Uncertainty Quantification

APPROXIMATE BAYESIAN ENSEMBLING

Amritha S. Agarwal
230854
Supervisor: Simon Klüttermann

Overview

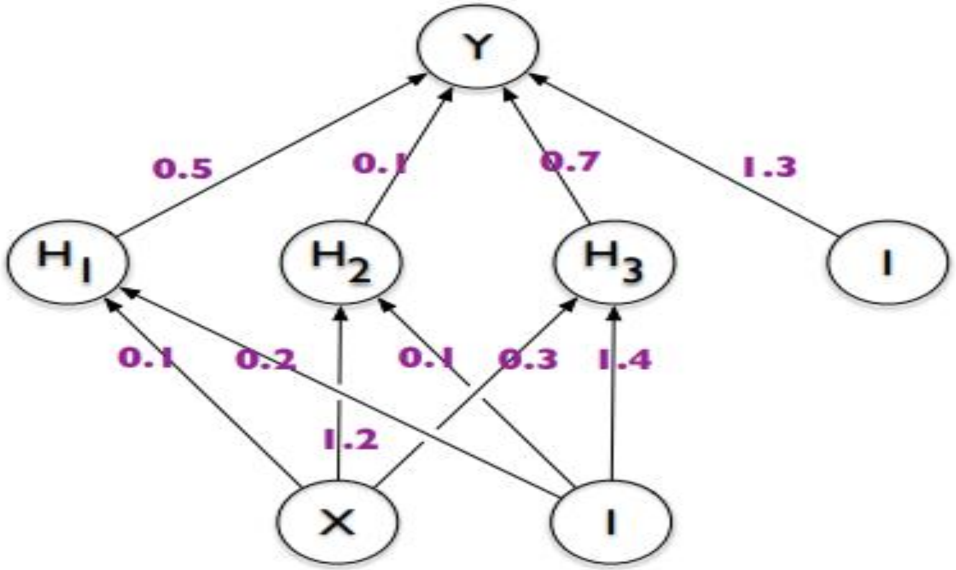
- Motivation
- Bayesian NN‘s
  - Challenges with BNN‘s
- Ensembling
- Anchored Ensembling
  - Randomised MAP sampling(RMS)
  - RMS for NN‘s
- Conclusion

Motivating Uncertainty and BNN's

- Neural Network architecture + $\frac{P(B|A)\,P(A)}{P(B)}$ = Bayesian Neural Network(BNN)

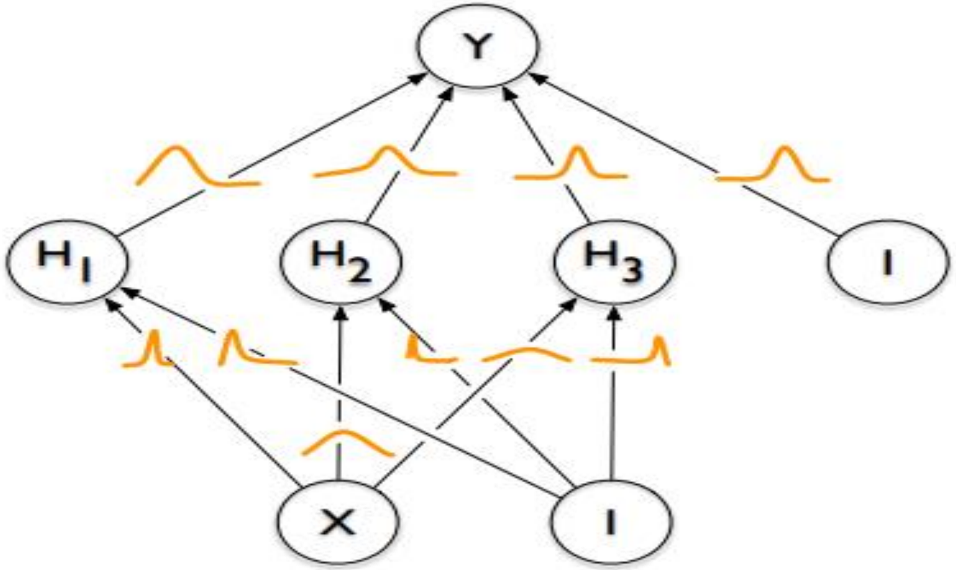
Regular NNs

- Learn a single point estimate of the parameter.



Bayesian NNs

- Learn the probability density over a parameter space.



