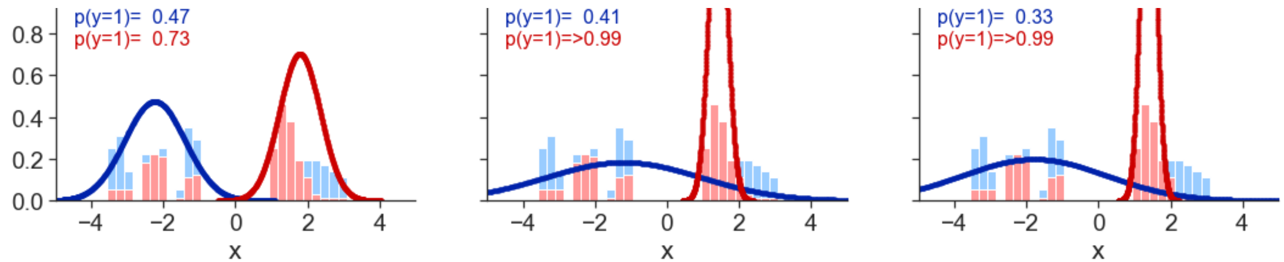


"I can't believe supervision for latent variable models is not better:"

The Case for Prediction Constrained training

Michael C. Hughes
Assistant Professor of Computer Science
Tufts University

joint work with
Erik Sudderth, Gabe Hope (UC Irvine)
Madina Abdrakhmanova, Xiaoyin Chen (UC Irvine)
Finale Doshi-Velez & Joe Futoma (Harvard)

slides / papers / code
www.michaelchughes.com

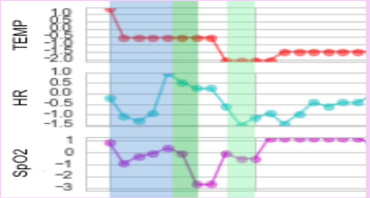
Motivation

Given: dataset $\mathcal{D}$ with many examples of:

- Features x
- Label y

<u>Psychiatry application</u>
x : patient's health records
y : successful medication

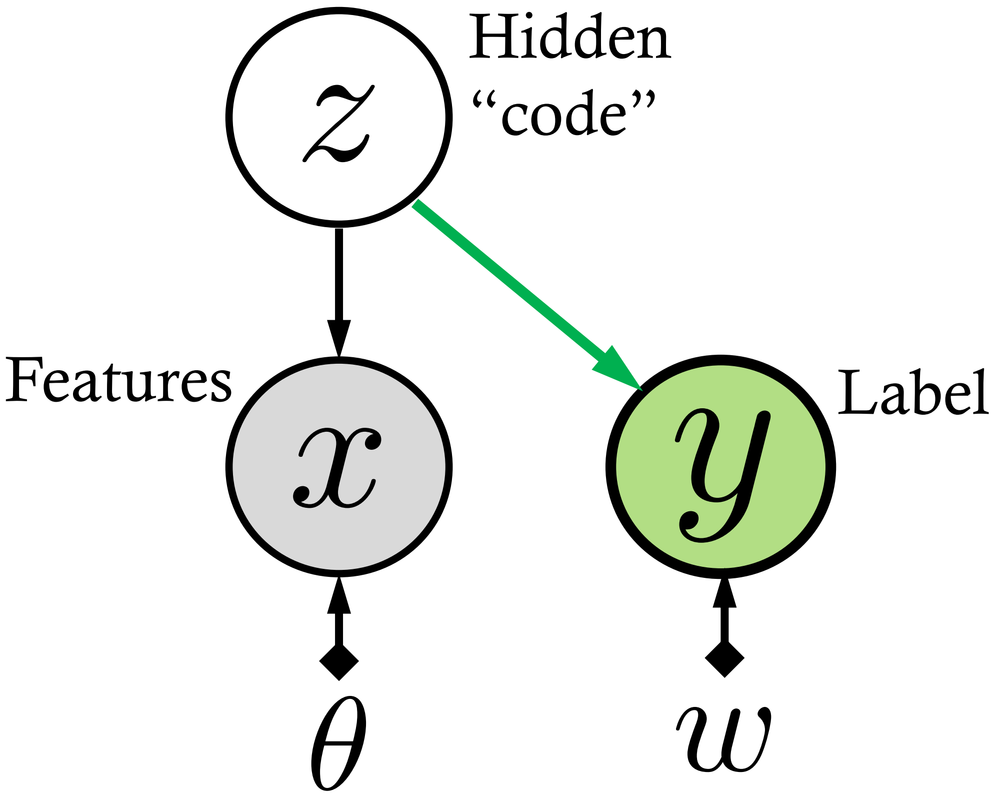
<u>Intensive Care application</u>
x : time-series of vitals
y : need for ventilator



Goals:

- Most important: $p(y|x)$
  - Predict labels from features well at test time
- Also important: $p(x, y)$
  - Predict even when missing features
  - Train even if only some examples are labeled
  - Offer interpretable structure

Latent variable models (LVMs) with **supervision**



Prior $p(z)$

Feature likelihood $p_\theta(x|z)$

Label likelihood $p_w(y|z)$

Vast literature of unsupervised LVMs. Could add supervision to any of them. (Many have.)

"Shallow" LVMs
- Probabilistic PCA
- Mixture models
- Topic models
- Hidden markov models
- Linear dynamical systems

"Deep" LVMs
- Variational Autoencoders
- Deep GMMs
- Deep topic models
- Recurrent SLDS
- … and many more

I want to believe …

Why use Supervised LVMs? (deep or shallow)

Goals:

- Most important: $p(y|x)$
  - [ ] Predict labels from features well at test time
- Also important: $p(x, y)$
  - [✓] Predict even when missing features
  - [✓] Train even if only some examples are labeled
  - [✓] Offer interpretable structure

I want to believe …

Why use Supervised LVMs? (deep or shallow)

Goals:

- Most important: $p(y|x)$
  - [?] Predict labels from features well at test time
- Also important: $p(x, y)$
  - [✓] Predict even when missing features
  - [✓] Train even if only some examples are labeled
  - [✓] Offer interpretable structure

Key question: are predictions good enough?

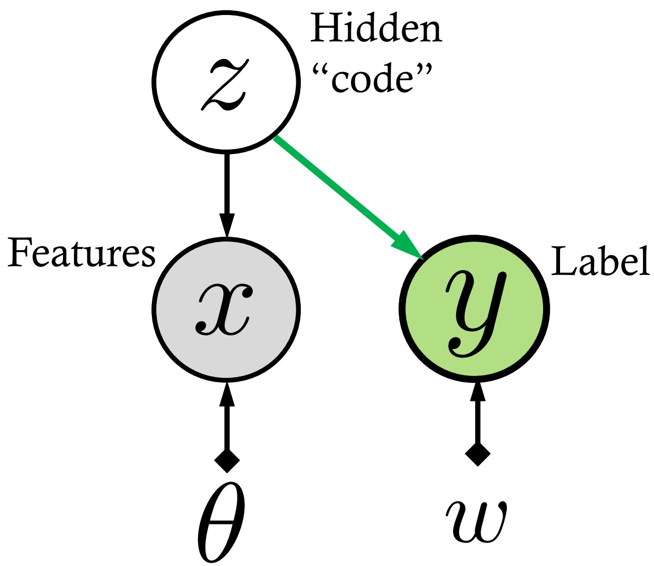
… but I can't believe it is not better

Claim: Standard ways of supervising LVMs deliver *little added value* when predicting labels given features, especially on real data.

Typically, when all methods have similar capacity, supervised LVMs are:

- **No better** than unsupervised baselines.
- **Inferior** to discriminative methods (if labeled data is abundant)

Latent variable models with **supervision**



Prior	$p(z)$
Feature likelihood	$p_\theta(x\|z)$
Label likelihood	$\boxed{p_w(y\|z)}$

How to train? Maximize (lower bound of) marginal likelihood

Feature
marginal likelihood:

$$p_\theta(x) = \int p_\theta(x|z)p(z)dz$$

Joint (Feature+Label)
marginal likelihood:

$$p_{\theta,w}(x,y) = \int \boxed{p_w(y|z)}p_\theta(x|z)p(z)dz$$

How to train a supervised LVM?

<u>(A) Maximize joint likelihood</u>

$$\max_{\theta, w} \sum_{x, y \in \mathcal{D}} \log p_{\theta, w}(x, y)$$

How to train a predictor based on **unsupervised** LVM?

