Modern ML for particle physics

0. Plan of attack

1. Bayesian neural networks – (errors and regularization)
2. Generative Models I: GANs + applications + VAE
3. Generative Models II: Normalizing flow + applications + (diffusion models)
4. Anomaly detection

I. Bayesian Neural Networks (nothing really bayesian about them)

So far: NNs are deterministic → one input → one output

However: in physics we are not only interested in results, but also in errorbars

Main idea:

deterministic weights θ → distribution $\theta \sim p(\theta|X)$

1 input → distribution of output → $\langle y \rangle + \sigma_y$
$p(y|x)$

I.1. Basics

Now, we are interested to evaluate $p(y|x)$, $y = f(x)$ $\langle y(x) \rangle$ (average output)

which is given by:

$$p(y|x) = \int d\theta \; p(y|\theta, x)\, p(\theta|x)$$

↑ require knowledge of !

⇒ Variational approximation:

$$p(\theta|x) \approx q_\alpha(\theta|x) \overset{\text{simple}}{=} q_\alpha(\theta)$$

↑ parameters to adjust to match $p(\theta|x)$

KL:

$$\arg\min_\alpha \; KL\left[q_\alpha(\theta), p(\theta|x)\right]$$

where KL is given by:

$$KL[q_\alpha(\theta), p(\theta|x)] = \int d\theta\, q_\alpha(\theta) \log \frac{q_\alpha(\theta)}{p(\theta|x)}$$

Baye's Theorem: likelihood, prior

$$p(\theta|x) = \frac{p(x|\theta)\cdot p(\theta)}{p(x)}$$

$p(\theta|x)$: posterior; $p(x)$: normalization

$$= \int d\theta\, q_\alpha(\theta) \log \frac{q_\alpha(\theta)\, p(x)}{p(\theta)\, p(x|\theta)}$$

$$= KL[q_\alpha(\theta), p(\theta)] - \int d\theta\, q_\alpha(\theta) \cdot \log p(x|\theta) + \underbrace{\log p(x)}_{\text{intractable!}} \underbrace{\int d\theta\, q_\alpha(\theta)}_{=1}$$

From this, we can define: ① + ② = − ELBO → see later!

$$\mathcal{L}_{BNN} = \underbrace{-\int d\theta\, q_\alpha(\theta) \log p(x|\theta)}_{①} + \underbrace{KL[q_\alpha(\theta), p\theta]}_{②}$$

① neg-log likelihood averaged over $q_\alpha(\theta)$ ← in practice see later!

② $q_\alpha(\theta)$ should not deviate too much from prior!

In practice: p & q are gaussians $\alpha \in \{\mu, \sigma\}$

$$q_\alpha(\theta) \equiv q_{\mu\sigma}(\theta) = \frac{1}{\sqrt{2\pi\sigma_q^2}} \cdot e^{-\frac{(\theta-\mu_q)^2}{2\sigma_q^2}}$$

θ: mean μ_q and with σ_q

→ same for $p(\theta)$ ~ usually $\mu_p = 0$

$$\rightarrow KL[q_\alpha(\theta), p(\theta)] = \frac{\sigma_q^2 - \sigma_p^2}{2\sigma_p^2} + \frac{\mu_q^2}{2\sigma_p^2} + \log \frac{\sigma_p}{\sigma_q}$$

→ ② is regularization!

Deterministic limit: $\sigma_q \to 0$, $\mu_q \to \theta_0$ $\} \to q(\theta) = \delta(\theta - \theta_0)$

$$\mathcal{L}_{BNN} = -\log(p(x|\theta_0) + \underbrace{\frac{\theta_0^2}{2\sigma_p^2}}_{\text{L2 regularization!}} + \text{const}$$

usually λ free parameter!

Comments:

- for K parameters of NN use product:

$$q_\alpha(\theta) = \prod_{i=1}^{K} N(\theta_i | \mu_i, \sigma_i)$$

⇒ BNN has 2K parameters (Mean-field approx. [1601.00670])

with correlations → BNN ~ K^2 params → bad scaling

↳ deepness of NN compensates for approx?

→ open question!

I.2. Example: Bayesian regression [2206.14831]

we are interested in fitting amplitudes A

training data: $\{x, A(x)\}$

We can define $p(A|x)$ ~ prob of A @ pos x ($\sim p(y|x)$)

then we can define:

$$\boxed{\langle A(x)\rangle = \int dA\, A\, p(A|x)} \quad \text{with} \quad \boxed{p(A|x) = \int d\theta\, p(A|\theta,x)\, p(\theta|x)}$$

Now, using our approximation $p(\theta|x) \simeq q_\alpha(\theta)$

Now we want to describe this as mean and σ of prediction so we write:

$$(*) \quad \langle A\rangle = \int dA\, A\, p(A|x)$$

$$= \int dA \int d\theta\, A\, p(A|\theta,x)\, q_\alpha(\theta) \qquad \text{(network output)}$$

$$= \int d\theta\, q_\alpha(\theta) \underbrace{\int dA\, p(A|x,\theta)\, A}_{\equiv \bar{A}(x,\theta)}$$

$$= \int d\theta\, q_\alpha(\theta)\, \bar{A}(x,\theta) \leftarrow \text{first network output!}$$

