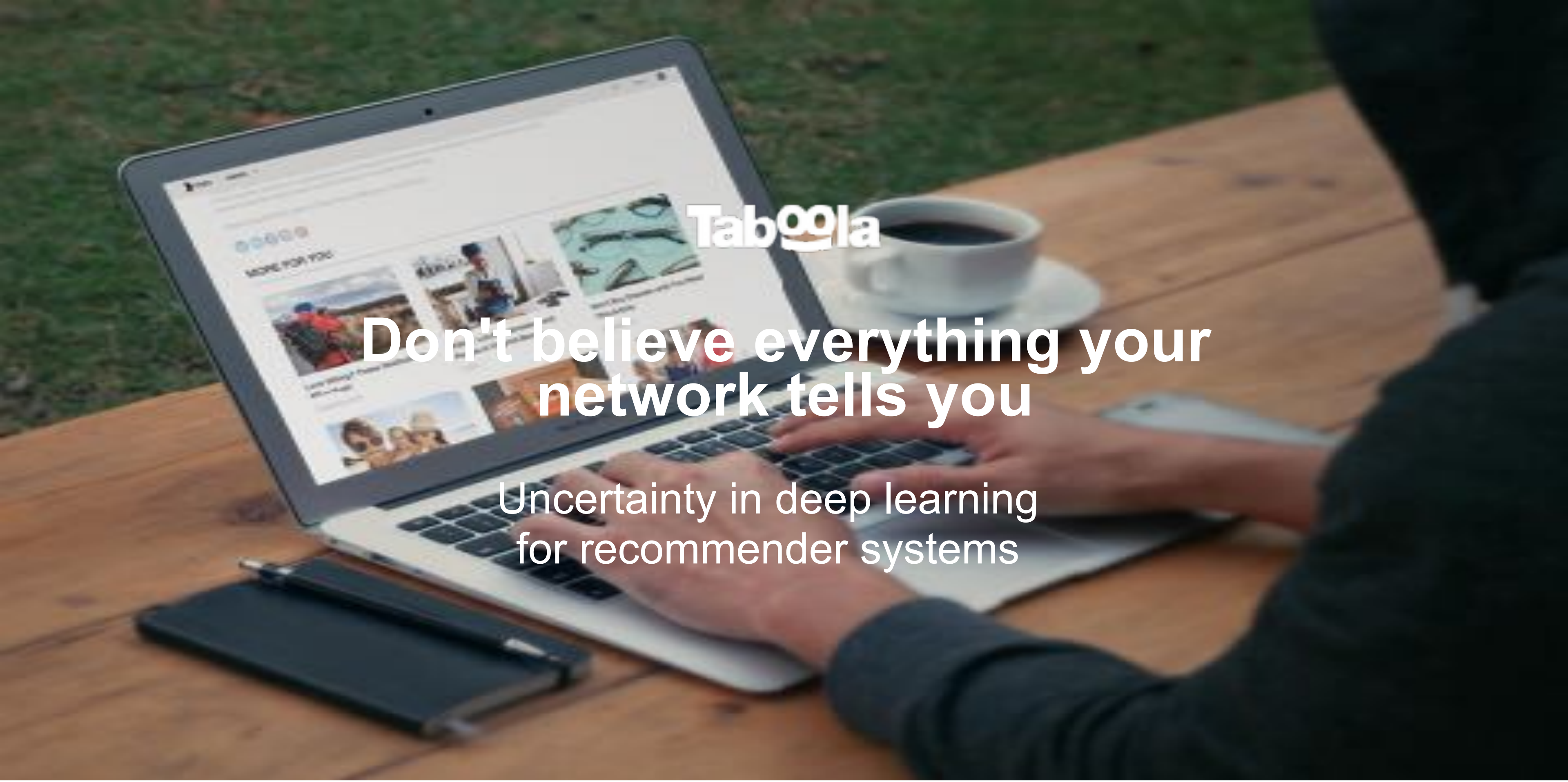


Don't believe everything your network tells you

Uncertainty in deep learning for recommender systems

Gil Chamiel, Director of Data Science and Algorithms Engineering

October 2017

Outline

- Content Discover and why go deep?
- Uncertainty in Deep Learning – why is it important?
- Data Uncertainty
  - Capturing the true variance of your prediction
- Model Uncertainty
  - Shedding light on where the model lacks data

Content Discovery

~1M Possible Recommendations

Rank N recommendations by *CTR* * *CPC*

Context Metadata

Location Region-based Information

User Historical Data

Machine Learning + Discovery = Hard

Implicit Feedback

Data is Very Sparse

No Human Baseline

World is Ever-changing

Rank N recommendations by CTR * CPC

“Walmart cameras captured these hilarious photos”

“15 rarely seen WW2 Photos Discovered”

Why Go Deep for Discovery?

- Cold start is a huge issue
- Many hard sub problems
  - Language modeling
  - Image classification
  - User Profiling
- There are many complex relations

Why is it so important?

