THE UNIVERSITY OF UTAH

Online Bayesian Sparse learning with Spike-and-Slab Priors

Shikai Fang, Shandian Zhe, Kuang-chih Lee,
Kai Zhang, Jennifer Neville

Presenter: Shikai Fang
School of computing, The University of Utah

For ICDM 2020

Outline

1. Motivation
2. Spike and Slab prior
3. Online inference
4. Experiments on real-world data
5. Summary

1. Motivation:

- Many predictive tasks involve a large number of features, e.g., Click-Through (CTR) prediction.

- Too many features could
  I. lead to complicate models, thus request massive training data and computational resources (to avoid over-fitting)
  II. be memory or computationally intensive, not handy for online prediction

- Advantages of Sparse learning
  I. Computational efficiency
  II. Good interpretations and benefit feature engineering

2. Spike and Slab prior

How non-Bayesian people get sparsity?

- L1 regularize

$$\hat{\beta} \equiv \underset{\beta}{\mathbf{argmin}} \left(\|y - X\beta\|^2 + \boxed{\lambda \|\beta\|_1} \right)$$

- L1 & L2 mixture regularize(Elastic Net)

$$\hat{\beta} \equiv \underset{\beta}{\mathbf{argmin}} \left(\|y - X\beta\|^2 + \boxed{\lambda_2 \|\beta\|^2 + \lambda_1 \|\beta\|_1} \right)$$

- Sparse coding, Dictionary learning, Compressed sensing…

All shrinkage uniformly over all feature weights!

2. Spike and Slab prior

How Bayesian people get sparsity?

- Spike and Slab priors :

Introduce binary selection indicators $s_1, s_2, \ldots, s_d$ on each weight

Mild Regularize (**slab**)   Strong shrinkage(**spike**)

2. Spike and Slab prior

How Bayesian people get sparsity?

- Spike and Slab priors :

Introduce binary selection indicators $s_1, s_2, \ldots, s_d$ on each weight

$$p(\mathbf{y}, \mathbf{s}, \mathbf{w} | \rho_0, \tau_0, \mathbf{X}) = p(\mathbf{s}|\rho_0) p(\mathbf{w}|\mathbf{s}, \tau_0) p(\mathbf{y}|\mathbf{w}, \mathbf{X})$$

binary indicators

$$p(\mathbf{s}|\rho_0) = \prod_{j=1}^{d} \text{Bernoulli}(s_j|\rho_0) = \prod_{j=1}^{d} \boxed{\rho_0^{s_j}(1-\rho_0)^{(1-s_j)}}$$

$$\delta(w_j) = \underset{\mathbf{v}\to 0}{\text{limit}}\, \mathcal{N}(w_j|0, v)$$

$$p(\mathbf{w}|\mathbf{s}, \tau_0) = \prod_{j=1}^{d} p(w_j|s_j, \tau_0) = \prod_{j=1}^{d} \boxed{s_j \mathcal{N}(w_j|0, \tau_0) + (1-s_j)\boxed{\delta(w_j)}}$$

Selective shrinkage based on indicator

$$p(\mathbf{y}|\mathbf{w}, \mathbf{X}) = \prod_{j=1}^{n} p(y_j|\mathbf{w}, \mathbf{x}_j)$$

3. Online Inference

The overview of the whole model

- Focus on general binary classification problems, with the form:

$$\boxed{p(y_j|\mathbf{w},\mathbf{x}_j) = \Phi(y_j\mathbf{w}^\top\mathbf{x}_j)} \text{ where } \Phi(t) = \int_{-\infty}^{t} \mathcal{N}(u|0,1)\mathrm{d}u$$

data likelihood!

- Assign spike and slab **prior** over model weights

$$p(\mathbf{s} \mid \rho_0)p(\mathbf{w} \mid \mathbf{s}, \tau_0) = \prod_{j=1}^{d} \rho_0^{s_j}(1-\rho_0)^{(1-s_j)} \prod_{j=1}^{d} s_j\mathcal{N}(w_j \mid 0, \tau_0) + (1-s_j)\delta(w_j)$$

- Goal of Bayesian inference: get the posterior

$$p(\mathbf{y}, \mathbf{s}, \mathbf{w} \mid \rho_0, \tau_0, \mathbf{X}) = p(\mathbf{s} \mid \rho_0)p(\mathbf{w} \mid \mathbf{s}, \tau_0)p(\mathbf{y} \mid \mathbf{w}, \mathbf{X})$$ Joint distribution !

$$p(\mathbf{w}, \mathbf{s} \mid \mathbf{X}, \mathbf{y}, \rho_0, \tau_0) = \frac{p(\mathbf{w}, \mathbf{s}, \mathbf{y} \mid \mathbf{X}, \rho_0, \tau_0)}{\int p(\mathbf{w}, \mathbf{s}, \mathbf{y} \mid \mathbf{X}, \rho_0, \tau_0)\mathbf{dwds}}$$ Posterior !

