Bayesian Inference Lighter and Faster

Coby Penso, Dr. Ethan Fetaya

Faculty of Engineering, Bar Ilan University

What we'll see

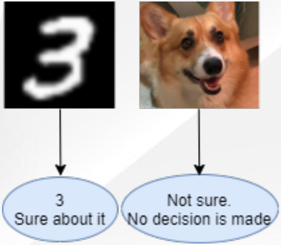
1. Uncertainty
2. Bayesian Networks
3. Existing Methods
4. Our Method
5. Experiments and Results

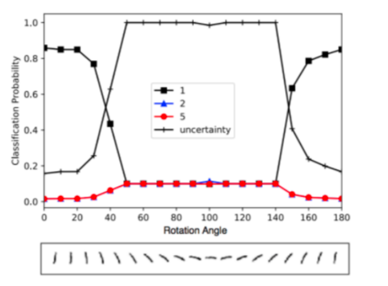
Model Uncertainty

Neural Networks achieving amazing results in many fields such as

- Computer Vision
- Natural Language Processing
- Medical Diagnosing

But suffer often from **Over confidence** in their predictions and specifically confidence in wrong once.
That's why we would like to have a measurement of uncertainty.





Bayesian Modeling

Bayesian Inference

Bayesian inference is a method of statistical inference in which Bayes' theorem is used to update the probability for a hypothesis as more evidence or information becomes available.

Start with a prior on the the model parameters:

$$X = \{x_1, x_2, ..., x_N\}, \quad Y = \{y_1, y_2, ..., y_N\}$$

$$p(y = d|x, w) = \frac{exp(f_d(x; w)}{\sum_{d'} exp(f_{d'}(x, w))}$$

From Bayes rule we get:

$$p(w|X, Y) = \frac{p(Y|X, w)p(w)}{p(Y|X)}$$

Bayesian Modeling

At inference time: Given new input x^* the output y^*:

$$p(y^*|x^*, X, Y) = \int p(y^*|x^*, w)p(w|X, Y)dw$$

A key component in posterior evaluation is the normaliser, also called model evidence:

$$p(Y|X) = \int p(Y|X, w)p(w)dw$$

Posterior Sampling Techniques

Calculating the Posterior analytically is infeasible.
Family of methods tries to sample from the posterior:

- Monte Carlo Markov chain - MCMC, HMC, NUTS
- Stochastic gradient Langevin dynamics

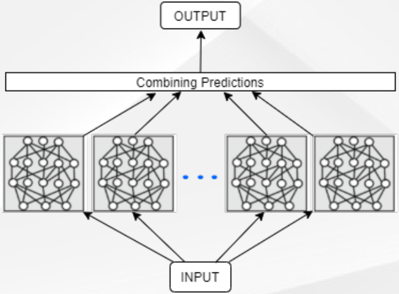
,
Such that in the end of training we have M Posterior Samples $\{W_1, W_2, \ldots, W_M\}$

At inference:

$$p(y^*|x^*, X, Y) = \frac{1}{Z}\sum_{i=1}^{M} p(y^*|x^*, W_i)$$

Difficulties with current methods

- **Time Complexity:**
  Predict Through each of the M models for final prediction
- **Memory Complexity:**
  At Inference time, hold in memory all M models parameters.



Our Method

The Idea: Model the predictive distribution directly

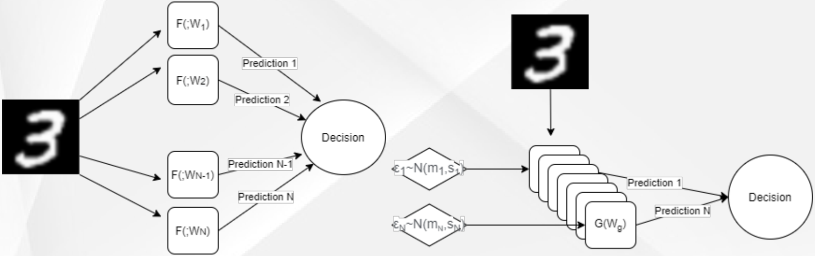


Figure: Left - Posterior samples used in inference. Right - Generative model that models the predictive distribution

Training Phase

Algorithm 1 Training

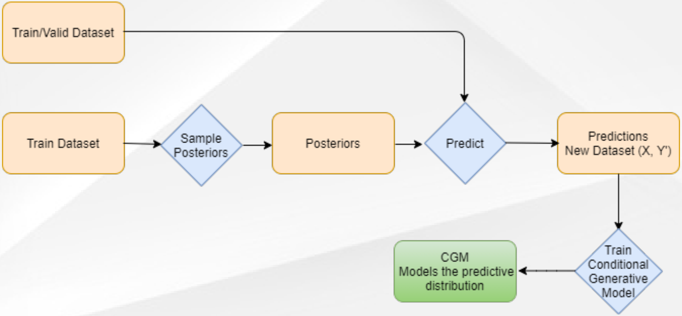
1: Given (X_{train}, Y_{train}) and Model $F(\theta)$