Uncertainty in Deep Learning

Exploration/Exploitation in Recommender Systems

Best Performing Recommendations

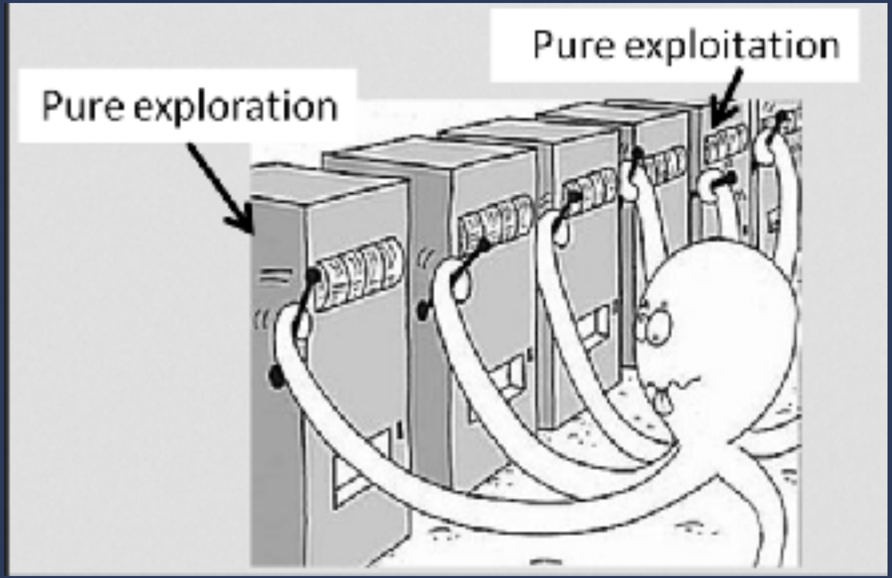
Search for new stars

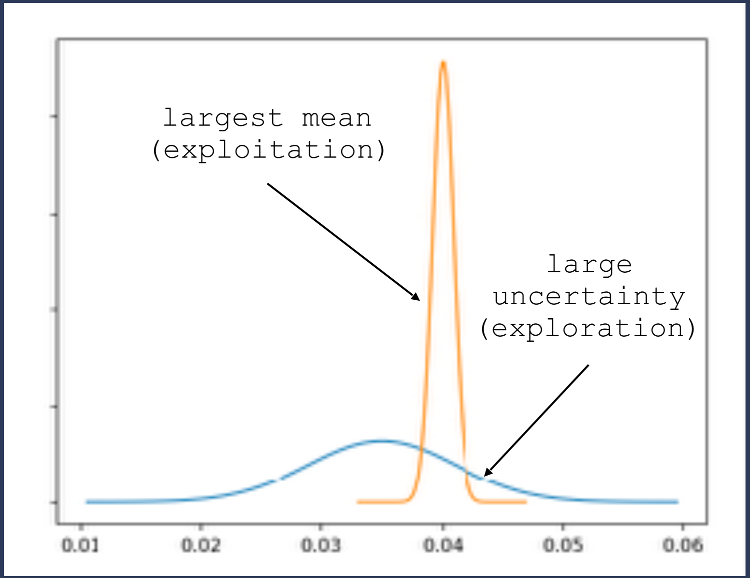
Add new Information

Fight selection bias

Exploration at random doesn't work

Multi-arm Bandits





- ✓ Thompson Sampling
- ✓ Mean +/- k*Std

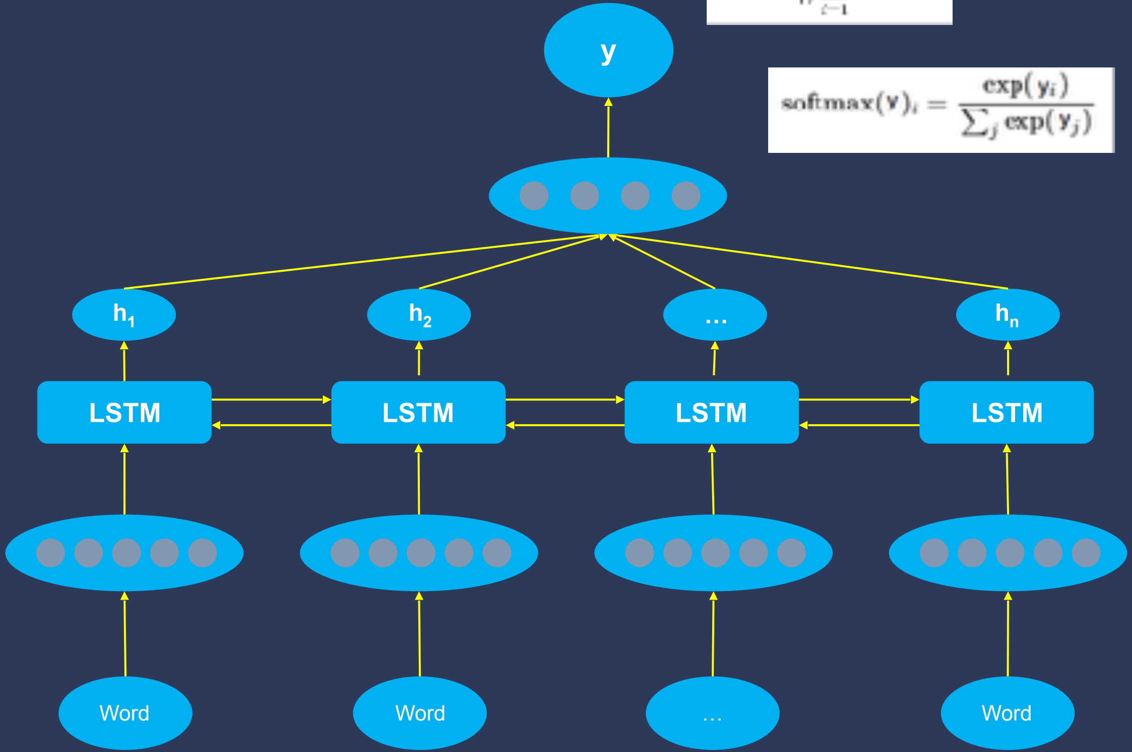
Capturing Model Blind spots is Crucial

CAPTURING DATA UNCERTAINTY

Uncertainty in Deep Learning: not out of the box

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$\mathrm{softmax}(\mathbf{y})_i = \frac{\exp(\mathbf{y}_i)}{\sum_j \exp(\mathbf{y}_j)}$$

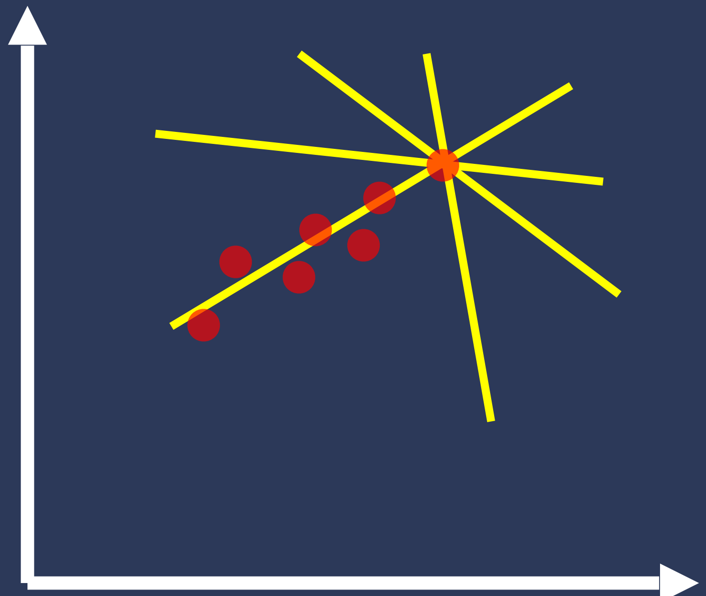


Two Types of Uncertainty

Model Uncertainty

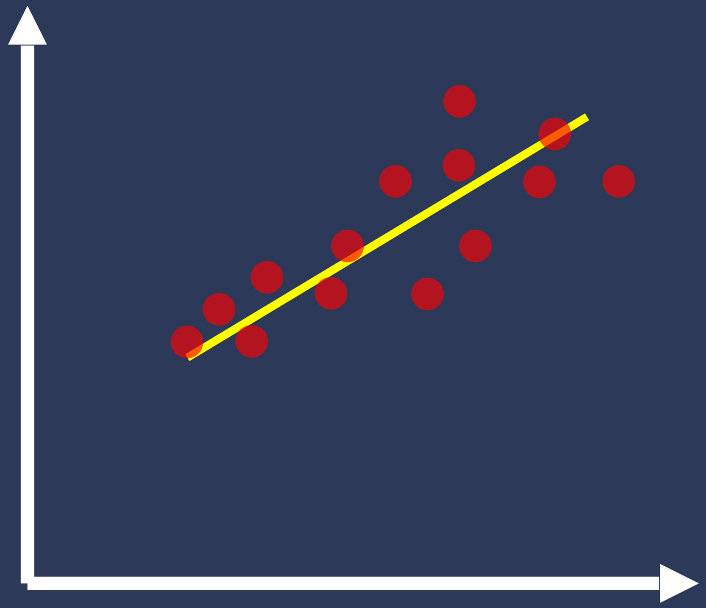
Data Uncertainty

Model Uncertainty



More Data Please!