3. Online Inference

The overview of the whole model

- Goal of Bayesian inference: get the posterior

$$p(\mathbf{w}, \mathbf{s} \mid \mathbf{X}, \mathbf{y}, \rho_0, \tau_0) = \frac{p(\mathbf{w}, \mathbf{s}, \mathbf{y} \mid \mathbf{X}, \rho_0, \tau_0)}{\boxed{\int p(\mathbf{w}, \mathbf{s}, \mathbf{y} \mid \mathbf{X}, \rho_0, \tau_0) \mathbf{dwds}}}$$

Normalizer, constant but intractable

- Exact calculation is not feasible
- **MCMC sampling**: slow converge with high dimensional space
- **Expectation Propagation(EP):**

approximate the intractable posterior with some easy distribution family!

3. Online Inference

- Expectation Propagation:
  Pick the exponential (Exp) family as approximation, which offers good **closure property under multiplication**

$$q(\boldsymbol{\theta}) = \exp\left(\boldsymbol{\lambda}^\top T(\boldsymbol{\theta}) - A(\boldsymbol{\lambda})\right)$$

- Factorize both the exact posterior & approximation posterior

$$p(\boldsymbol{\theta}) \propto \overset{\text{prior}}{\boxed{f_0(\boldsymbol{\theta})}} \boxed{\prod_j f_j(\boldsymbol{\theta})}\ \text{likelihood} \qquad q(\boldsymbol{\theta}) \propto \boxed{\tilde{f}_0(\boldsymbol{\theta})} \boxed{\prod_j \tilde{f}_j(\boldsymbol{\theta})}$$

Global approximation by factor-wise approximation!

$$\tilde{f}_0(\boldsymbol{\theta}) \approx f_0(\boldsymbol{\theta}), \tilde{f}_j(\boldsymbol{\theta}) \approx f_j(\boldsymbol{\theta})$$

$$\tilde{f}_0(\boldsymbol{\theta}), \tilde{f}_1(\boldsymbol{\theta}) \ldots, \tilde{f}_n(\boldsymbol{\theta}) \in \mathrm{Exp}(\boldsymbol{\theta})$$

3. Online Inference

- Expectation Propagation, standard version
  - Go through every factor $f_0(\boldsymbol{\theta}), \{f_j\}_{j=1}^{n}$

$$q^{\backslash j}(\boldsymbol{\theta}) \propto \frac{q(\boldsymbol{\theta})}{\tilde{f}_j(\boldsymbol{\theta})}$$

Calibration distribution (context)

Tilted distribution

$$q^*(\boldsymbol{\theta}) = \underbrace{\operatorname{argmin} \quad \mathrm{KL}\left(\boxed{q^{\backslash j}(\boldsymbol{\theta}) f_j(\boldsymbol{\theta})} \| q(\boldsymbol{\theta})\right)}_{\text{Moment match!}}$$

$$\tilde{f}_j^{\text{new}}(\boldsymbol{\theta}) \propto \frac{q^*(\boldsymbol{\theta})}{q^{\backslash j}(\boldsymbol{\theta})}$$

•Must cyclically go-through all data points and approximate corresponding factors … <u>*inefficient storage when # of data is large!*</u>

3. Online Inference

- Stochastic Expectation Propagation
  - estimate an **average approx. likelihood factor**
  - Update the in average approx. likelihood factor **an online fashion!**

$$q(\boldsymbol{\theta}) \propto \tilde{f}_0(\boldsymbol{\theta}) \boxed{\prod_j \tilde{f}_j(\boldsymbol{\theta})}$$

Standard Expectation Propagation

$$p(\boldsymbol{\theta}) \propto f_0(\boldsymbol{\theta}) \prod_{j=1}^{n} f_j(\boldsymbol{\theta})$$

Exact posterior

$$q(\boldsymbol{\theta}) \propto \tilde{f}_0(\boldsymbol{\theta}) \boxed{\left(\tilde{f}_a(\boldsymbol{\theta})\right)^n}$$

