- Probabilistic M.L.
  - Models, Algorithms
    - Training, Estimation
    - Inference, testing

- Bayesian NN
- Topic Model
- Graphical Model
  - MRF — Undirected
  - BN — Directed.

- Gaussian Process

Each r.v. is a gaussian

Uses Kernel → defines an inner product and induces a space called Hilbert space.

Normalized:

$$K(x,x)=1, \quad -1 \leq K(x,x') \leq 1$$

$$K(x,x') = K(x',x) \qquad v^T K v \geq 0 \quad \forall v$$

For each node in BNN, there's a μ_j and σ_j

$$\mu_j = \sigma\left(\sum_i w_{ij} x_i + b_j\right)$$

$$\sigma_j = \sigma\left(\sum_i \phi_{ij} x_i + b_j'\right)$$

$$z_j = \text{sample}(\mu_j, \sigma_j)$$

Similar to VAE

Backprop is difficult :(

Another approach:

$$z_{ij} = \sigma\left(\sum_i w_{ij} x_i + b_j\right)$$

$$G_{ij} \sim (\mu_{ij}, \sigma_{ij})$$

Similar to Dropout

↳ Learns wt. distr.

- Epistemic
  - Related to training

} Reducible

- Using more data
- Better optimization.

- Aleatoric
  - Related to data/problem

BNN done in broadly two aspects :

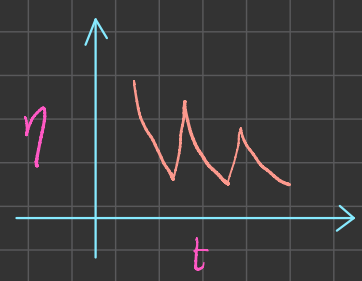
- distr. over activation
- " " wts.

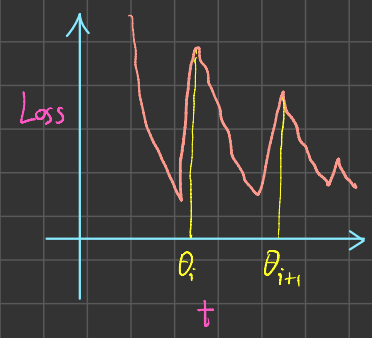
Variational Inference is distr. over wts.

$$\theta^0, \theta^1, \ldots, \theta^T \longrightarrow \overbrace{\text{SWA}}^{\text{Stochastic weighted avg.}}\text{G}$$

Gaussian

Uses cyclic learning rate





$$\bar{\theta} = \frac{1}{T}\sum_{t=1}^{T}\theta^t \qquad \hat{y} = f(x, \hat{\theta})$$

ECE → Expected Calibration Error

↳ Used as metric of uncertainty.

$$\frac{1}{B}\sum_{i=1}^{|B|}\left|Acc_i - Prob_i\right|$$

Eg -

0-5	5-10	....

Acc_1 = Accuracy over all classes with confidence from 0 to 5

$Prob_1$ = Prediction probability over 0 to 5 confidence classes.

— Epistemic — model related

— Alleatoric — data / problem related

(not reducible) ⟶ Bayes error rate

— Prob. classifier

$$ECE = \sum_{b=1}^{B} \frac{n_b}{N} \left| acc(b) - conf(b) \right|$$

Binning is done else if the no. of elements in each bin goes to 1, the acc(b) becomes very noisy (0 or 1)

Generalization ⟶ test inputs from distribution not in training.

OOD ⟶ out-of-distribution

Ensembles are computationally expensive!

Deep Ensemble : W is trained first and then W is frozen and r_i, s_i are updated in one go [SGD in a mini-batch fashion for loss for each r_i, s_i]

Hyperparameter Ensemble: [a generalization of Deep Ensemble]

$$\left(w \circ r_i s_i^T\right) + e(\lambda)\Delta$$

$$\underbrace{(\theta_{init}, \lambda)}_{\text{Searched over these}} \quad \Big| \quad \underbrace{(r, s, \Delta, e)}_{\text{for each search, optimise this}}$$

Bayesian models require a good prior, Ensembles don't.