Data Uncertainty

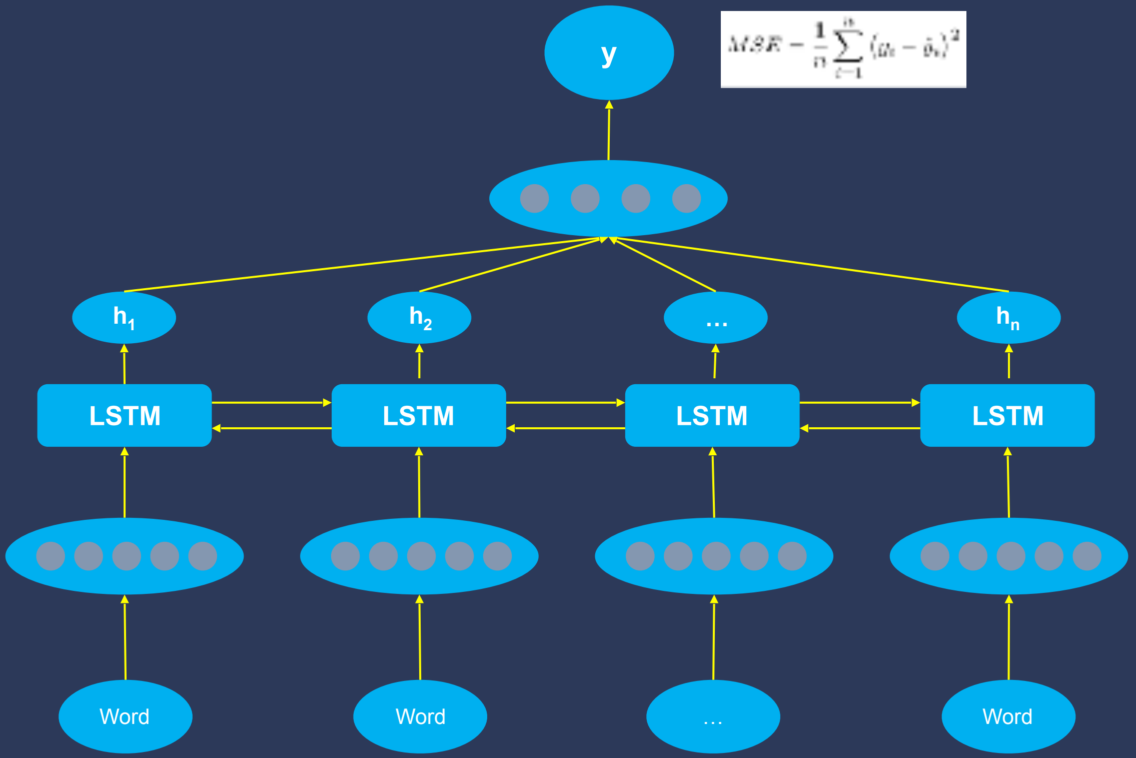


More Data Won't Help – I want to know how good my predictions are

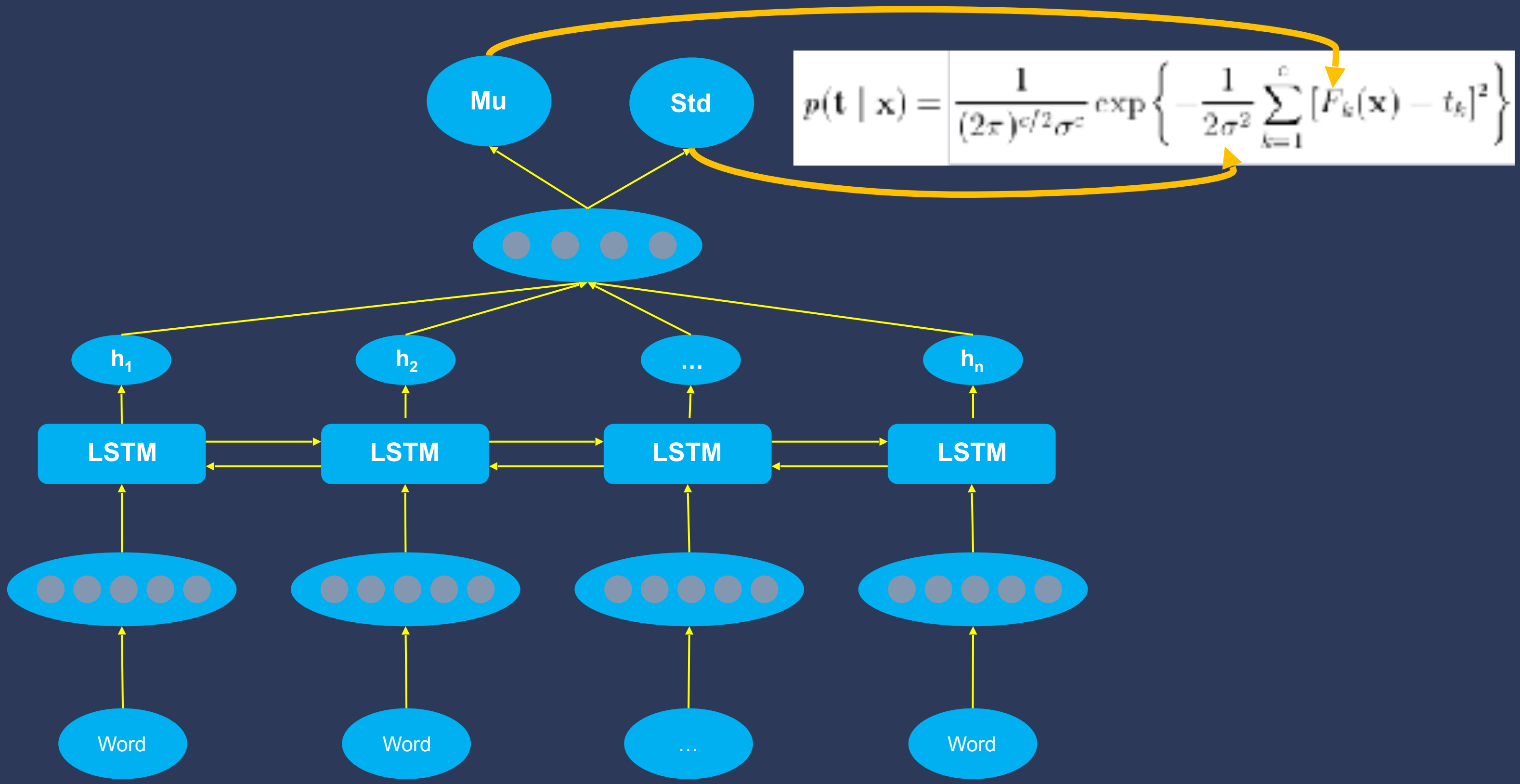
Know what you don't know

Capturing Data Uncertainty

Likelihood as loss



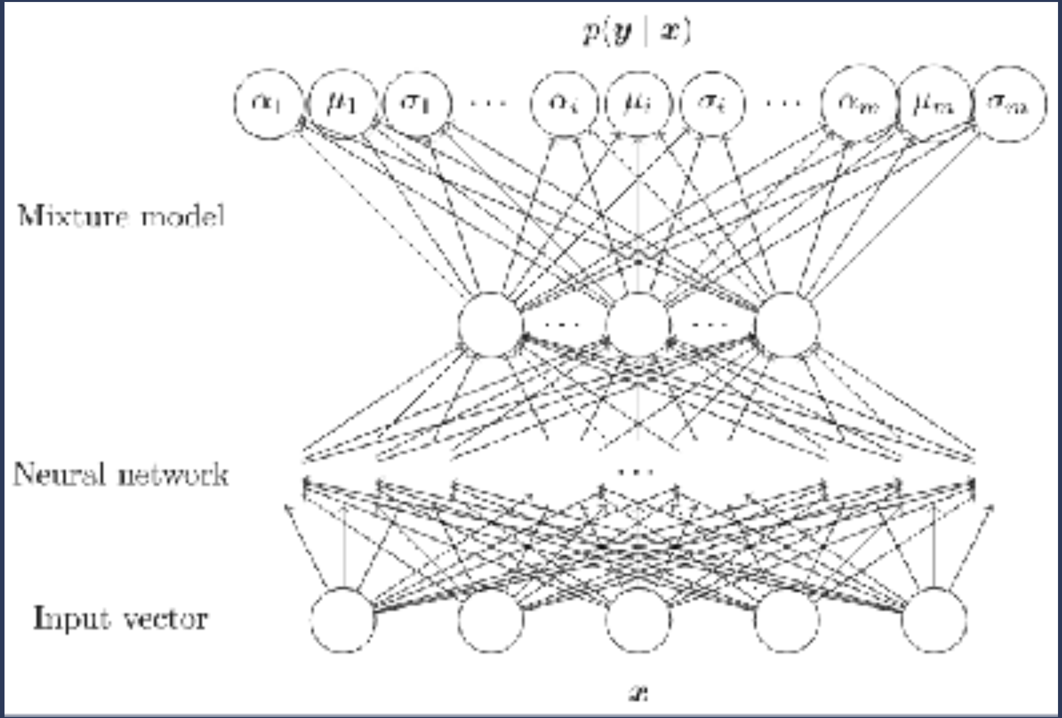
Likelihood as loss