2: Sample M posteriors $\{\theta_1, \ldots, \theta_M\}$

3: Predict on X_{valid}: $\forall i \in \{1, \ldots, N\}, \forall x_j \in X_{valid} : Y_i = F(x_{;i})$

4: Using $\{(X_i, Y_i^j)\}_{j=1,\ldots,N}^{i=1,\ldots,M}$ Train Conditional Generative Model(CGM) Such that

$$(G(\epsilon_1; X), \ldots, G(\epsilon_M; X)) \sim (F(X; \theta_1), \ldots, F(X; \theta_M))$$



Inference Phase

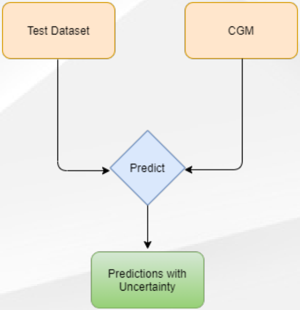
Algorithm 2 Inference

1: Given X_{test}

2: Predict using CGM:

$$Y = argmax(G(\epsilon_1; X), ..., G(\epsilon_M; X))$$

$$Unceratainty = Histogram(G(\epsilon_1; X), ..., G(\epsilon_M; X))$$



Get into details

How do we achieve

$$(G(\epsilon_1; X), \ldots, G(\epsilon_N; X)) \sim (F(X; \theta_1), \ldots, F(X; \theta_N))$$

Trying the following approaches:

- MMD for probability matching:
  CGM learns to model the predictive distribution.
- Mapping each posterior sampled model to a learned prior in the CGM:
  Based on Wasserstein Auto Encoder, each posterior is mapped to a Normal distribution, which ϵ is sampled from

Experiments and Results

Experiments

Metrics used for comparison:

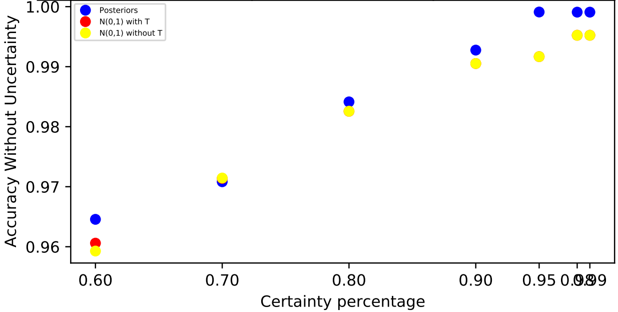
- Accuracy
- Accuracy without uncertain examples
- Confidence on wrong predictions
- Calibration
- Uncertainty count

The following results are on MNIST dataset and a two layer neural network.

Accuracy and Accuracy Filtered

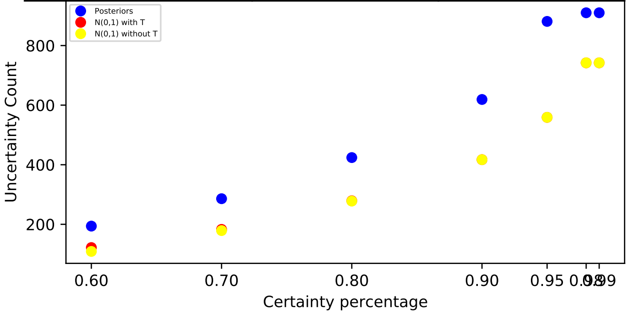
	Posteriors	N(0,1) with T
Accuracy	0.9204999804496765	0.9350000023841858

	Posteriors	N(0,1) with T	N(0,1) without T
0.6	0.9645625948905945	0.9605963826179504	0.959280788898468
0.7	0.9708284735679626	0.971381425857544	0.9714442491531372
0.8	0.9841370582580566	0.9825682640075684	0.9825783967971802
0.9	0.9927588701248169	0.9905242919921875	0.9905242919921875
0.95	0.9991063475608826	0.9916724562644958	0.9916724562644958
0.98	0.9990825653076172	0.995230495929718	0.995230495929718
0.99	0.9990825653076172	0.995230495929718	0.995230495929718



Uncertainty Count

	Posteriors	N(0,1) with T	N(0,1) without T
0.6	194	122	109
0.7	286	183	179
0.8	424	279	278
0.9	619	417	417
0.95	881	559	559
0.98	910	742	742
0.99	910	742	742



Calibration

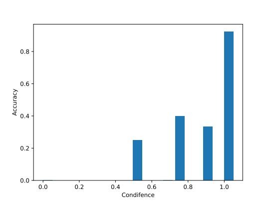


Figure: Generator - Before T

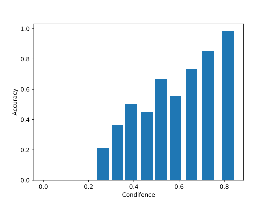


Figure: Generator - After T

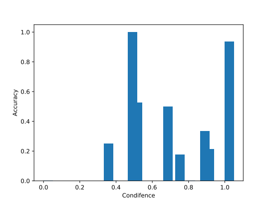


Figure: Posteriors - Before T

Questions?