Stochastic Expectation Propagation

3. Online Inference

- Stochastic Expectation Propagation
  - Initialize $\tilde{f}_0(\boldsymbol{\theta})$ $\tilde{f}_a(\boldsymbol{\theta})$
  - Go through each data samples $(j = 1, \ldots, n)$

$$q^{\backslash j}(\boldsymbol{\theta}) \propto \frac{q(\boldsymbol{\theta})}{\tilde{f}_a(\boldsymbol{\theta})}$$

$$q^*(\boldsymbol{\theta}) = \text{argmin} \quad \mathrm{KL}\big(q^{\backslash j}(\boldsymbol{\theta}) f_j(\boldsymbol{\theta}) \| q(\boldsymbol{\theta})\big)$$

$$\tilde{f}_j^{\text{new}}(\boldsymbol{\theta}) \propto \frac{q^*(\boldsymbol{\theta})}{q^{\backslash j}(\boldsymbol{\theta})}$$

$$\tilde{f}_a^{\text{new}}(\boldsymbol{\theta}) = \frac{1}{n}\big((n-1)\tilde{f}_a(\boldsymbol{\theta}) + \tilde{f}_j^{\text{new}}(\boldsymbol{\theta})\big)$$

stochastic update

$$= (1-\epsilon)\tilde{f}_a(\boldsymbol{\theta}) + \boxed{\epsilon} \cdot \tilde{f}_a^{\text{new}}(\boldsymbol{\theta})$$

learning rate

3. Online Inference

What we have as far?

- Spike and Slab Prior
- Stochastic Expectation Propagation framework

What the next step?

- Design the exact form of approximation factor: fully factorized form

$$p(\mathbf{y}, \mathbf{s}, \mathbf{w} | \rho_0, \tau_0, \mathbf{X})$$

$$= \prod_{i=1}^{d} \text{Bernoulli}(s_i|\rho_0) \prod_{i=1}^{d} \boxed{\big(s_i \mathcal{N}(w_i|0, \tau_0) + (1 - s_i)\delta(w_i)\big)} \cdot \prod_{j=1}^{n} \boxed{\Phi(y_j \mathbf{w}_{t_j}^{\top} \hat{\mathbf{x}}_j)}$$

Some Exp distribution family ?

Some Exp distribution family ?

3. Online Inference

Design the exact form of approximation factor

$$p(\mathbf{y}, \mathbf{s}, \mathbf{w} | \rho_0, \tau_0, \mathbf{X})$$

$$= \prod_{i=1}^{d} \text{Bernoulli}(s_i|\rho_0) \prod_{i=1}^{d} \boxed{\left(s_i \mathcal{N}(w_i|0, \tau_0) + (1 - s_i)\delta(w_i)\right)} \cdot \prod_{j=1}^{n} \boxed{\Phi(y_j \mathbf{w}_{t_j}^\top \hat{\mathbf{x}}_j)}$$

$$q(\mathbf{y}, \mathbf{s}, \mathbf{w}) \propto \prod_{i=1}^{d} \text{Bernoulli}(s_i|\rho_0) \prod_{i=1}^{d} \boxed{\text{Bernoulli}(s_i|\rho_i)\mathcal{N}(w_i|\mu_{1i}, v_{1i})}$$

Bernoulli+Gausssian as the approx. of spike and slab

$$\prod_{j=1}^{n} \prod_{k \in t_j} \boxed{\mathcal{N}(w_k|\mu_{2k}^{+}, v_{2k}^{+})^{\mathcal{I}(y_j=1)} \mathcal{N}(w_k|\mu_{2k}^{-}, v_{2k}^{-})^{\mathcal{I}(y_j=-1)}}$$

maintain two types of average likelihood (two Gaussian!) approximations!

for positive samples & for negative samples correspondingly

3. Online Inference

By arranging the terms, we get the final form of approx. posterior

$$q(\mathbf{y}, \mathbf{s}, \mathbf{w}) \propto \prod_{i=1}^{d} \overbrace{\boxed{\text{Bernoulli}(s_i|\rho_0)\text{Bernoulli}(s_i|\rho_i)\mathcal{N}(w_i|\mu_{1i}, v_{1i})}}^{\text{Approx. factors on each weight prior}} \overbrace{\boxed{\mathcal{N}(w_i|\mu_{2i}^{+}, v_{2i}^{+})^{n_i^{+}}}}^{\text{Approx. factor on each pos. sample}} \underbrace{\boxed{\mathcal{N}(w_i|\mu_{2i}^{-}, v_{2i}^{-})^{n_i^{-}}}}_{\text{Approx. factor on each neg. sample}}$$