$$p(\mathbf{t} \mid \mathbf{x}) = \frac{1}{(2\pi)^{c/2}\sigma^{c}} \exp\left\{ -\frac{1}{2\sigma^{2}} \sum_{k=1}^{c} [F_k(\mathbf{x}) - t_k]^{2} \right\}$$

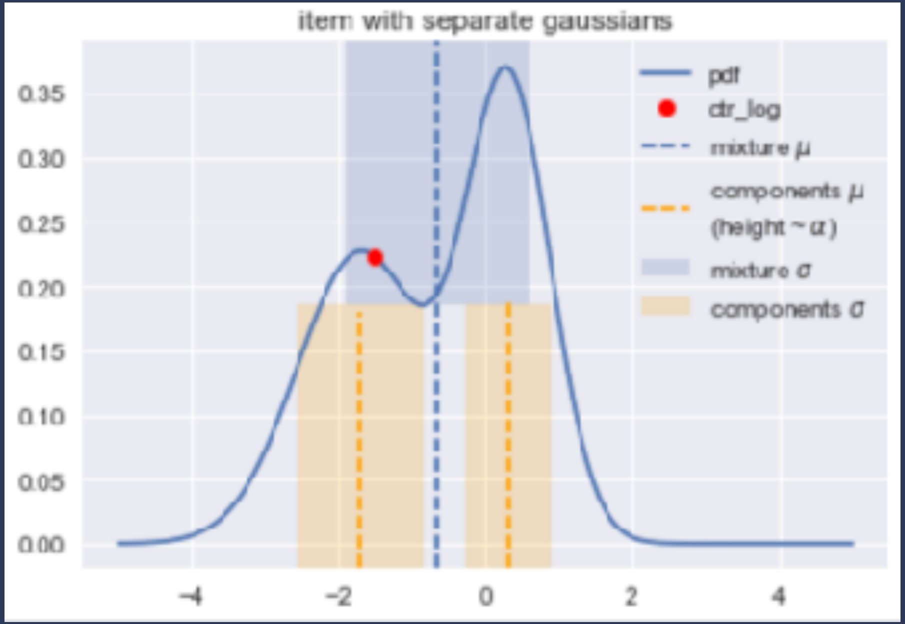
Mixture Density Network

$$p(\mathbf{t} \mid \mathbf{x}) = \sum_{i=1}^{m} \alpha_i(\mathbf{x})\phi_i(\mathbf{t} \mid \mathbf{x})$$

