wise) mappings A is a scalar r.v. with statistic (μ, σ^2). $Y = f(A - Z)$ with independent noise. Assume $A' = A - Z$ is normally / logistically distributed with (μ', σ'^2). We can approximate the expectation and variance of $Y = f(A')$

$$\tilde{\mu} = \mathbb{E}_{q(A')}[f(A')], \quad \tilde{\sigma}^2 = \mathbb{E}_{q(A')}[f^2(A')] - \tilde{\mu}^2$$

Example 2 (Heaviside nonlinearity with noise)

We consider $Y = h(A - Z)$, where Z is standard logistic noise. Statistic of $A' = A - Z$ is given by $\mu' = \mu$, $\sigma'^2 = \sigma^2 + \sigma_s^2$, where $\sigma_s^2 = \pi^2/3$

The posterior distribution of each layer $k>0$ given the observations x^0 expresses recurrently as

$$p(x^k|x_0) = \int p(x^k|x^{k-1})\, p(x^{k-1}|x_0)\, dx^{k-1}$$

Remark 2 Computing these (conditional) marginal distributions is usually NP-hard. However, sampling a realisation $x^1, \dots, x^\ell$ given x^0 is often easy (linear in model size).

B. Standard NN → Probabilistic NN

Standard (deterministic) artificial neuron

$$y = f(\underbrace{\textstyle\sum_i w_i x_i + b}_{a}) = f(a)$$

Inject noise Z with known distribution (e.g. normal distr.)

$$Y = f(A - Z)$$

We have then $F_Y(u) = \mathbb{P}(f(A-Z) < u)$

Example 1 (Stochastic binary unit)

Let Y be a binary r.v. $Y = h(A-Z)$, where h is the Heaviside step function and Z is noise with cdf. F_Z. Then

$$\mathbb{P}(Y=1|A) = \mathbb{P}(A-Z>0|A) = \mathbb{P}(Z<A) = F_Z(A).$$

If for instance Z has standard logistic distribution, then $P(Y=1|A) = S(A)$, where S is the sigmoid function $S(a) = (1+e^{-a})^{-1}$

Assuming A' to have logistic distribution, we get the statistic of Y by

$$\hat{\mu} = \mathbb{E}[h(A')] = \mathbb{P}(A'>0) = \mathbb{P}\left(\frac{A'-\mu'}{\sigma'/\sigma_s} \geq -\frac{\mu'}{\sigma'/\sigma_s}\right) =$$

$$= S\left(\frac{\mu'}{\sigma'/\sigma_s}\right) = S\left(\frac{\mu}{\sqrt{\sigma^2/\sigma_s^2+1}}\right)$$

□

Using such approximations for ReLU, max-pooling and for categorical layers, we can forward propagate means and variances through the layers of probabilistic NN and then apply back-propagation.