- The weight for positive & negative samples are decided by n_i^{+} n_i^{-}
- We can keep the count from the data or we can set it manually (equivalent to duplicate data)
- The updating of approx.prior factors are affected by settings of n_i^{+} n_i^{-}

THE UNIVERSITY OF UTAH

3. Online Inference

Final algorithm framework

• Initialize (uninformative initialization)

$$\rho_i = 0.5, \mu_{1i} = \mu_{2i}^{+} = \mu_{2i}^{-} = 0, v_{1i} = v_{2i}^{+} = v_{2i}^{-} = 10^6 (1 \le i \le d)$$

• Go through each data sample j, and do:

- Calculate calibrate distribution

$$q^{\backslash j}(\mathbf{w}_{t_j}) = \frac{q(\mathbf{w}_{t_j})}{\prod_{k \in t_j} \mathcal{N}(w_k | \mu_{2k}^{+}, v_{2j}^{+})^{\mathcal{I}(y_j=1)} \mathcal{N}(w_k | \mu_{2k}^{-}, v_{2k}^{-})^{\mathcal{I}(y_j=-1)}}$$

- Update the approx. average likelihood accordingly (see details in paper)

$$v_{2k}^{+\ -1} \leftarrow \frac{n_k^{+} - 1}{n_k^{+}} v_{2k}^{+\ -1} + \frac{1}{n_k^{+}} v_{2k}^{+,\mathrm{new}\ -1} \quad , \frac{\mu_{2k}^{+}}{v_{2k}^{+}} \leftarrow \frac{n_k^{+} - 1}{n_k^{+}} \frac{\mu_{2k}^{+}}{v_{2k}^{+}} + \frac{1}{n_k^{+}} \frac{\mu_{2k}^{+}}{v_{2k}^{+,\mathrm{new}}}$$

$$v_{2k}^{-\ -1} \leftarrow \frac{n_k^{-} - 1}{n_k^{-}} v_{2k}^{-\ -1} + \frac{1}{n_k^{-}} v_{2k}^{-,\mathrm{new}\ -1} \quad , \frac{\mu_{2k}^{-}}{v_{2k}^{-}} \leftarrow \frac{n_k^{-} - 1}{n_k^{-}} \frac{\mu_{2k}^{-}}{v_{2k}^{-}} + \frac{1}{n_k^{-}} \frac{\mu_{2k}^{-}}{v_{2k}^{-,\mathrm{new}}}$$

- If N_batch samples has been processed, update weight prior factors(see details in paper) $\mathrm{Bernoulli}(s_i|\rho_i)\mathcal{N}(w_i|\mu_{1i}, v_{1i})$. $(1 \le i \le d)$

4. Experiments

When we finish training...

- How to do feature selection?
  - Check the posterior feature selection indicator
  - choose feature with weight $\left\{ j \mid q(s_j = 1) > \frac{1}{2} \right\}$

- How to do predict?
  - **Simply prediction**: just use the posterior mean of selected feature weight(what we pick for experiments)

  - Bayesian prediction: use both the posterior mean and variance, final predictions are given as a distribution (need sampling trick)

4. Experiments

Dataset and Baseline

- Online Ad click dataset **Gemini and BrightRoll** from Yahoo! Platform
  - Split in three groups(see details in paper)

	Train size	Test size	Num of features
GROUP1	9.7M	553.6M + 878.7M+ 546.8M	1,074,917
GROUP2	1.8M	116.0M + 110.2M+ 133.7M	204,327
GROUP3(unbalanced)	798,152(5004 clicks)	547,043(3688 clicks)	617,258

- Baselines
  - FTRL-proximal: sparse online logistic regression with L1&L2 regularization(fine-tunning the optimal regularization weights)
  - Vowpal Wabbit(VW): online logistic regression with all features