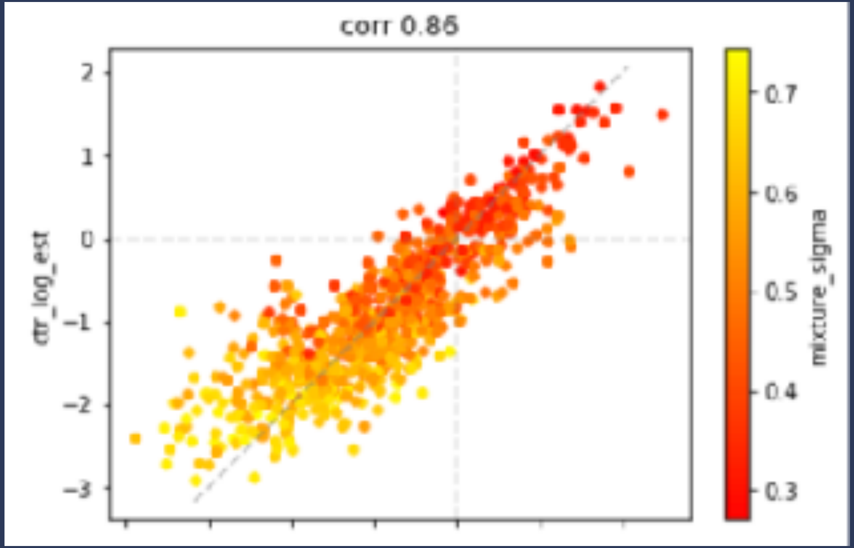


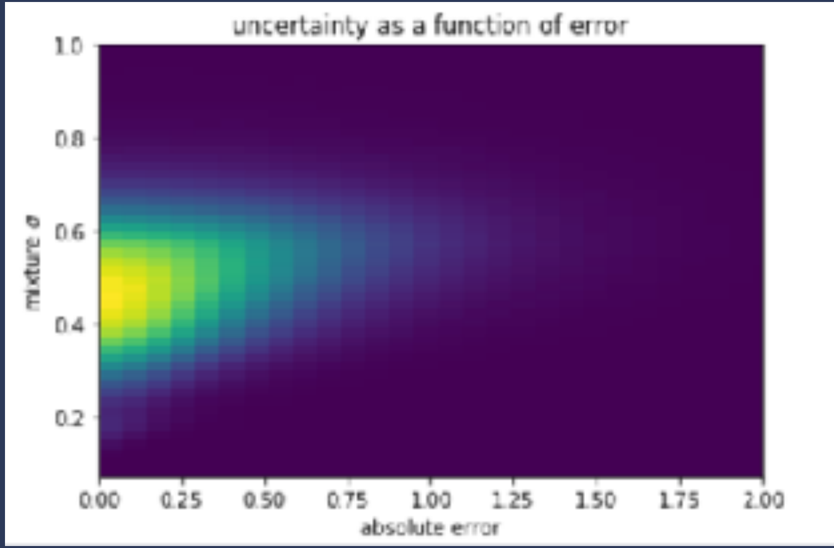
Christopher M. Bishop, Mixture density networks (1994)