<u>(B) Unsupervised-then-predict (2 stage)</u>

1. Train to maximize feature likelihood.

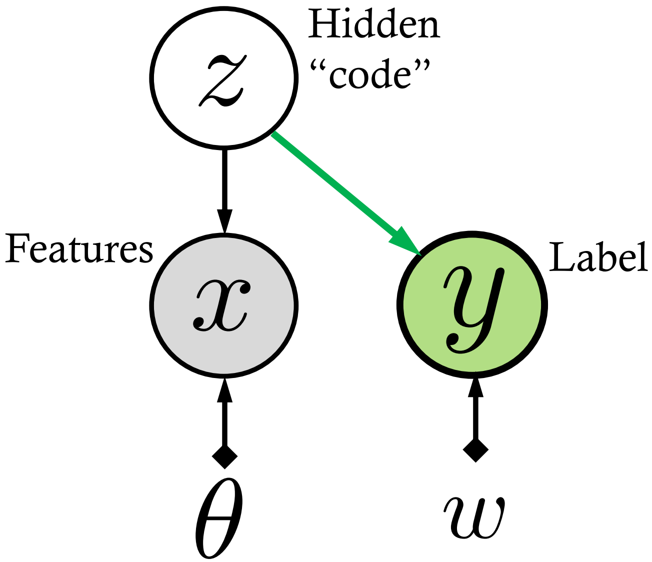
$$\max_{\theta} \sum_{x \in \mathcal{D}} \log p_{\theta}(x)$$

2. Fit label-from-hidden predictor.

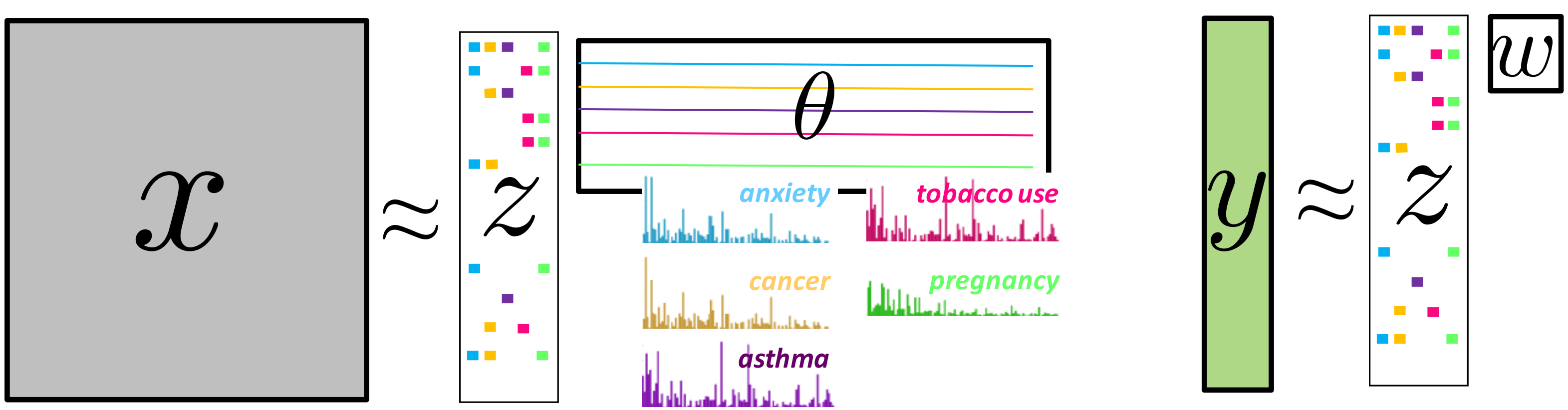
$$\max_{w} \sum_{x, y \in \mathcal{D}} \log p_{w}(y \mid \mathbb{E}_{p_{\theta}(z|x)}[z])$$

Example 1:
Supervised topic models for count data

Blei & McAuliffe (2010)

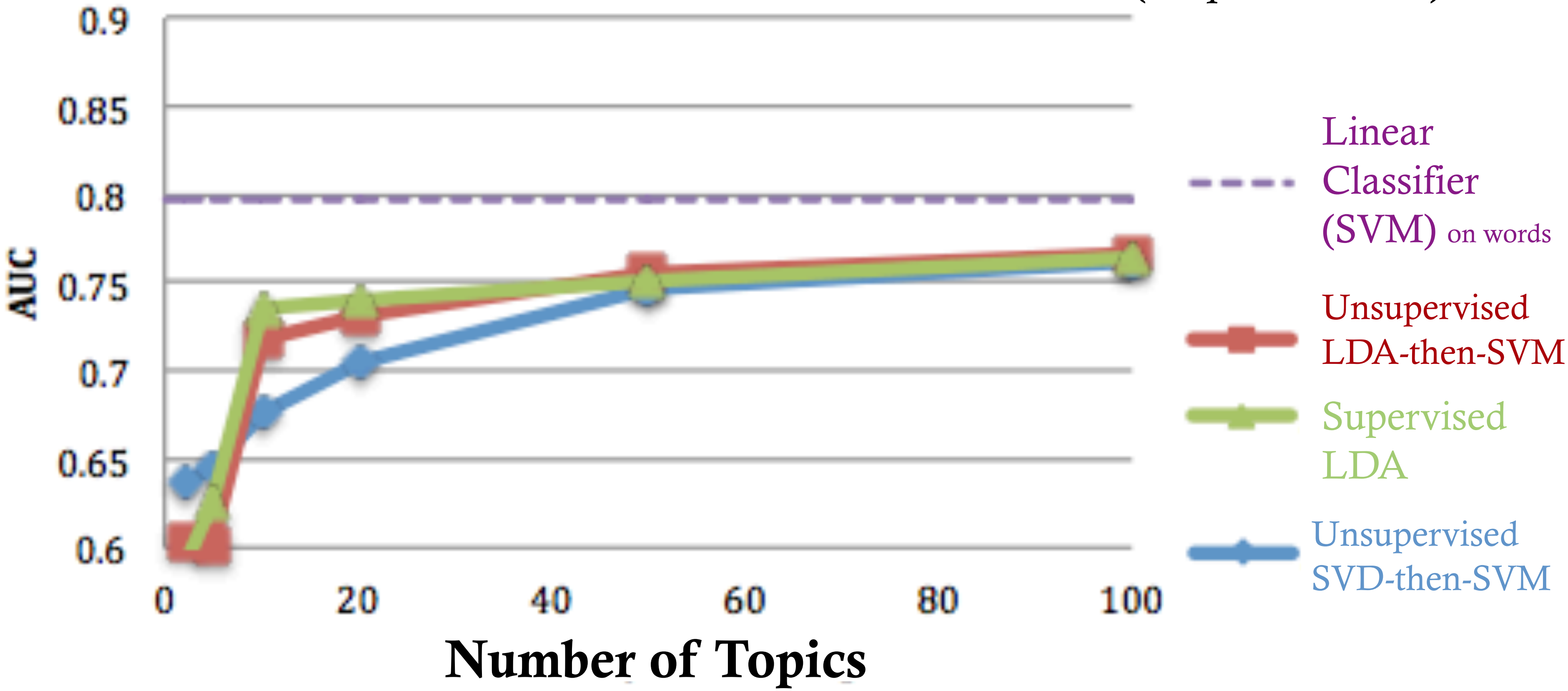


$$p(z) = \mathrm{Dir}(0.1, \ldots, 0.1)$$
$$p_\theta(x|z) = \mathrm{Mult}(\textstyle\sum_k z_k \theta_k)$$
$$p_w(y|z) = \mathrm{Bern}(\sigma(\textstyle\sum_k z_k w_k))$$



Supervised topic models predict *poorly*

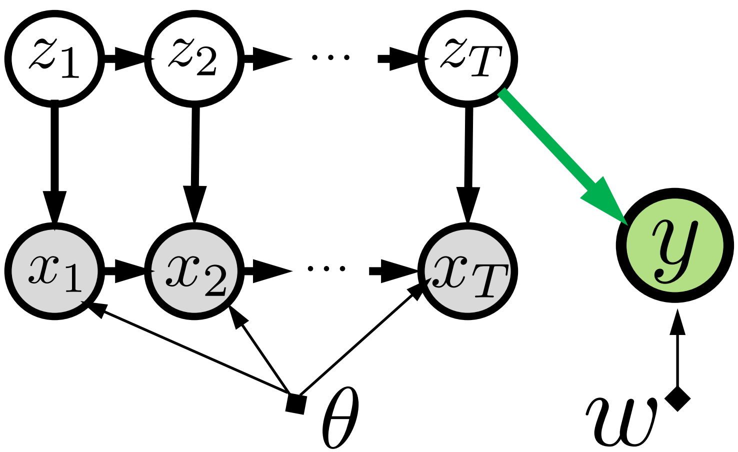
Task: Predict ICU Admission from Clinical Notes (*Halpern et al '12*)