4. Experiments

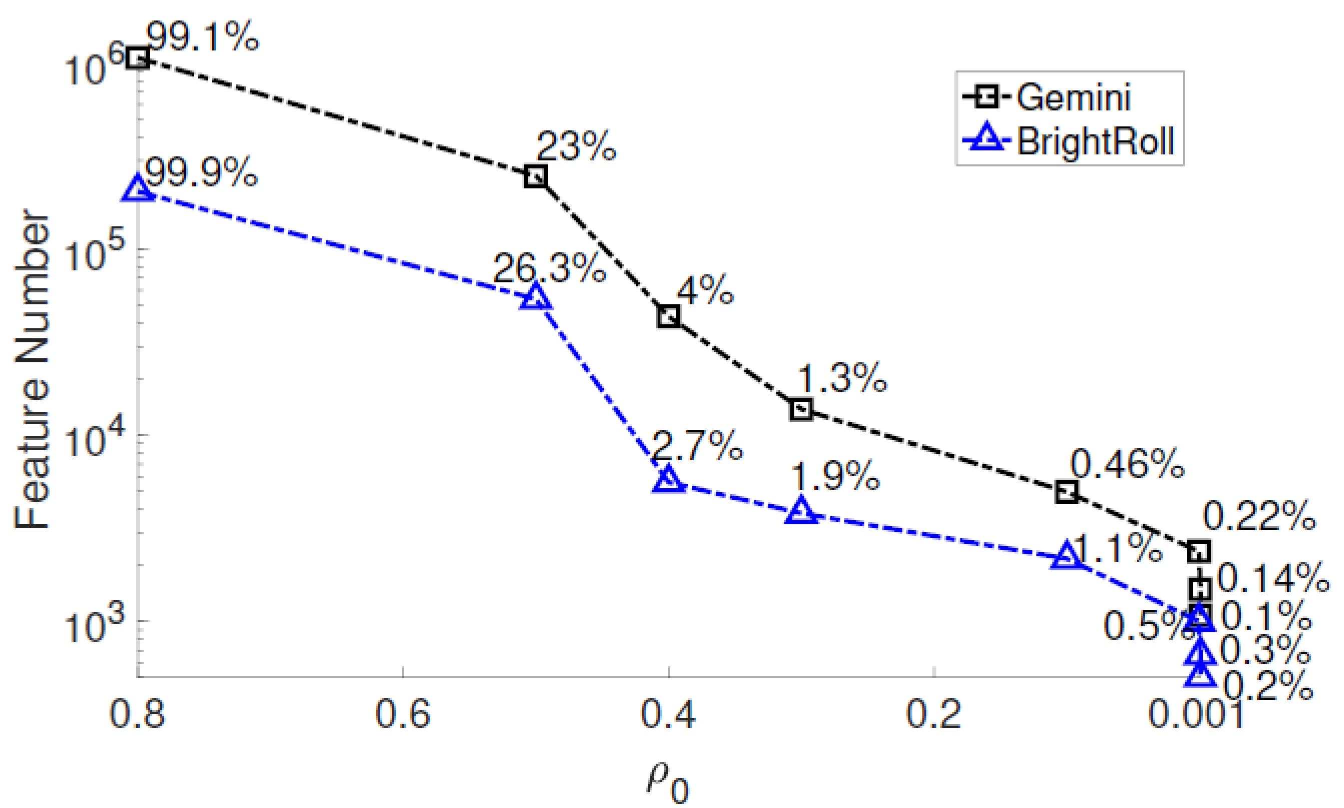
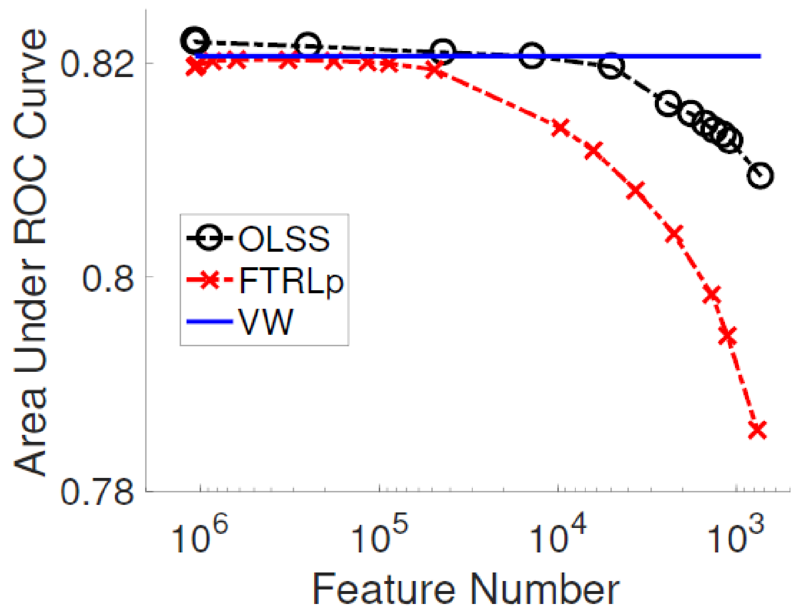


Fig. 1. The sparsity levels achieved by OLSS: number of selected features *v.s.* setting of ρ_0. Numbers on data points show the feature selection ratio.