Capturing Data Uncertainty

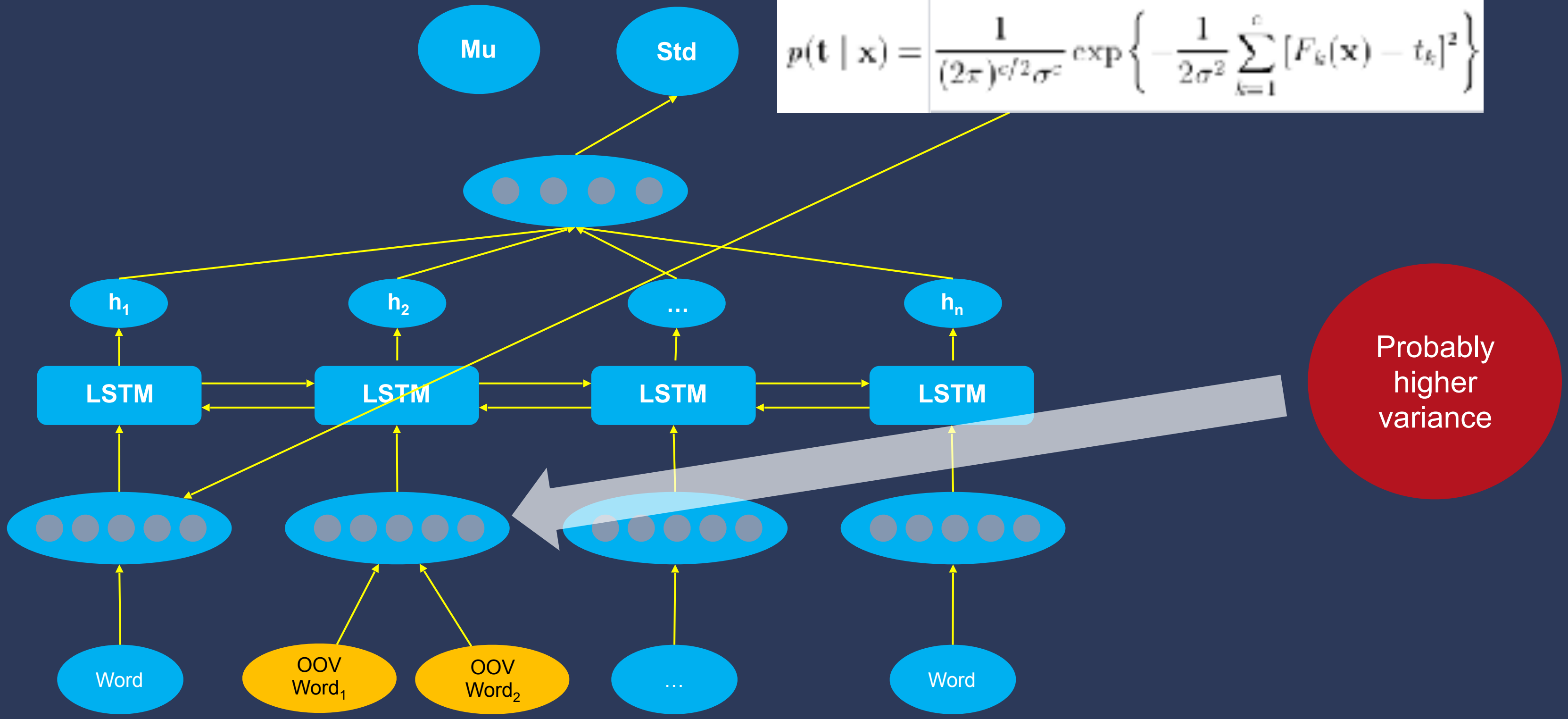


Data Uncertainty and Training Error



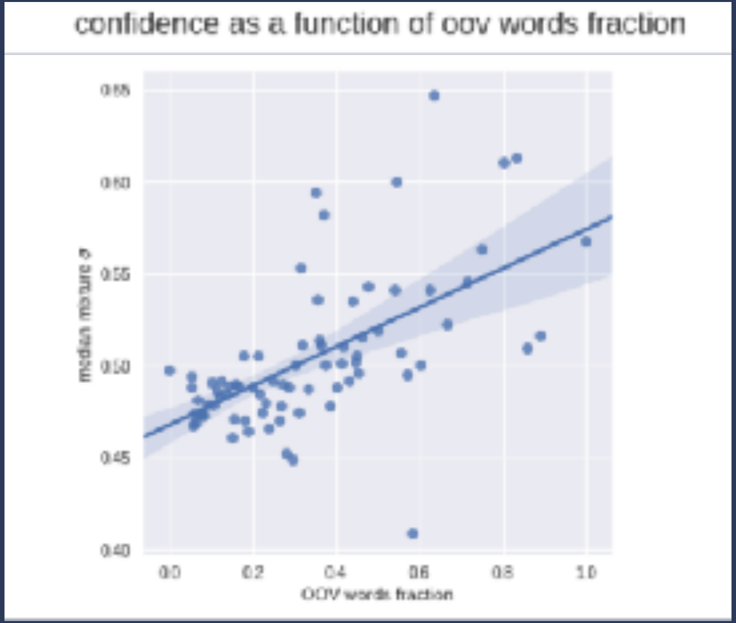


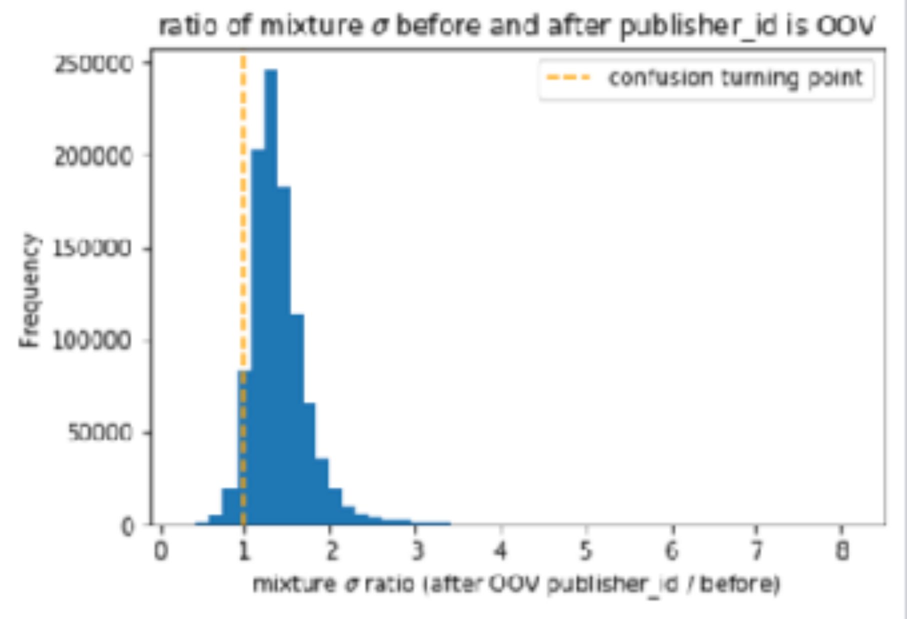
Data Uncertainty and OOV



$$p(\mathbf{t}\mid\mathbf{x}) = \frac{1}{(2\pi)^{c/2}\sigma^{c}}\exp\left\{-\frac{1}{2\sigma^{2}}\sum_{k=1}^{c}[F_k(\mathbf{x})-t_k]^{2}\right\}$$

Data Uncertainty and OOV

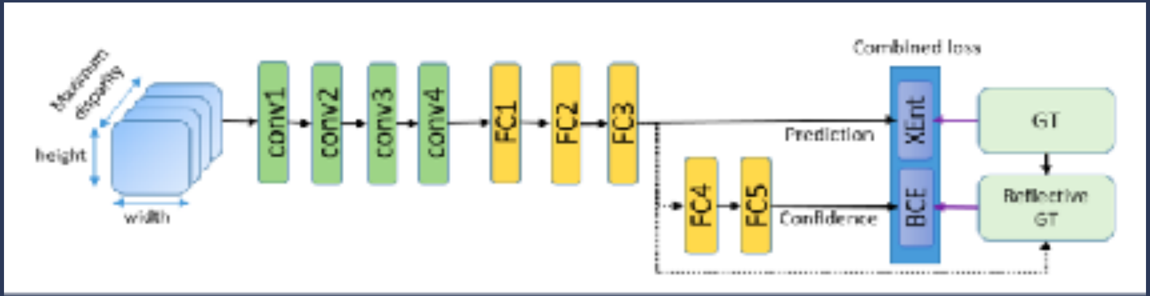




Useful as a debugging tool!

What about classification?

- Assume a binomial distribution with Beta prior (natural for binary classification)
- Reflective loss



Shaked & Wolf, Improved Stereo Matching with Constant Highway Networks and Reflective Confidence Learning (2016)

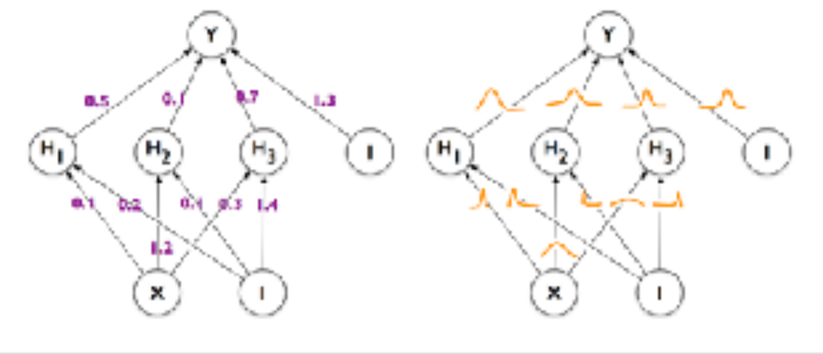
Know when you can get better

Capturing Model Uncertainty

Bayesian Neural Networks and Variational Inference

- Which function generated our data?
- Bayesian approach:
  - Assume some prior distribution over the space of possible functions
  - Look for the posterior distribution given your data
- Analytical solution is intractable at inference time
- Approximation is needed => Variational Inference