Compared to methods with similar capacity, supervised LDA is:

- **No better** than unsupervised-LDA-then-predict
- **Inferior** to linear classifier of labels given word features

Example 2: Supervised Hidden Markov Models



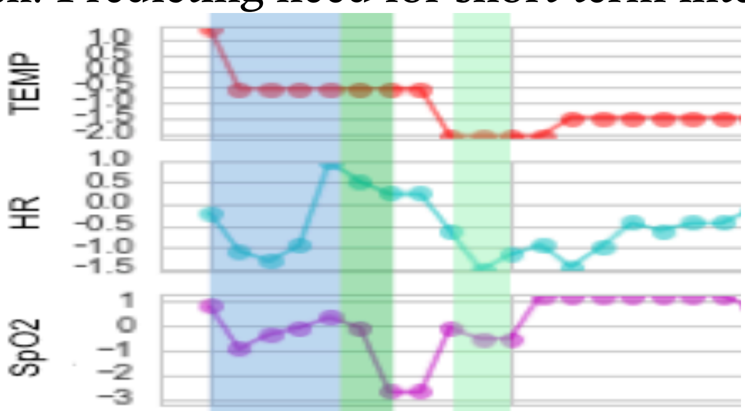
Sticky HMM with autoregressive likelihood

$$p(z_{1:T}) = p(z_1) \prod_{t=2}^{T} p(z_t | z_{t-1})$$

$$p_\theta(x_{1:T} | z_{1:T}) = \prod_{t=1}^{T} \mathcal{N}(x_t | A^\theta_{z_t} x_{t-1}, \Sigma^\theta_{z_t})$$

$$p_w(y | z_{1:T}) = \mathrm{Bern}(y | \sigma(w_{z_T}))$$

Task: Predicting need for short-term intervention in ICU from vital sign time series

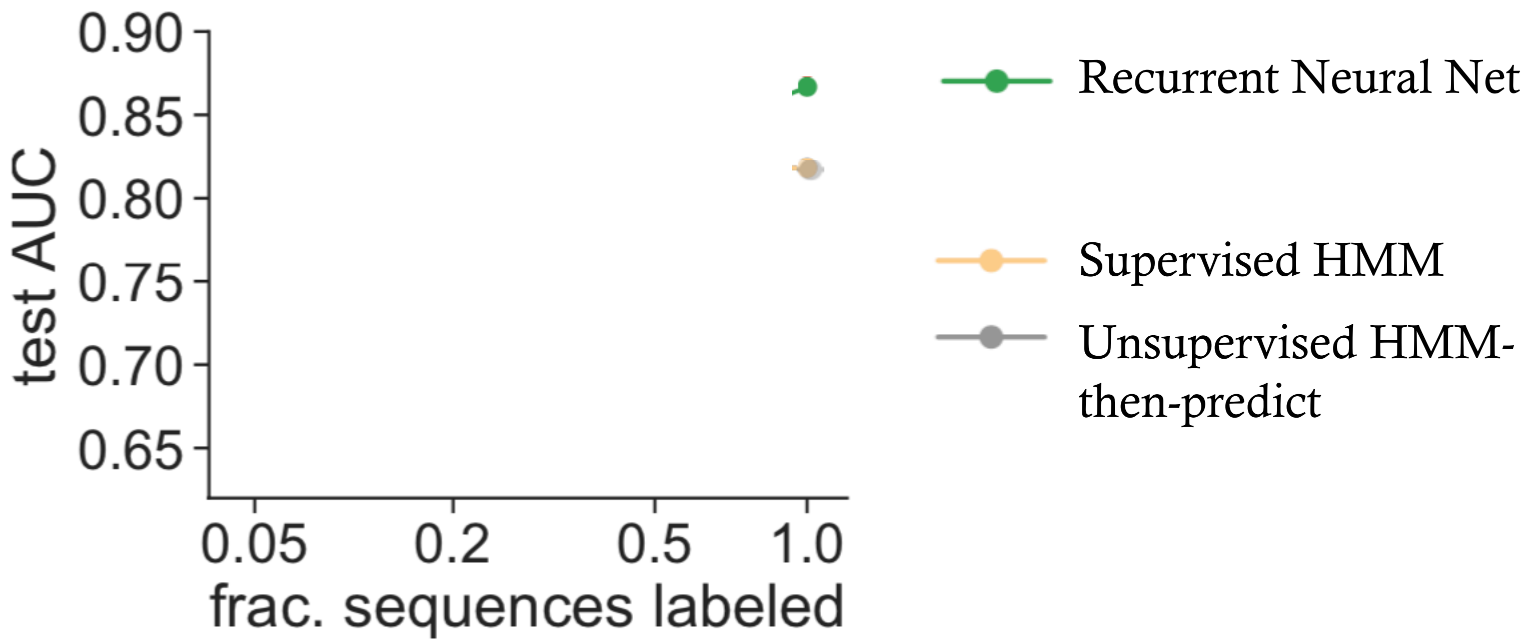


Will patient need ventilator in one hour?

Features x: Time series of 7 vitals and 11 labs

Supervised HMMs predict *poorly*

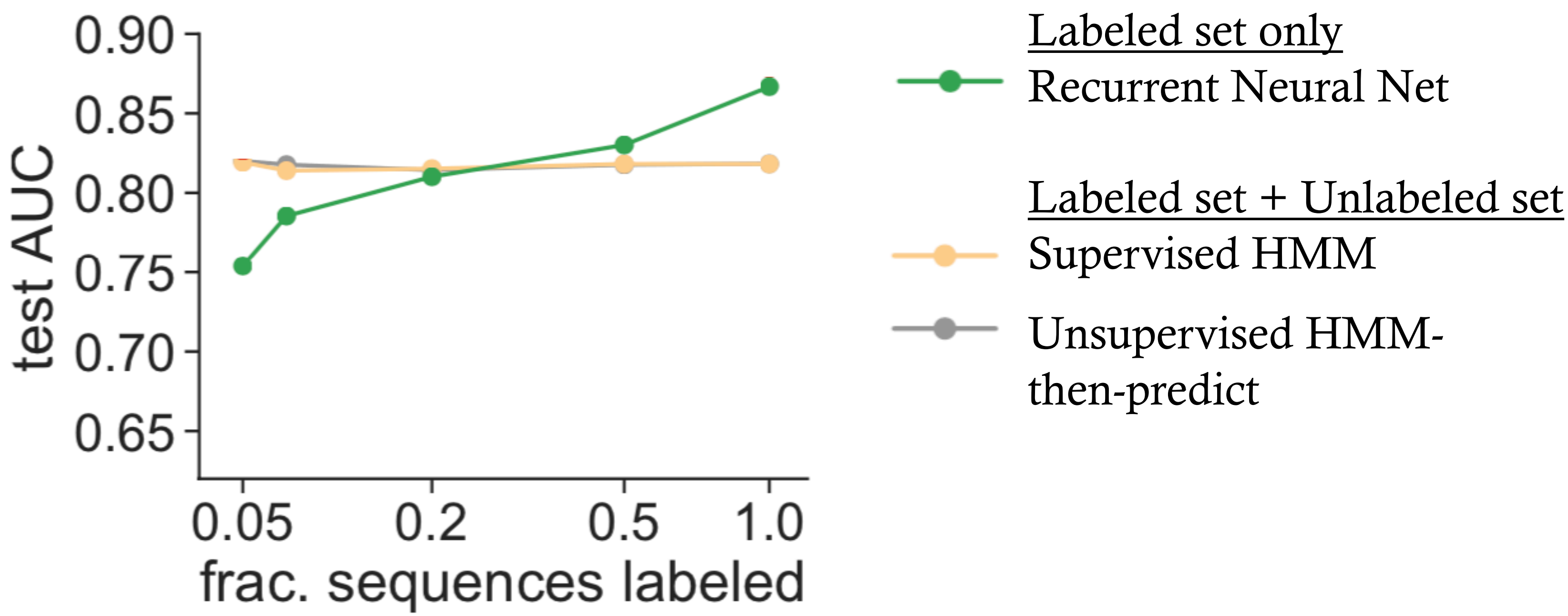
Task: Predicting need for short-term intervention from vital time series
16492 sequences from Boston-area ICU (MIMIC III dataset)



When labels are **abundant,** compared to methods with similar capacity, supervised HMMs tend to be:
- **No better** than unsupervised-then-predict
- **Inferior** to discriminators

Semi-supervised HMMs predict *poorly*

Task: Predicting need for short-term intervention from vital time series
Labeled set: 5%, 10% , 20%, and 50% of 16492 sequences.