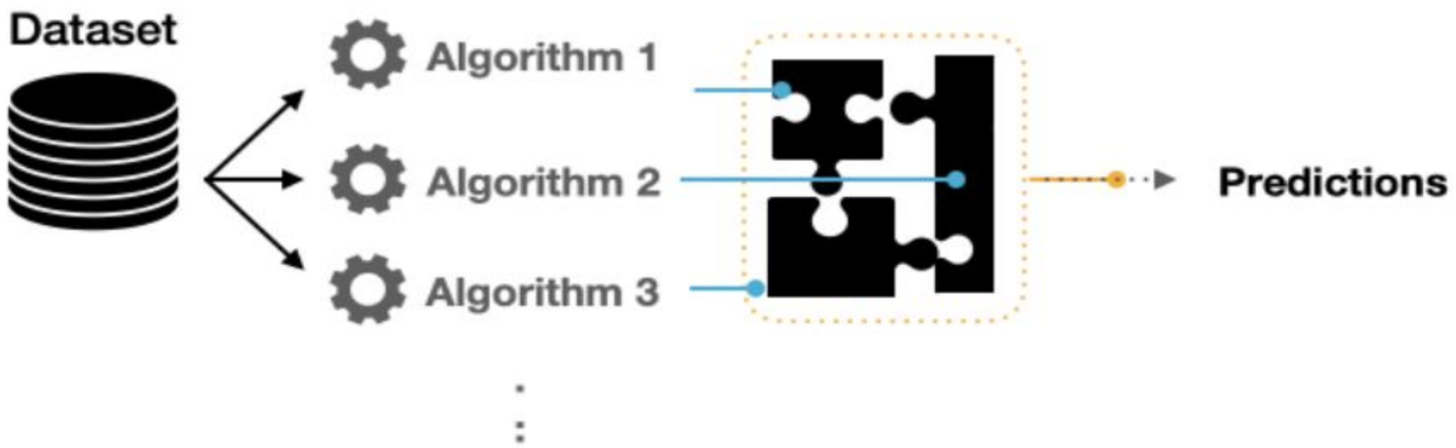
Source: https://sanjaykthakur.com/2018/12/05/the-very-basics-of-bayesian-neural-networks/

Challenges with BNN‘s

- Trouble with many parameters (usually really big NN‘s)
- Too many data points
- Very expensive

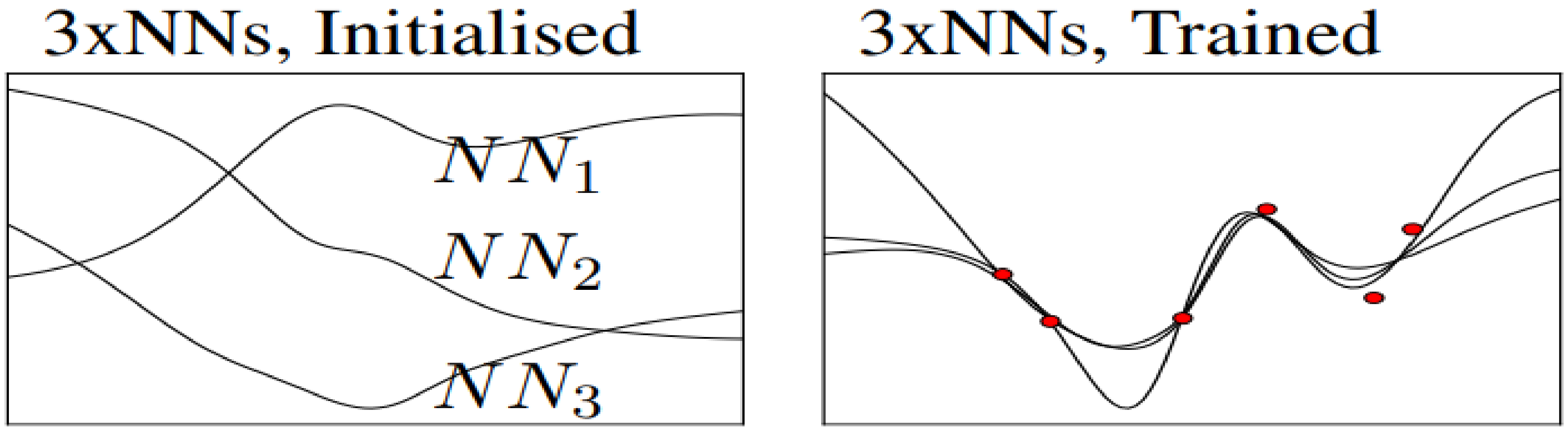
Ensembling (Unbayesian approach)