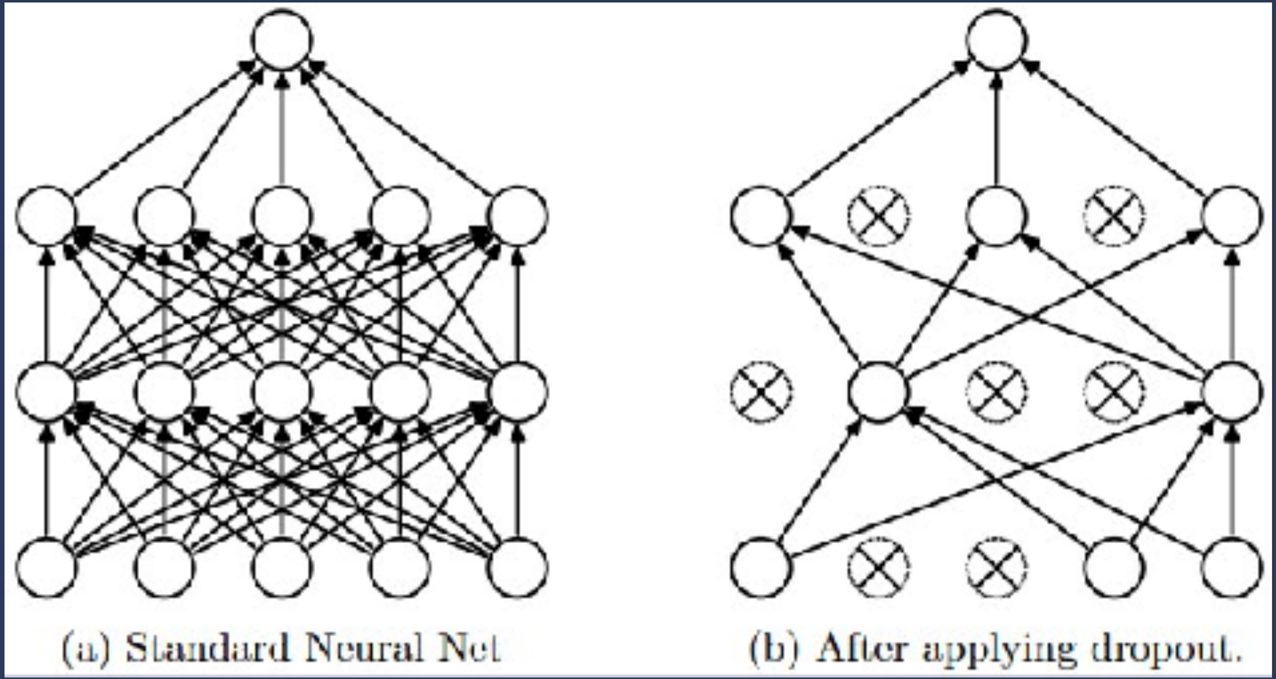
$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{Y}) = \int p(\mathbf{y}^*|\mathbf{f}^*)p(\mathbf{f}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{Y})d\mathbf{f}^*$$



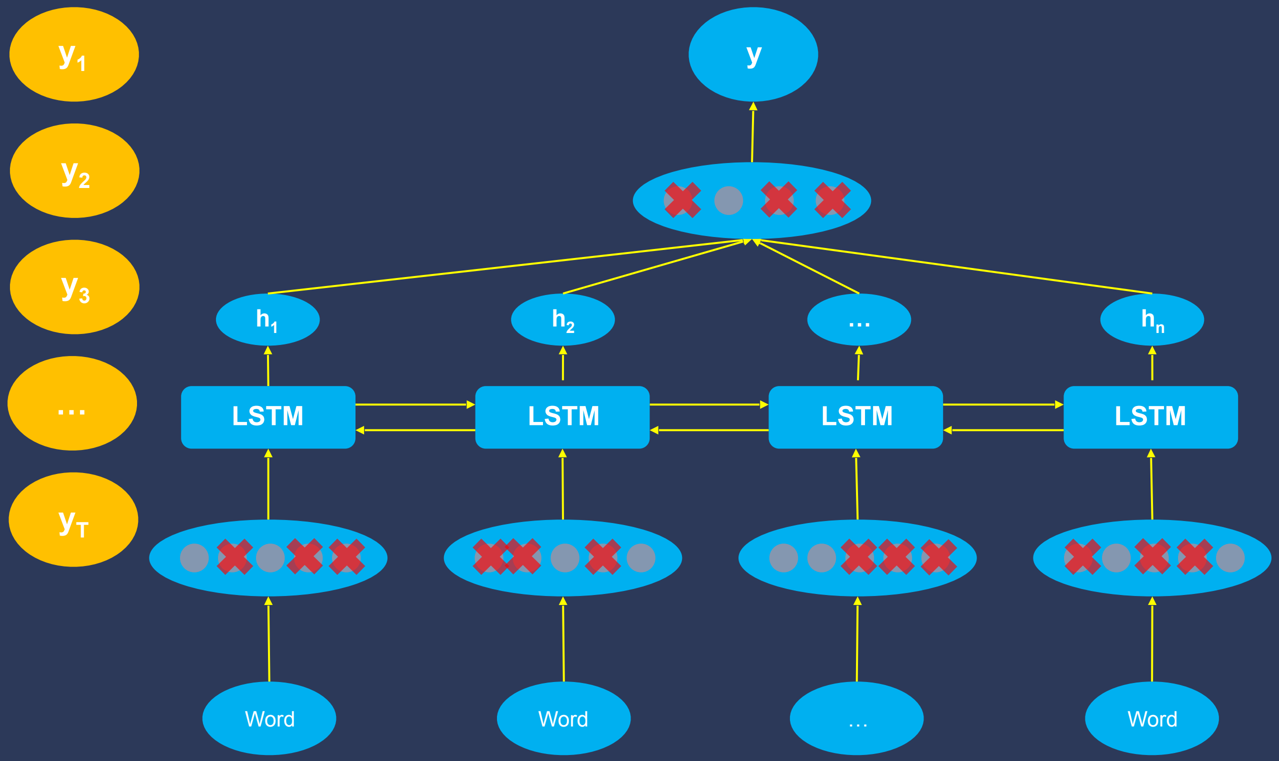
Yarin Gal, “what my deep model doesn’t know”

Blundell et al, Weight Uncertainty in Neural Networks (2015)

Recap: Dropouts – a regularization technique



Dropout variational inference as Bayesian Approximation



$$\mathbb{E}(\mathbf{y}^*) \approx \frac{1}{T}\sum_{t=1}^{T} \hat{\mathbf{y}}_t^*(\mathbf{x}^*)$$

$$\mathrm{Var}\left(\mathbf{y}^*\right) \approx \tau^{-1}\mathbf{I}_D + \frac{1}{T}\sum_{t=1}^{T} \hat{\mathbf{y}}_t^*(\mathbf{x}^*)^T \hat{\mathbf{y}}_t^*(\mathbf{x}^*) - \mathbb{E}(\mathbf{y}^*)^T \mathbb{E}(\mathbf{y}^*)$$

gal & ghahramani, Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning (2015)