When labels are **rare,** compared to methods with similar capacity, supervised HMMs tend to be:

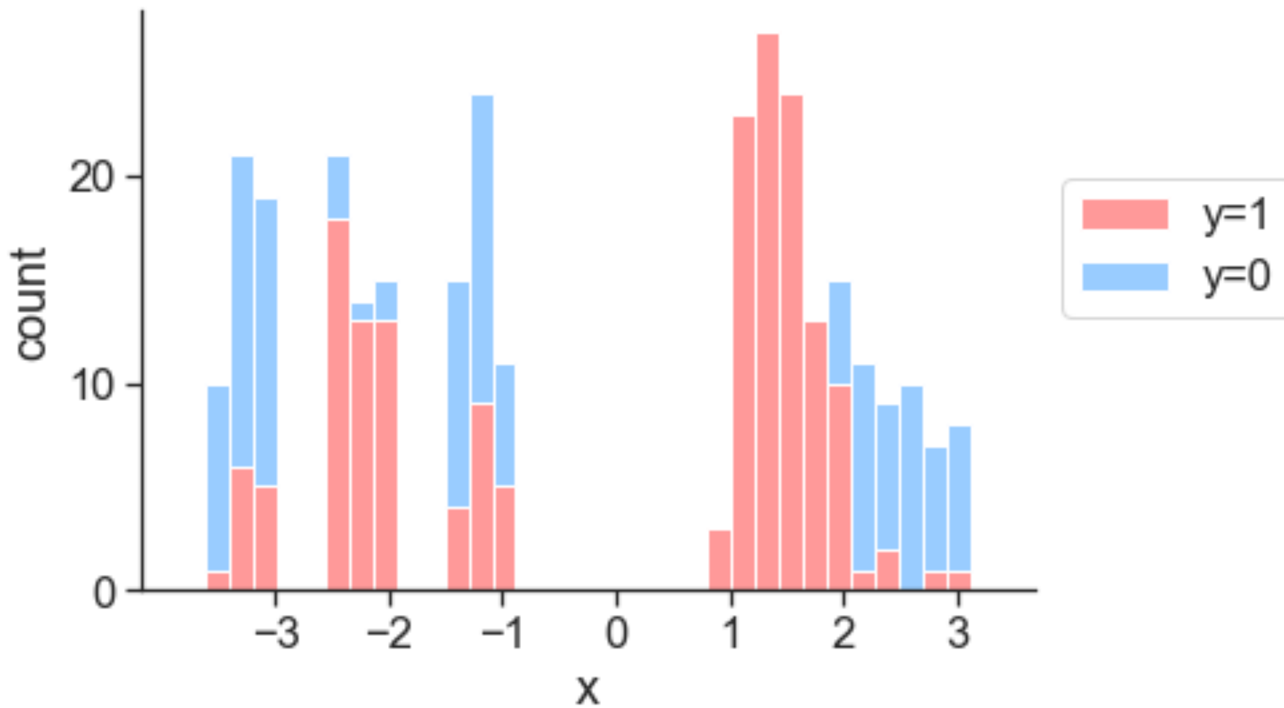
- **No better** than unsupervised-then-predict
- **Superior to** labeled-set-only discriminators

Goals of this Talk

Show that existing supervised LVM training objectives add little predictive value when model is **misspecified.**

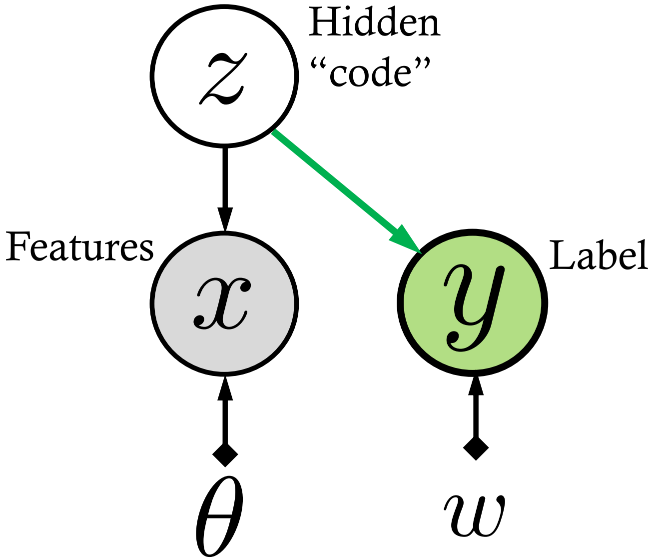
Propose **new training objective** – prediction-constrained (PC) training – that can deliver better label-from-feature predictions despite misspecification.

Toy Data Experiment



Goal: How do supervised LVM training objectives balance two goals in tension: *generative* vs. *discriminative*

Supervised Gaussian Mixture Model



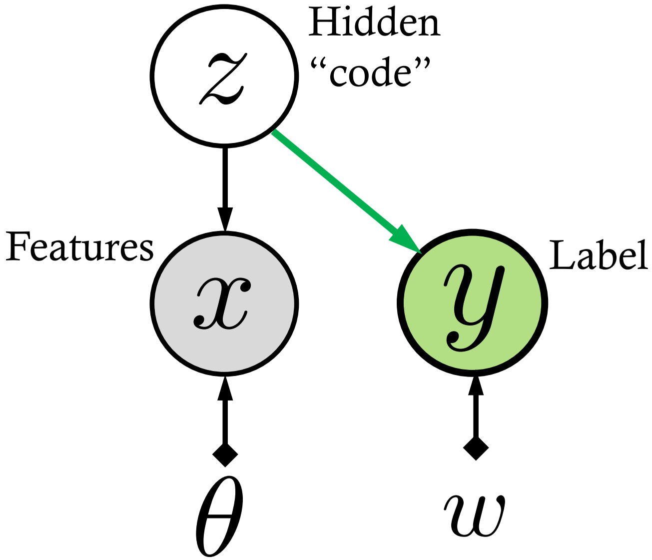
Assume K possible clusters

$$z_n \sim \text{Discrete}(\pi_1, \ldots \pi_K)$$

$$x_n | z_n = k \sim \text{Normal}(\mu_k, \sigma_k^2)$$

$$y_n | z_n = k \sim \text{Bern}(w_k)$$

Supervised Gaussian Mixture Model



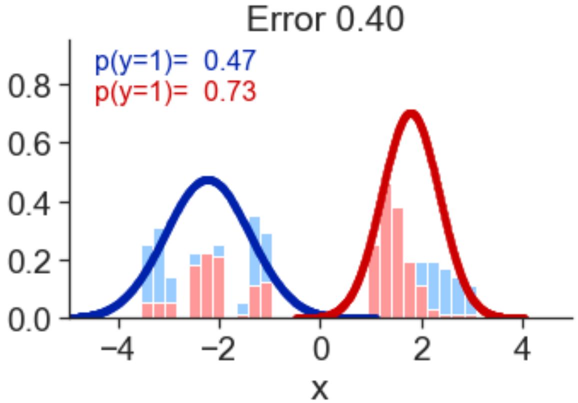
Assume K possible clusters

$$z_n \sim \text{Discrete}(\pi_1, \ldots \pi_K)$$

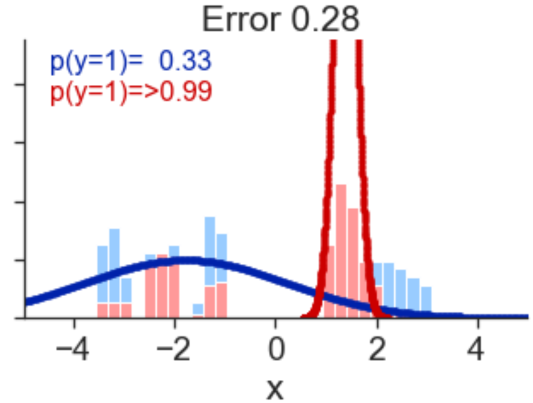
$$x_n | z_n{=}k \sim \text{Normal}(\mu_k, \sigma_k^2)$$

$$y_n | z_n{=}k \sim \text{Bern}(w_k)$$

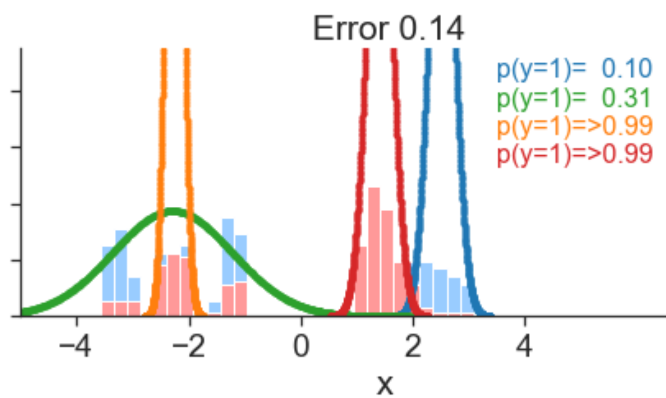
Manual GMM K=2
"good feature likelihood"