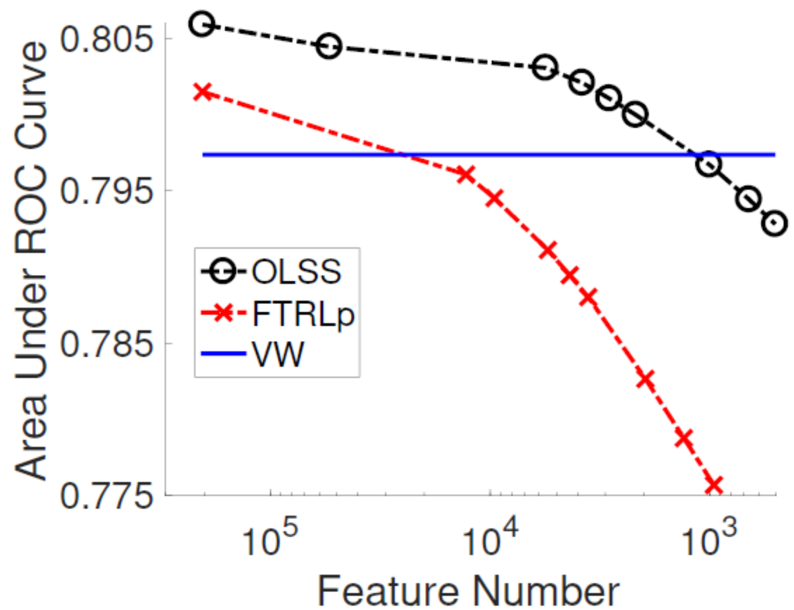
4. Experiments

Gemini

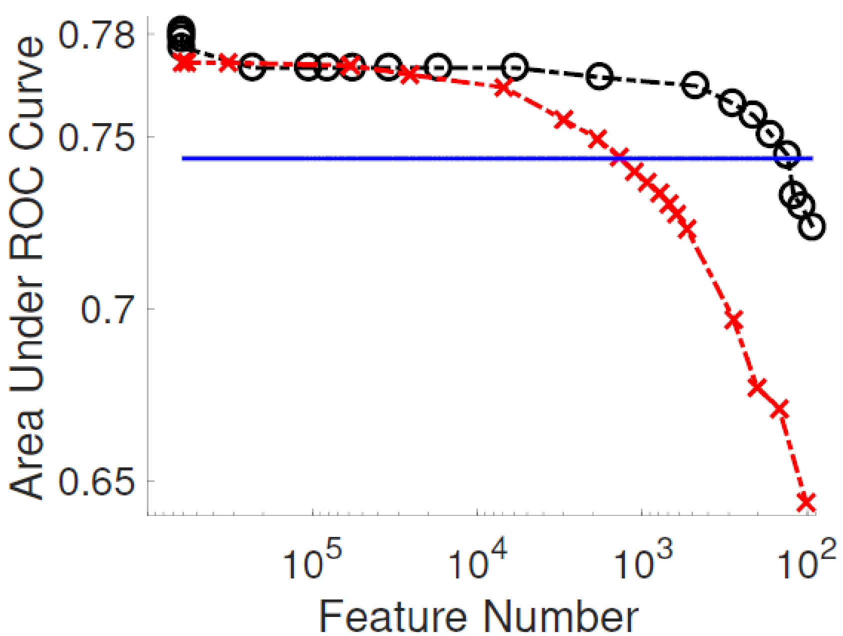


(a) 05/02/2015

BrightRoll



(a) 07/22/2016

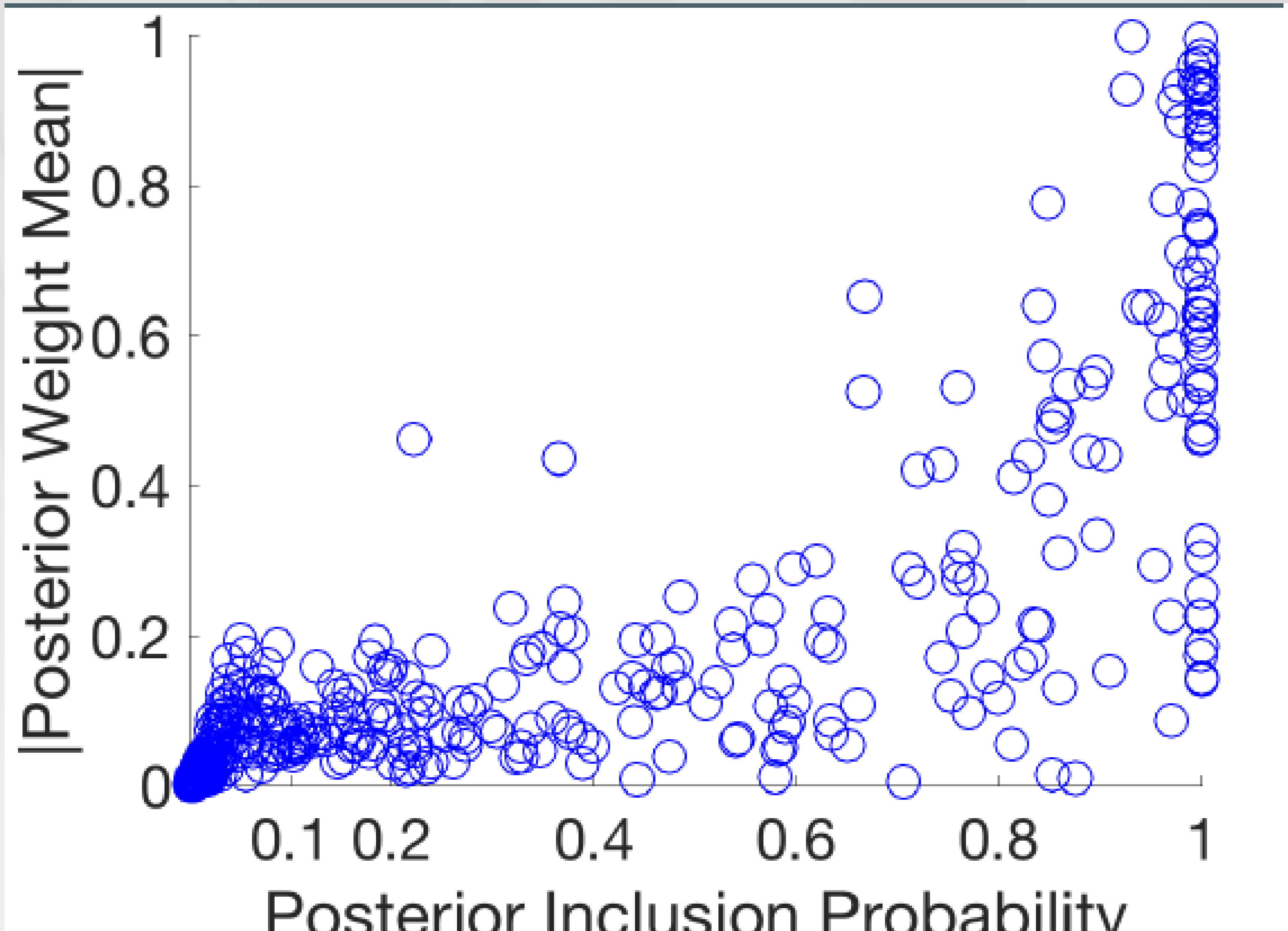


(e) Imbalanced training

Imbalanced Group

THE UNIVERSITY OF UTAH

4. Experiments



(f) Selection Uncertainty

5. Summary

1. We apply the Bayesian *spike and slab prior* on general binary classification problems where the sparsity helps the feature selection over large number of feature.

2. Based on the *Stochastic Expectation Propagation* framework, we developed an online inference algorithm, which could handle large-scale size data efficiently by giving closed form update.

3. We test the proposed method on real-word dataset, and get convincing results on both fair predictions and the sparsity of model.

THE UNIVERSITY OF UTAH

Thanks for attention
Q&A Time

My email: shikai.fang@utah.edu

Focus: Probabilistic model, Bayesian deep learning and its application

Group Homepage: https://www.cs.utah.edu/~zhe/