- Bunch of independently trained models & form a mixture of models – ensemble

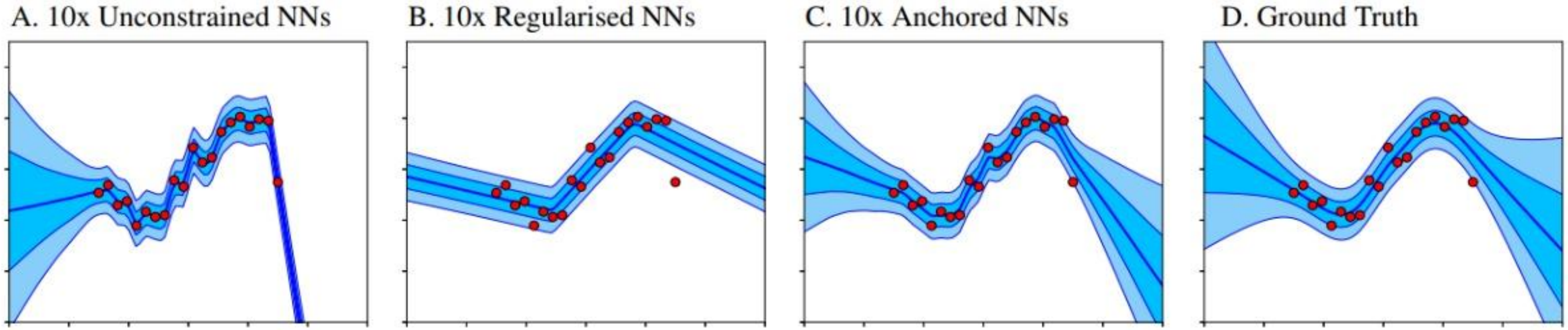


Source: https:https://builtin.com/machine-learning/ensemble-model

Ensembling



Anchored Ensembling



$$\frac{1}{N}||y - \hat{y}_j||_2^2$$

$$\frac{1}{N}||y - \hat{y}_j||_2^2 + \frac{1}{N}||\Gamma^{\frac{1}{2}}(\theta_j)||_2^2$$

$$\frac{1}{N}||y - \hat{y}_j||_2^2 + \frac{1}{N}||\Gamma^{\frac{1}{2}}(\theta_j - \theta_{anc,j})||_2^2$$

Anchored Ensembling

Randomised MAP sampling

- Maximum a posteriori (MAP) estimate
  - Adding a regularisation term to loss function returns a MAP estimate of the posterior (point estimate)
  - Add noise and sample

RMS for NN's

θ_{MAP} = argmax$_\theta P(\theta|\mathcal{D})$

θ_{MAP} = argmax$_\theta P_{\mathcal{D}}(\mathcal{D}|\theta)P(\theta)$

θ_{MAP} = argmax$_\theta$ $\log\big(P_{\mathcal{D}}(\mathcal{D}|\theta)\big) + \log(P(\theta))$

Assume $P(\theta)$ = $\mathcal{N}(\mu, \Sigma)$, we get,

θ_{MAP} = argmax$_\theta$ $\log\big(P_{\mathcal{D}}(\mathcal{D}|\theta)\big) - \frac{1}{2}||\Sigma^{-1/2}(\theta - \mu)||_2^2$

Now, for RMS, replace μ_{prior} with θ_{anc}

$\boldsymbol{f}_{MAP}(\theta_{anc})$ = argmax$_\theta$ $\log\big(P_{\mathcal{D}}(\mathcal{D}|\theta)\big) - \frac{1}{2}||\Sigma^{-1/2}(\theta - \theta_{anc})||_2^2$

Challenge: choosing a distribution for θ_{anc} such that $P\big(\boldsymbol{f}_{MAP}(\theta_{anc})\big) = \; P(\theta|\mathcal{D})$

Regression: $Loss_j = \frac{1}{N}||\big(y - \hat{y}_j\big)||_2^2 + \frac{1}{N}||\Sigma^{-1/2}\big(\theta_j - \theta_{anc,j}\big)||_2^2$

RMS for NN's

- Assume $P_\theta(\mathcal{D}|\theta) \propto \mathcal{N}(\mu_{like}, \Sigma_{like})$

$P(\theta_{anc}) = \mathcal{N}(\mu_{anc}, \Sigma_{anc})$ where, $\mu_{anc} = \mu_{prior}$; $\Sigma_{anc} = \Sigma_{prior} + \Sigma_{prior}\Sigma_{like}{}^{-1}\Sigma_{prior}$

- Practical workaround: Set $\Sigma_{anc} = \Sigma_{prior}$

$P(\theta_{anc}) = \mathcal{N}(\mu_{anc}, \Sigma_{anc})$ with $\mu_{anc} = \mu_{prior}$; $\Sigma_{anc} = \Sigma_{prior}$

CIFAR-10 Image Classification, VGG-13 CNN

	Accuracy	Train	Edge	Fashion	MNIST	Scramble	Invert	Noise
1xNNs Reg.	81.6%	0.671	0.466	0.440	0.540	0.459	0.324	0.948
5xNNs Uncons.	85.0%	0.607	0.330	0.208	0.275	0.175	0.209	0.380
5xNNs Reg.	86.1%	0.594	0.296	0.219	0.188	**0.106**	0.153	0.598
5xNNs Anch.	85.6%	0.567	**0.258**	**0.184**	**0.149**	0.134	**0.136**	**0.118**
10xNNs Anch.	86.0%	0.549	0.256	0.119	0.145	0.122	0.124	0.161