Manual GMM K=2
"good label prediction"

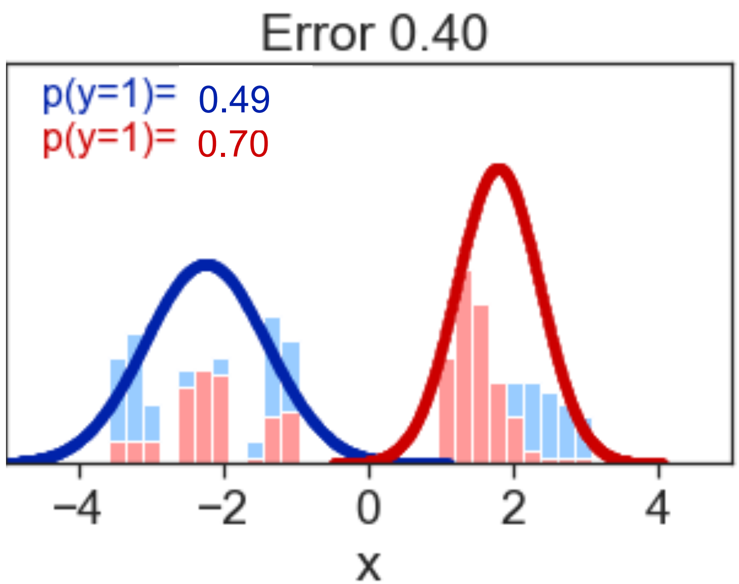


Manual GMM K=4
"good label prediction"

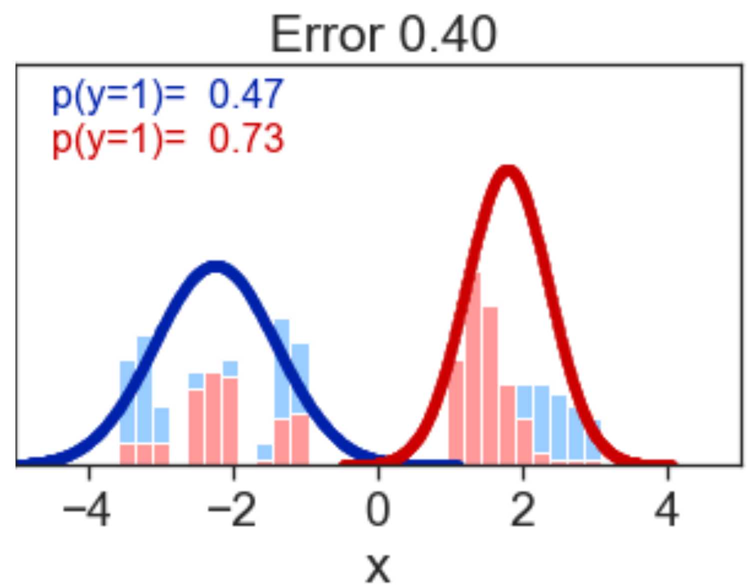


Supervision via joint likelihood *fails*

Unsupervised-then-predict
Best GMM with K=2



Supervised training
Best GMM with K=2

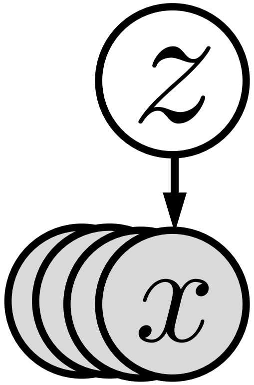


Why doesn't supervision help? *Misspecification*.
Forced to compromise p(y|x) to make p(x) look good.

If my goal prioritizes prediction using p(y|x), maximizing joint likelihood p(x,y) may yield poor results

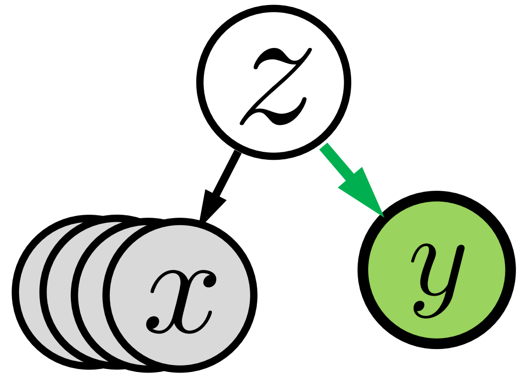
Explaining failure of joint likelihood

Unsupervised LVM



100s of words
or 100s of vitals

Supervised LVM



100s of words
or 100s of vitals

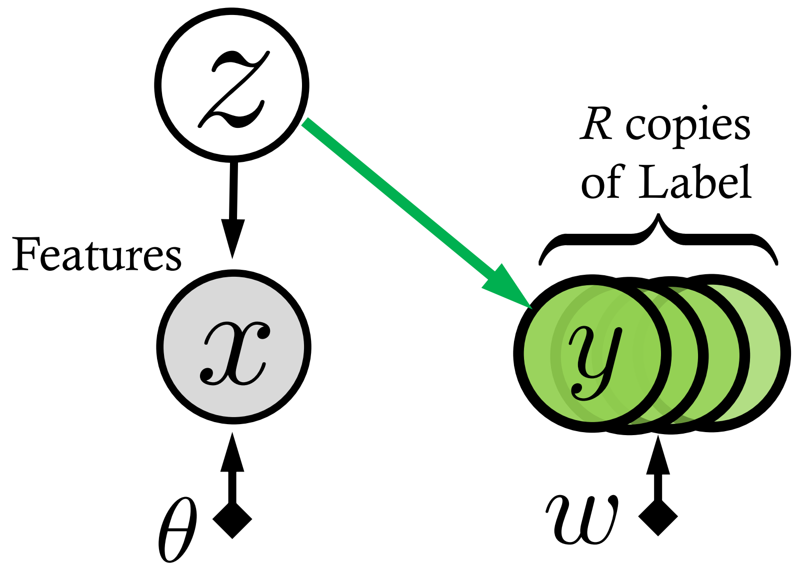
1 binary label

$$\max_{\theta, w} \quad \log p_{\theta, w}(x, y)$$

Supervised training objective treats x and y as interchangeable.
Claim: the **likelihood of x dominates** (due to its larger size).

Not too surprising learned models are indistinguishable.

Attempted fix from past work:
Label **Replication**



$$\max_{\theta,w} \; \log p_{\theta,w}(x, y, y, \ldots y)$$

Proposed separately in several past studies:

- *Vendatam, …, & Murphy (ICLR 2018) :* "Joint VAEs" for images + attributes
- *Zhang & Kjellstrom (2014) :* "Power sLDA" for supervised topic models

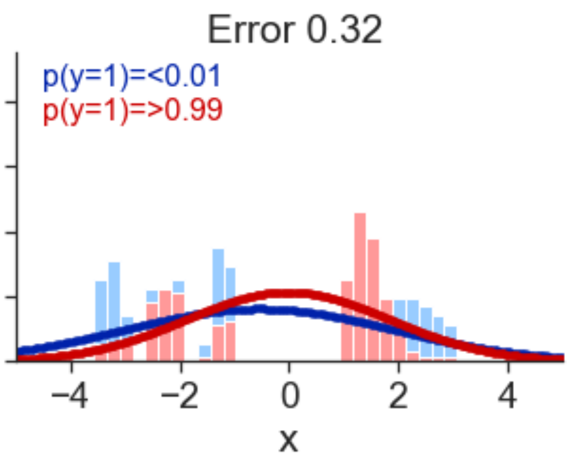
We can show other objectives are equivalent (once framed as point estimation)

