# Welcome and Introduction

Philip Schulz
https://vitutorial.github.io
https://github.com/vitutorial/VITutorial

# About me . . .

## Philip Schulz

- Applied Scientist at Amazon
- VI, Machine Translation, Bayesian Models

# What is a probabilistic model?

A probabilistic model predicts **possible** outcomes of an experiment.
Most modern machine learning models are probabilistic.

# What is a probabilistic model?

A probabilistic model predicts **possible** outcomes of an experiment.
Most modern machine learning models are probabilistic.

## Maximum Likelihood

$$\max_{\theta} p(x|\theta)$$

# Two Machine Learning Paradigms

Supervised problems: "learn a distribution over observed data"

- sentences in natural language, images, videos, ...

Unsupervised problems: "learn a distribution over observed and unobserved data"

- sentences in natural language + parse trees, images + bounding boxes ...

# What are the benefits of probabilistic models?

Probabilistic models allow to incorporate assumptions through:

- the choice of distribution
- the way that distribution uses side information
- stipulate unobserved data

# What are the benefits of probabilistic models?

Probabilistic models allow to incorporate assumptions through:

- the choice of distribution
- the way that distribution uses side information
- stipulate unobserved data

They return a distribution over outcomes.

# What are the benefits of probabilistic models?

- Can generate data (generative models)

# What are the benefits of probabilistic models?

- Can generate data (generative models)
- Allows to model unobserved data
  - $\int p(x|z,y)p(z|y)\mathrm{d}z$ can be easier than $p(x|y)$
  - Can reduce number of parameters
  - Provides explanation and can suggest improvements

# What are the benefits of probabilistic models?

- Can generate data (generative models)
- Allows to model unobserved data
  - $\int p(x|z,y)p(z|y)\mathrm{d}z$ can be easier than $p(x|y)$
  - Can reduce number of parameters
  - Provides explanation and can suggest improvements
- Informative to decision makers
  - Provides uncertainty estimates

# What are the benefits of probabilistic models?

We can get uncertainty estimates.

## Example: Binary classifier

$$\sigma\left(x^T\theta\right)$$

gives **one** distribution over outcomes.

A decision maker wants to know **how much** he can trust the classifier!

# What are the benefits of probabilistic models?

$$\sigma\left(x^T M\theta\right) p(M)$$

where $M$ is some matrix that modifies the classifier weights.

# What are the benefits of probabilistic models?

$$\sigma\left(x^T M\theta\right) p(M)$$

where $M$ is some matrix that modifies the classifier weights. This gives us many different distributions over outcomes!

# What are the benefits of probabilistic models?

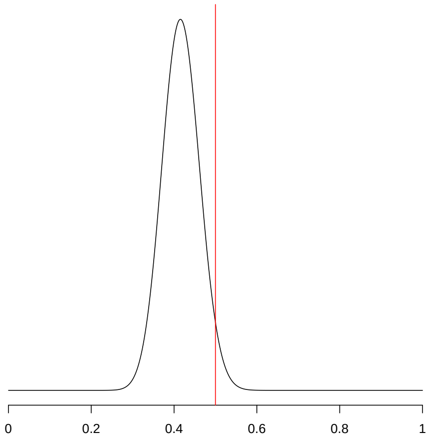
$$\sigma\left(x^T M\theta\right) p(M)$$

where $M$ is some matrix that modifies the classifier weights. This gives us many different distributions over outcomes!

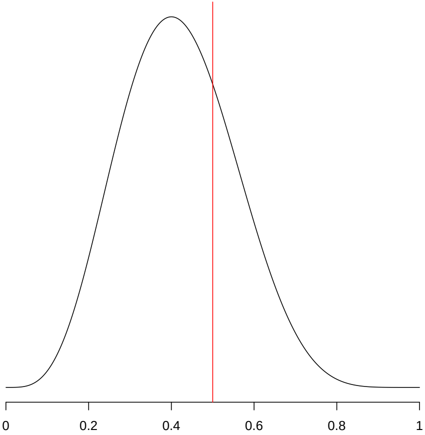
## Rule of Thumb

If the different distributions are similar, the classifier can be trusted.
If they are dissimilar, further context information is needed.

# What are the benefits of probabilistic models?



# What are the benefits of probabilistic models?



# Deep Generative Models

Naturally, one would like to combine the advantages of probabilistic models and neural nets. So why not have a neural net with latent variables?
Short answer: backpropagation breaks!

# Deep Generative Models

## Supervised MLE

$$\max_{\phi} p(x|\phi, y) \implies \max_{\theta} p\left(x|\mathrm{NN}_{\theta}(y)\right)$$

- $\phi = (\mu, \sigma)$, $\phi = (\theta_1, \ldots, \theta_K)$, ...