" " posterior weighted avg., in ensembles each prediction is directly the confidence score.

So, simple avg!

Summary of BNNs:

Used for posterior distr. over NN models.

→ Epistemic uncertainty estimation.

→ $q(\theta)$
→ SGD
→ MCMC

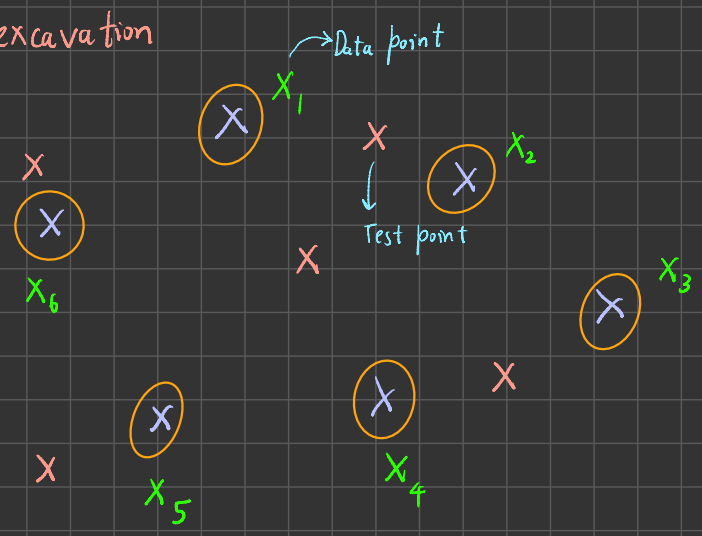
Ensemble vs Bayesian

Avg. over prediction (Inference)

[Posterior] Distribution over model parameters.

Gaussian Process ⟶ quintessential application is $\boxed{\text{Kriging}}$

Eg- Oil excavation



Distribution over oil at different data points, found using data points, assuming Gaussian Process.

Kernel $\longrightarrow$ A matrix which tells the "similarity"/"distance" b/w data points

To Deep Kernel Learning:

feature, NOT r.v.

$$K(x, x') = e^{\frac{-\|x - x'\|^2}{\sigma^2}}$$

$$e^{\frac{-\|\phi(x) - \phi(x')\|^2}{\sigma^2}} \qquad \phi - \text{D.N.N}$$

$$\hat{\mu} = K_N^T C_N^{-1} t$$

$\hookrightarrow$ true label vector for training points

$$\hat{\sigma} = c - K^T C_N^{-1} K$$

C_N — Training Kernel Matrix

K — Kernel vector b/w test and training data points.

For DNN:

$$\hat{\mu}(\phi) = K_N^T(\phi) C_N^{-1}(\phi) t$$

$$\hat{\sigma}(\phi) = c(\phi) - K^T(\phi) C_N^{-1}(\phi) K(\phi)$$

N.L.L. $(\mu(\phi), \sigma(\phi))$

$\hookrightarrow$ Minimise this, to get ϕ

Spectral Normalization:

hidden layer in DNN

$$(1-L)\,\|x-x'\| \le \|h(x)-h(x')\| \le (1+L)\,\|x-x'\|$$

$$\boxed{\|W\| \le C_L}$$

$$\overbrace{\phi^T\phi}^{=W} \le C_L$$

spectral norm $[\lambda_{max}(W)]$

Post-doc calibration: [why → Most models are uncalibrated, overconfident]

- Histogram Binning
- Platt → Composition under some function results in calibration.
- Temperature Scaling → Temperature param. learned to smoothen predictions out.

Conformal Prediction: User expects correctedness of answer to be $1-\varepsilon$

→ Predicts a small set of plausible answers, which provably contains right answer with prob. $1-\varepsilon$.

Coverage Score → $P(Y_{n+1} \in C)$