- Med-LDA by *Zhu et al. (2012)*

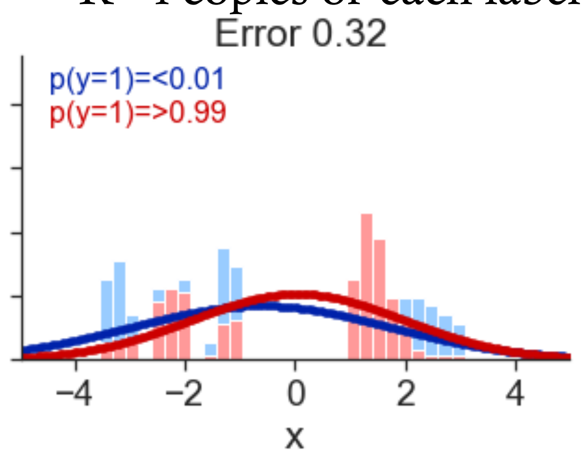
Label Replication *fails*

Supervised GMM with Label Replication

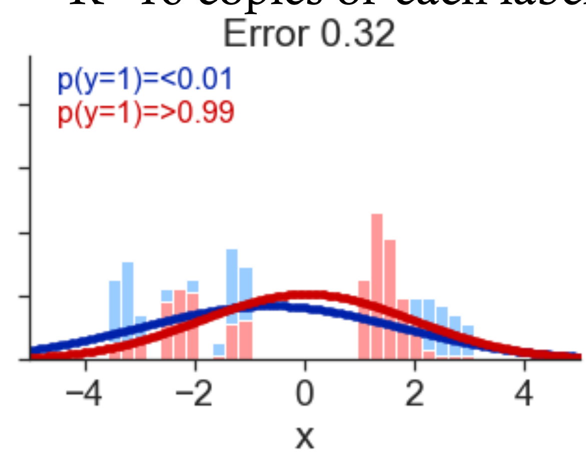
R=2 copies of each label



R=4 copies of each label



R=16 copies of each label



Why?

- During training, driven by the many observed copies of y
- But at test time, unable to perform label from feature prediction

Why does Label Replication fail?

Recall:

> **Goals:**
> - Most important: $p(y|x)$
>   - [?] Predict labels from features well at test time

Does Label Replication objective prioritize y from x?

No. Rewriting via chain rule suggests *no specific emphasis.*

$$p(x, y, y) = p(y, y|x)p(x)$$

y from x is one interpretation

$$= p(x|y, y)p(y, y)$$

x from y is equally valid interpretation

Replication does not emphasize our top priority: y from x

Proposed solution:
Prediction Constrained ("PC") training

Ideal version: Constrained optimization problem

Goal: "Find the ***best model for x*** *that* ***predicts y from x at desired quality****"*

$$\max \quad \textstyle\sum_{x \in \mathcal{D}} \log p_\theta(x)$$
$$\text{subject to:} \quad \textstyle\sum_{x,y \in \mathcal{D}} \log p_{\theta,w}(y|x) \geq \epsilon$$

ϵ is a threshold chosen by stakeholder

Proposed solution:

Prediction Constrained ("PC") training

Practical version: Unconstrained optimization (via Lagrange multiplier theory)

$$\max_{\theta,w} \quad \sum_{x,y\in\mathcal{D}} \underbrace{\log p_\theta(x)}_{\text{good generative model of features}} + \lambda \underbrace{\log p_{\theta,w}(y|x)}_{\text{good predictions of labels from features}}$$

good **generative** model of features

good **predictions** of labels from features

Prediction Constrained ("PC") *training*

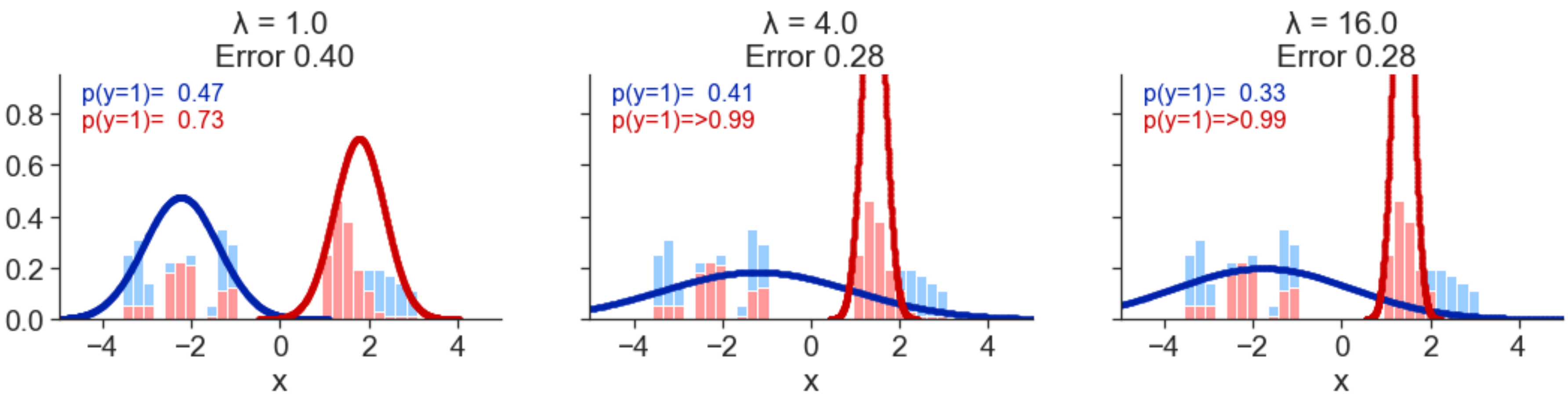
$\lambda > 1$ emphasize models that **predict y from x**

$\lambda = 1$ equivalent to standard supervision (maximizing joint likelihood)

PC can **overcome** misspecification

Equivalent
to max joint likelihood

Stronger
constraint



<u>Related work on learning that overcomes misspecification</u>

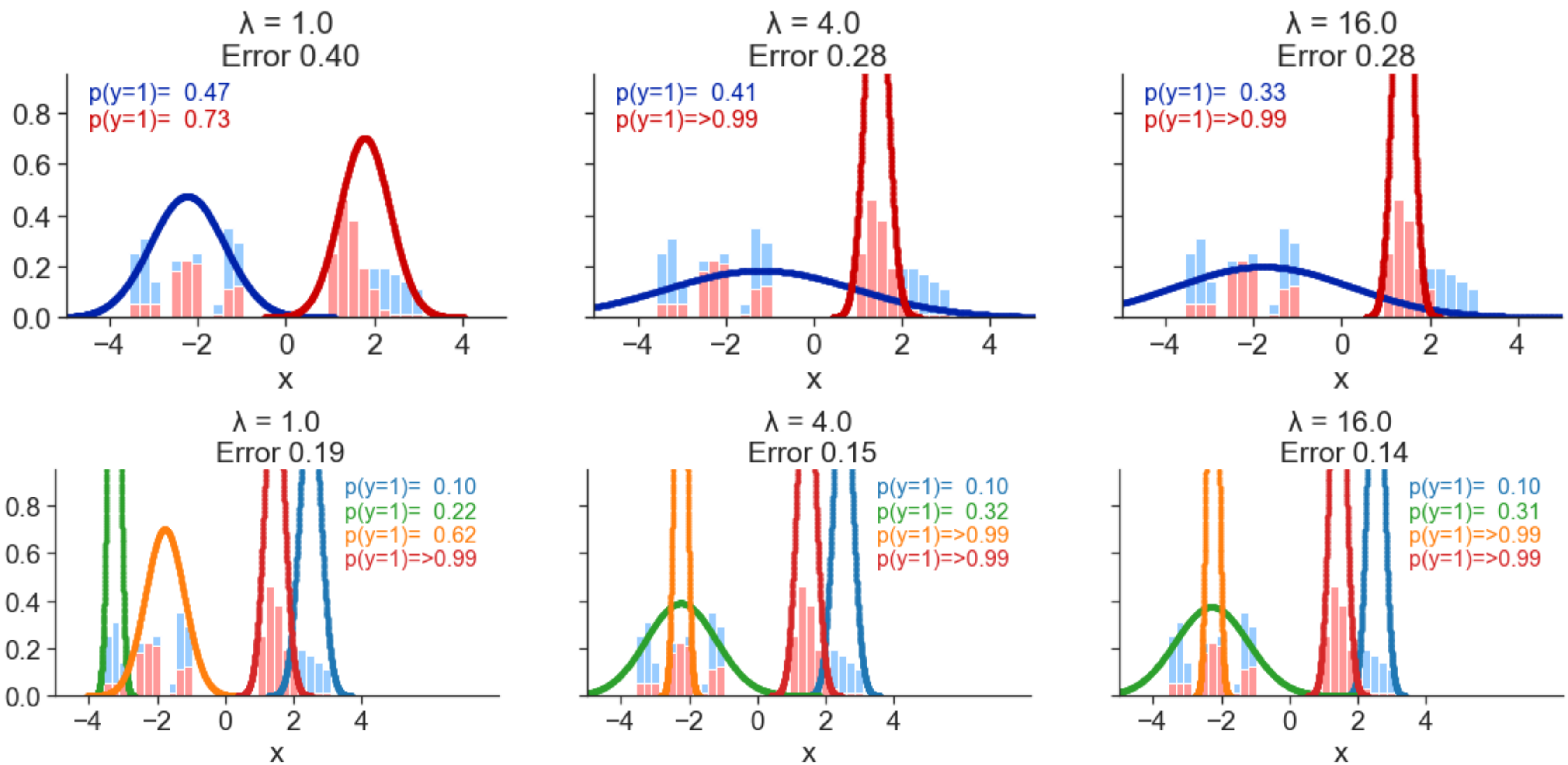
Generalized Bayes : *Bissiri, Holmes, & Walker (2016)* "Safe Bayesian" : *P. Grünwald (2012)*