# Deep Generative Models

## Supervised MLE

$$\max_{\phi} p(x|\phi, y) \implies \max_{\theta} p\left(x|\mathrm{NN}_{\theta}(y)\right)$$

- $\phi = (\mu, \sigma)$, $\phi = (\theta_1, \ldots, \theta_K)$, ...

## Unsupervised MLE

$$\max_{\phi} p(x|\phi, y, z)p(z|y, \phi) \implies \max_{\theta} p\left(x|\mathrm{NN}_{\theta}(z, y)\right) p\left(z|\mathrm{NN}_{\theta}(y)\right)$$

- $\phi = (\mu_x, \sigma_x, \mu_z, \sigma_z)$, ...

# Unsupervised Learning

Exact gradient is intractable

$$\nabla_\theta \log p(x|\theta)$$

# Unsupervised Learning

Exact gradient is intractable

$$\nabla_\theta \log p(x|\theta) = \nabla_\theta \log \int p(x, z|\theta)\,\mathrm{d}z$$

# Unsupervised Learning

## Exact gradient is intractable

$$
\begin{aligned}
\nabla_\theta \log p(x|\theta) &= \nabla_\theta \log \int p(x, z|\theta)\,\mathrm{d}z \\
&= \underbrace{\frac{1}{\int p(x, z|\theta)\,\mathrm{d}z} \int \nabla_\theta p(x, z|\theta)\,\mathrm{d}z}_{\text{chain rule}}
\end{aligned}
$$

# Unsupervised Learning

## Exact gradient is intractable

$$
\begin{aligned}
\nabla_\theta \log p(x|\theta) &= \nabla_\theta \log \int p(x, z|\theta)\,\mathrm{d}z \\
&= \underbrace{\frac{1}{\int p(x, z|\theta)\,\mathrm{d}z} \int \nabla_\theta p(x, z|\theta)\,\mathrm{d}z}_{\text{chain rule}} \\
&= \frac{1}{p(x|\theta)} \int \underbrace{p(x, z|\theta) \nabla_\theta \log p(x, z|\theta)}_{\text{log-identity for derivatives}}\,\mathrm{d}z
\end{aligned}
$$

# Unsupervised Learning

Exact gradient is intractable

$$\begin{aligned}
\nabla_\theta \log p(x|\theta) &= \nabla_\theta \log \int p(x,z|\theta)\,\mathrm{d}z \\
&= \underbrace{\frac{1}{\int p(x,z|\theta)\,\mathrm{d}z} \int \nabla_\theta p(x,z|\theta)\,\mathrm{d}z}_{\text{chain rule}} \\
&= \frac{1}{p(x|\theta)} \int \underbrace{p(x,z|\theta) \nabla_\theta \log p(x,z|\theta)}_{\text{log-identity for derivatives}}\,\mathrm{d}z \\
&= \int p(z|x,\theta) \nabla_\theta \log p(x,z|\theta)\,\mathrm{d}z
\end{aligned}$$

# Unsupervised Learning

## Exact gradient is intractable

$$
\begin{aligned}
\nabla_\theta \log p(x|\theta) &= \nabla_\theta \log \int p(x,z|\theta)\,\mathrm{d}z \\
&= \underbrace{\frac{1}{\int p(x,z|\theta)\,\mathrm{d}z} \int \nabla_\theta p(x,z|\theta)\,\mathrm{d}z}_{\text{chain rule}} \\
&= \frac{1}{p(x|\theta)} \int \underbrace{p(x,z|\theta)\nabla_\theta \log p(x,z|\theta)}_{\text{log-identity for derivatives}}\,\mathrm{d}z \\
&= \int p(z|x,\theta)\nabla_\theta \log p(x,z|\theta)\,\mathrm{d}z \\
&= \mathbb{E}_{p(z|x,\theta)}\left[\nabla_\theta \log p(x,Z|\theta)\right]
\end{aligned}
$$

# Variational Inference

Computing the posterior distribution $p(z|x, \theta)$ is hard. In VI we will optimize an auxiliary distribution $q(z|x, \lambda)$ to approximate the exact posterior.

# What are you getting out of this today?

As we progress we will

- develop a shared vocabulary to talk about generative models powered by NNs
- derive crucial results step by step

# What are you getting out of this today?

As we progress we will

- develop a shared vocabulary to talk about generative models powered by NNs
- derive crucial results step by step

Goal

- you should be able to navigate through fresh literature
- and start combining probabilistic models and NNs