Uncertainty as a function of amount of data

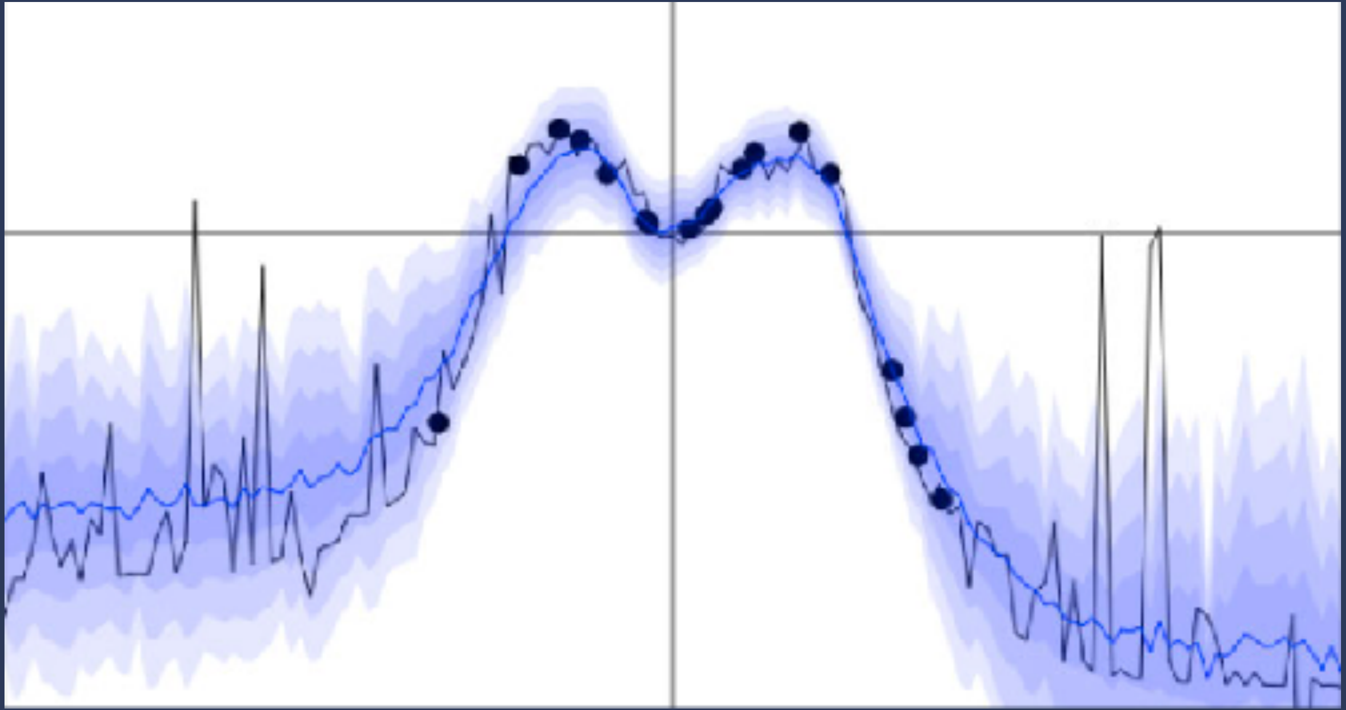
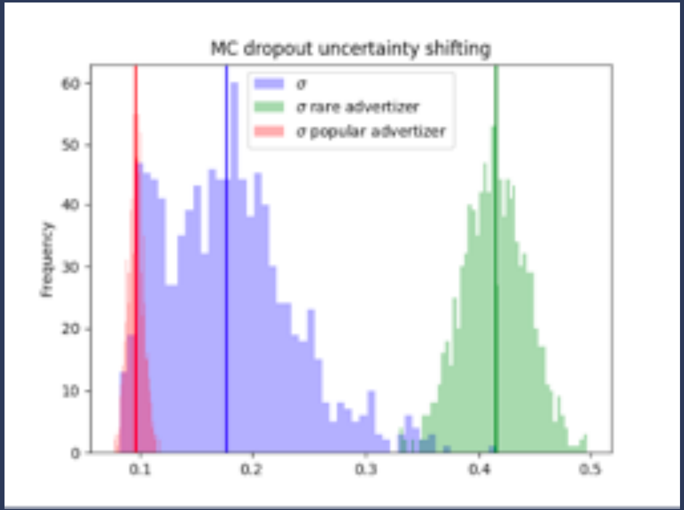
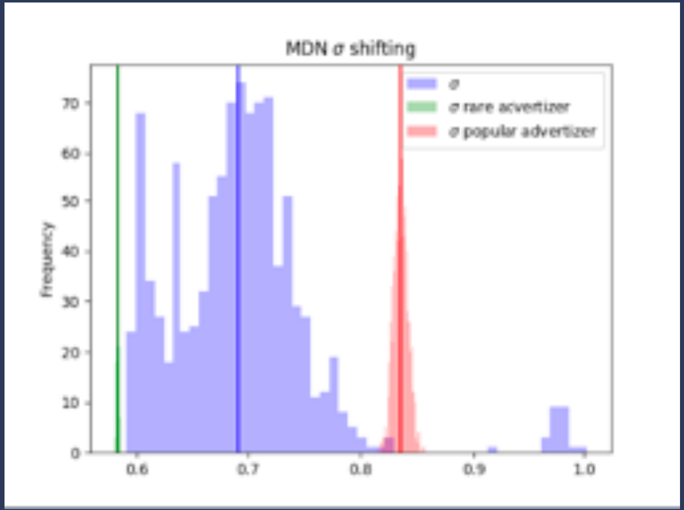


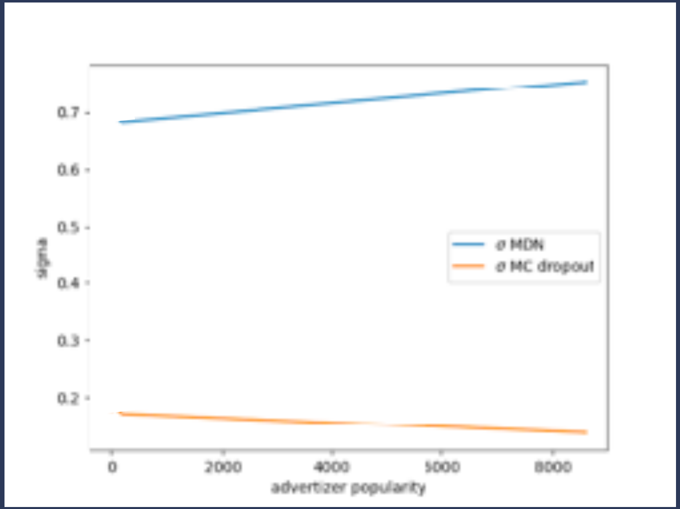
Image: Yarin Gal, “what my deep model doesn’t know”

Uncertainty as a function of amount of data





Uncertainty as a function of amount of data



Summary

- Two types of uncertainty: model and data
- Mixture Density Networks
  - Captures the true variance of your prediction
- Monte-Carlo Dropouts Variational Inference
  - Sheds light on where the model lacks data

- Also interesting: uncertainty due to **measurement noise**

Thank You