Power posteriors : *Miller and Dunson (JASA 2019)*

PC can **overcome** misspecification

Equivalent
to max joint likelihood

Stronger
constraint



PC shows benefits even as capacity grows (more clusters)

PC is distinct from Replication

PC upweights **entire y from x marginal likelihood**

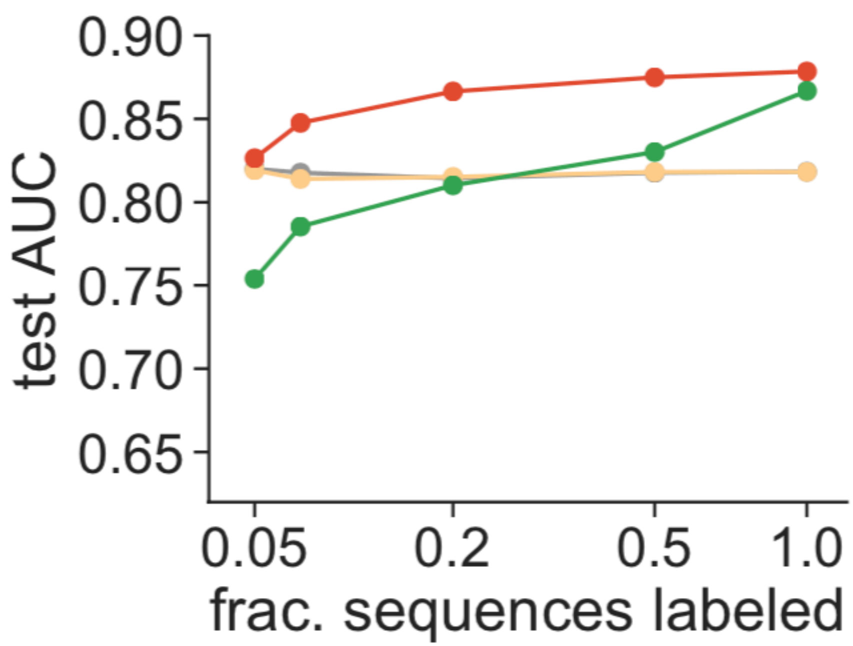
$$p(x)p(y|x)^\lambda$$
$$= p(x)\left(\int_z p_w(y|z)p_\theta(z|x)dz\right)^\lambda$$

Replication upweights **only y from z term**

$$p(x, \underbrace{y \dots y}_{R})$$
$$= \int_z p_w(y|z)^R p_\theta(x|z)p(z)dz$$

PC HMMs deliver better predictions

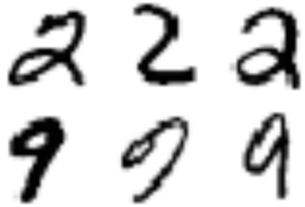
Task: Predicting need for short-term intervention from vital time series
16492 sequences from Boston-area ICU (MIMIC III dataset)



Labeled set only
Recurrent Neural Net
PC HMM
Supervised HMM
Unsupervised HMM-then-predict

- **Better than** unsupervised-then-predict
- **Superior to** labeled-set-only discriminators when labels are rare
- **Competitive with** labeled-set-only discriminators when labels abundant

Semi-Supervised VAEs

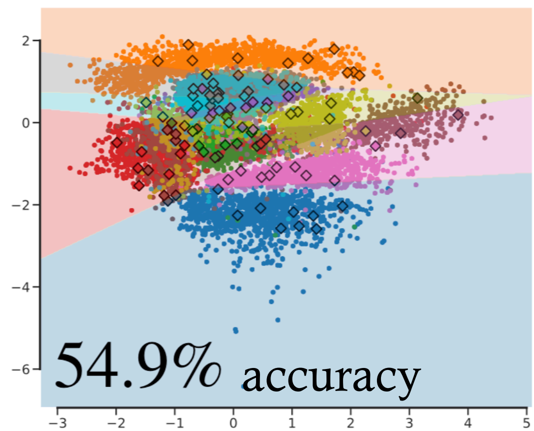


Task: Predict 10-class digit label given MNIST image via VAE

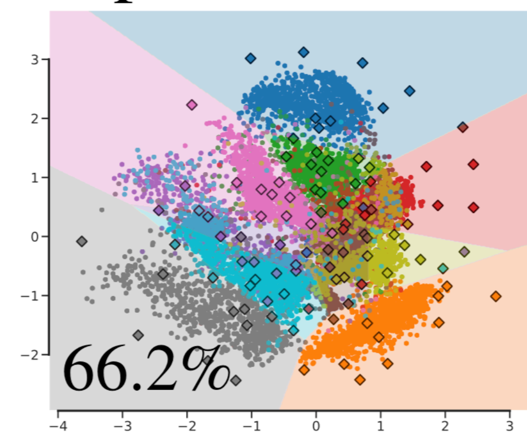
Code size: |z| = 2

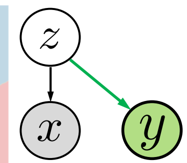
100 labeled
49900 unlabeled

VAE-then-MLP

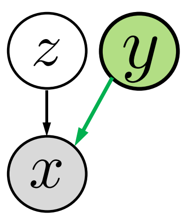


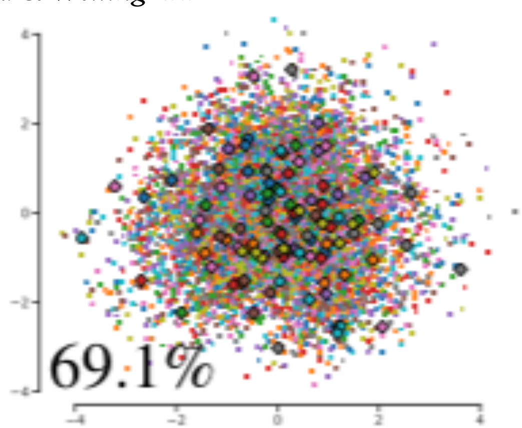
Supervised VAE



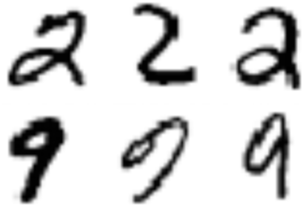


Kingma & Welling '14 M2





PC *improves* Semi-Supervised VAEs



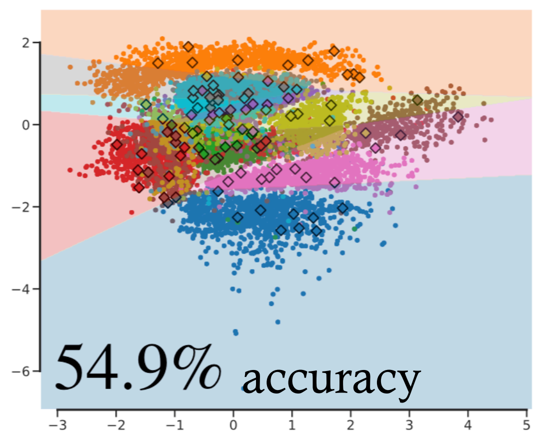
Hope, Abdrakhmanova, Chen, ***Hughes****, Sudderth (in preparation)*