Variance of A:

$$\sigma^2_{tot} = \langle (A - \langle A \rangle)^2 \rangle$$

$$= \int dA \; (A - \langle A \rangle)^2 \, p(A|x)$$

$$= \int dA \int d\Theta \; (A - \langle A \rangle)^2 \, q_\alpha(\Theta) \cdot p(A|x,\Theta)$$

$$= \int d\Theta \; q_\alpha(\Theta) \cdot \Big[\int dA \; A^2 \, p(A|x,\Theta) - 2\langle A \rangle \int dA \; A \, p(A,x,\Theta) + \langle A \rangle^2 \int dA \; p(A|x,\Theta) \Big]$$

$$= \int d\Theta \; q_\alpha(\Theta) \Big[\underbrace{\bar{A}^2(x,\Theta) - \bar{A}(x,\Theta)^2}_{\sigma^2_{model} \equiv \sigma_{stoch}} + \underbrace{\big(\bar{A}(x,\Theta) - \langle A \rangle\big)^2}_{\sigma^2_{pred}} \Big]$$

Defines two kind of uncertainties:

1. $$\sigma^2_{pred} = \int d\Theta \; q_\alpha(\Theta) \Big[\bar{A}(x,\Theta) - \langle A \rangle \Big]^2$$

   - ↳ following (*) this vanishes for $q(\Theta) \to \delta(\Theta - \Theta_0)$
   - ↳ with precise training data → better training
     - ↳ decreases with more data!
   - ↳ represents statistical uncertainty (epistemic)

2. $$\sigma^2_{stoch} \; \big(\equiv \langle \sigma^2_{stoch}(\Theta) \rangle \big) \qquad (**)$$
   $$= \int d\Theta \; q_\alpha(\Theta) \; \sigma^2_{stoch}(\Theta)$$
   $$= \int d\Theta \; q_\alpha(\Theta) \big(\bar{A}^2(x,\Theta) - \bar{A}(x,\Theta)^2 \big) \equiv \langle \sigma^2_{model}(\Theta) \rangle_\Theta$$

   - ↳ already occurs without sampling
   - ↳ does not vanish for $q(\Theta) \to \delta(\Theta - \Theta_0)$
   - ↳ reaches constant
   - ↳ represents systematic uncertainty (aleatoric)

     (stoch training data, bad hyps, model expressitivity)

(*) and (**) are network outputs:

$$\text{BNN}: x, \theta \to \begin{pmatrix} \bar{A}(\theta) \\ \sigma_{\text{model}}(\theta) \end{pmatrix} \leftarrow \text{in practic } \log \sigma^2$$

We obtain now the outputs as:

$$\langle A \rangle \cong \frac{1}{L} \sum_{i=1}^{L} \bar{A}(\theta_i) \qquad \sigma^2_{\text{stoch}} \simeq \frac{1}{L} \sum_{i=1}^{L} \sigma^2_{\text{stoch}}(\theta_i)$$

$$\sigma^2_{\text{pred}} \simeq \frac{1}{L} \sum_{i=1}^{L} (\bar{A}(\theta_i) - \langle A \rangle)^2 \qquad \text{with } \theta_i \sim q_\omega(\theta)$$

usually $L \simeq 10$ for evaluating and getting errorbars!

Possible we do not expect $\bar{A}(x,\theta)$ to be gaussian we can assum this for $\sigma_{\text{model}}(\theta)$. (Simplification) ($q_\omega(\theta)$ is gaussian)!

$$\mathcal{L}_{\text{BNN}} = \int d\theta \, q_\omega(\theta) \sum_j \left[\frac{\bar{A}_j(\theta) - A_j^{\text{truth}}}{2\sigma^2_{\text{model}}(x_j,\theta)} + \log \sigma_{\text{model}} \right]$$

$$+ \underbrace{\frac{\sigma_q^2 - \sigma_p^2 + (\mu_q - \mu_p)^2}{2\sigma_p^2} + \log \frac{\sigma_p}{\sigma_q}}_{KL(q_\omega(\theta), p(\theta))}$$

→ minimize μ_q and σ_q

→ data has no uncertainty → NN constructs one!

w/o sampling θ we obtain:

$$\mathcal{L}_{\text{heteroskedastic}} = \sum_j \left[\frac{(\bar{A}_j(\theta_0) - A_j^{\text{truth}})^2}{2\sigma_{\text{model},j}(\theta_0)^2} + \log \sigma_{\text{model},j}(\theta_0) \right] + \frac{\theta_0^2}{2\sigma_p^2}$$

→ show plots from paper → check-out for yourself!

↳ more details in thesis of Michel Luchmann

[Doi: 10.11588/heidok.00032414]