Task: Predict 10-class digit label given MNIST image via VAE

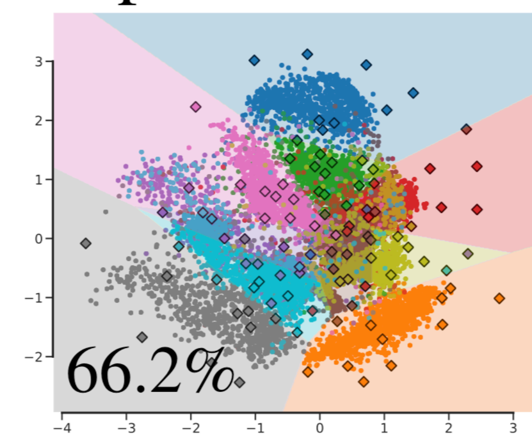
Code size: |z| = 2

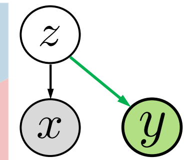
100 labeled
49900 unlabeled

VAE-then-MLP

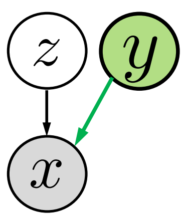


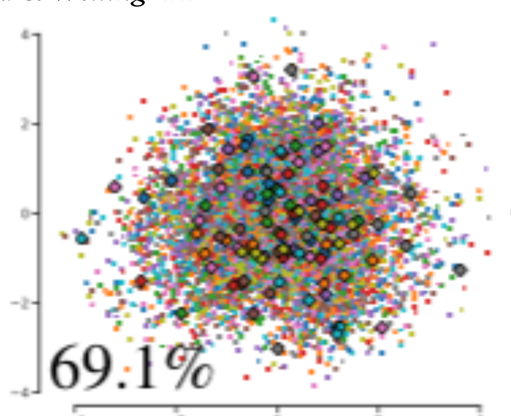
Supervised VAE



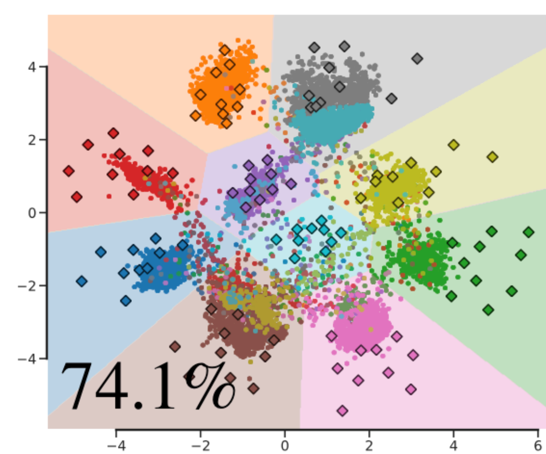


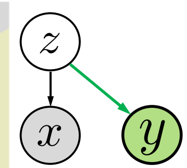
Kingma & Welling '14 M2





PC-VAE

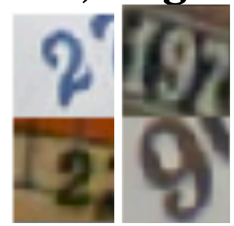




PC *improves* Supervised VAEs

*Hope, Abdrakhmanova, Chen, **Hughes**, Sudderth (in preparation)*

Task: Predict class label given image.
1000 labeled. 20,000+ unlabeled
VAE encoding size 50 (bigger than last slide)



		Method	SVHN (1000)	NORB (1000)
Semi-supervised LVM Methods	*Kingma & Welling '14*	M1 + M2	63.98% (±0.10)	-
	Maaløe et al '16	ADGM	77.14%	89.94% (±0.05)
	Maaløe et al '16	SDGM	83.39% (±0.24)	90.60% (±0.04)
		CPC VAE	**94.22**% (±0.62)	**92.00**% (±1.21)
Semi-supervised Discriminative CNN	*Miyato et al '19*	VAT	**94.23**% (±0.32)	-
Labeled-set only Discriminative CNN		WRN	87.7% (±1.02)	86.7% (±1.32)

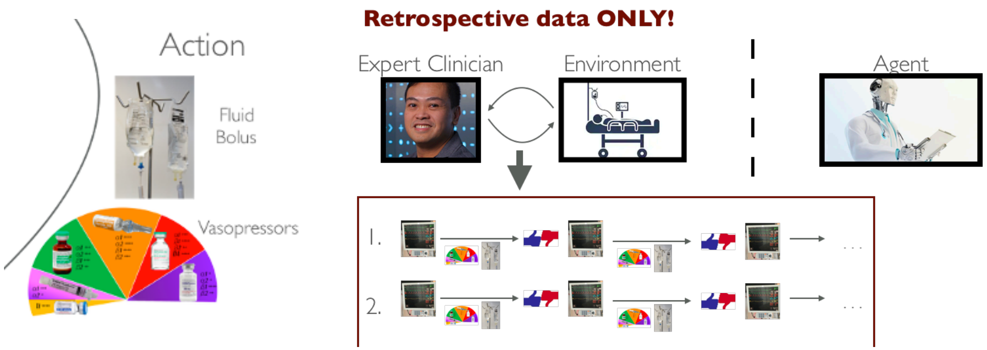
PC-VAEs are

- **Superior to** labeled-set-only discriminators
- **Competitive** with state-of-the-art SSL deep learning (discrim. only)

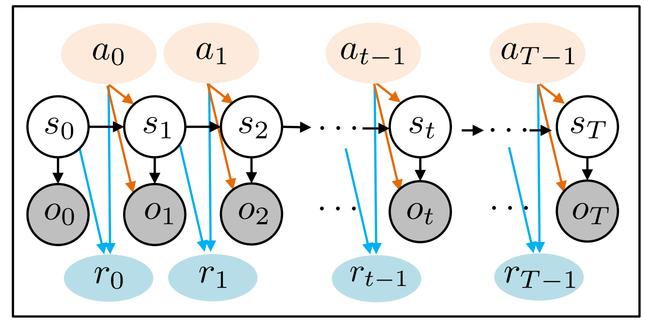
PC training for Model-based RL

*Futoma, **Hughes**, and Doshi-Velez (AISTATS 2020)*

Learning to treat high blood pressure



LVM: POMDP as Input-output HMM

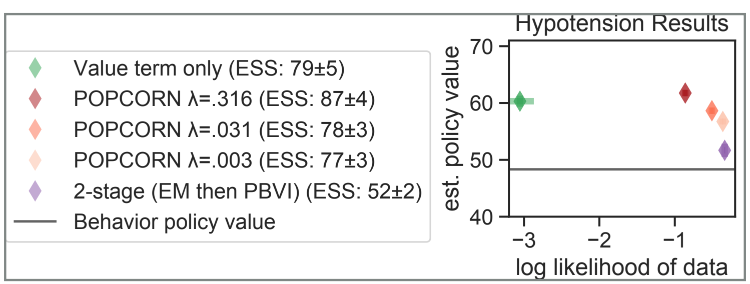


$$\max_{\theta} \;\; \log p_\theta(x) + \lambda V(\pi_\theta)$$

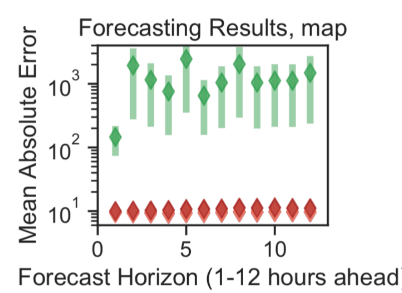
Generative likelihood of the observed features

Value of inferred policy under the generative model

Result: Improved policy value on ICU data



Result: Useful forecasts from model



Lessons Learned

Need to spend more time choosing our objectives

Always debug on simple examples

+ Separate bad algorithm from bad objective
+ Need to **work very hard** to avoid poor local optima

  We show best of 20 runs even for K=2 GMM

Tuning hyperparameters so important

+ Limitation of PC approach: Grid search for lambda

Summary: The Case for Prediction Constrained Training

Existing training objectives add little predictive value when the model is **misspecified.**

New **training objective** – prediction-constrained (PC) training – can deliver better label-from-feature predictions despite misspecification.

<u>PC training delivers all goals</u>

- Most important: $p(y|x)$
  - [ ] Predict labels from features well at test time
- Also important: $p(x, y)$
  - [ ] Predict even when missing features
  - [ ] Train even if only some examples are labeled
  - [ ] Offer interpretable structure

<u>Publications</u>

PC for semi-supervised topic models

Hughes et al. AISTATS 2018

Application to recommending antidepressants

Hughes et al. JAMA Network Open 2020

PC for semi-supervised VAEs

Hope et al. in preparation

PC for POMDPs

Futoma et al. AISTATS 2018