Bayesian Modelling of Concept Drift

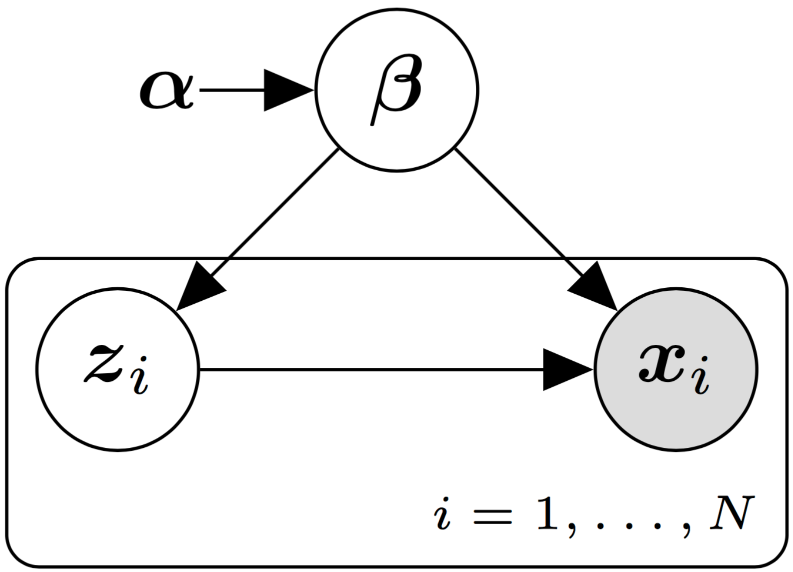
Andrés R. Masegosa

University of Almería
Spain

1st August 2018, Berlin

The problem

SETTINGS



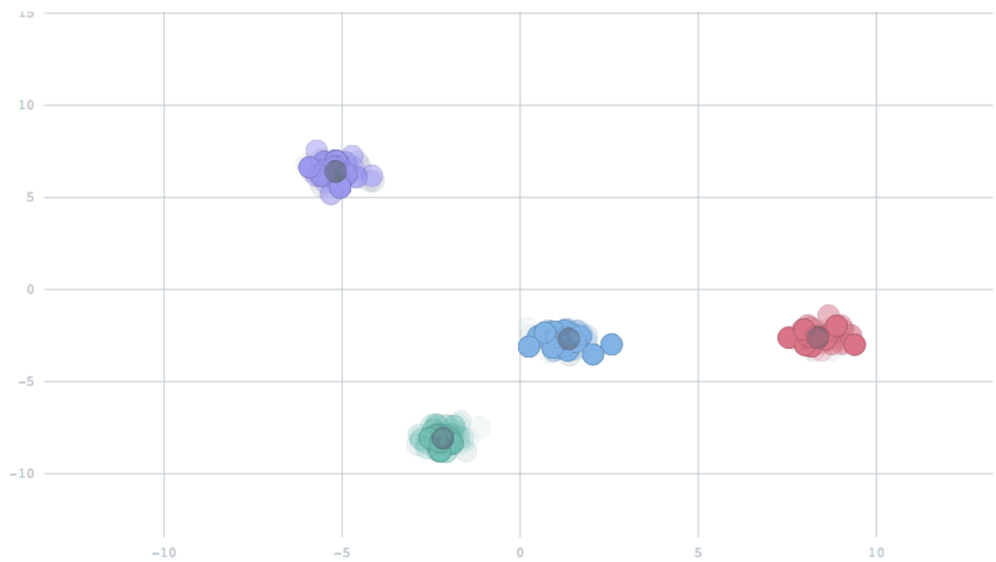
Latent Variable Models (LVMs):

- PCA, MoG, LDAs, HMM, Kalman Filter, Factorial Models, Hierarchical Linear Regression, Matrix Factorization, etc.

Winn & Bishop, 2005 Hoffman et al., 2013

THE PROBLEM

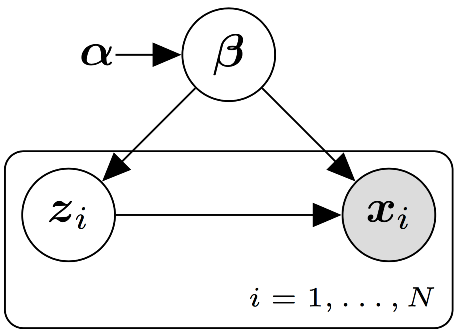
Freeman J. Introducing streaming k-means in Apache Spark 1.2.
https://databricks.com/blog/2015/01/28/introducing-streaming-k-means-in-spark-1-2.html



- **Learning LVMs from (non-stationary) Data Streams**
  - Continuous Model Updating.
  - Presence of Concept Drift.

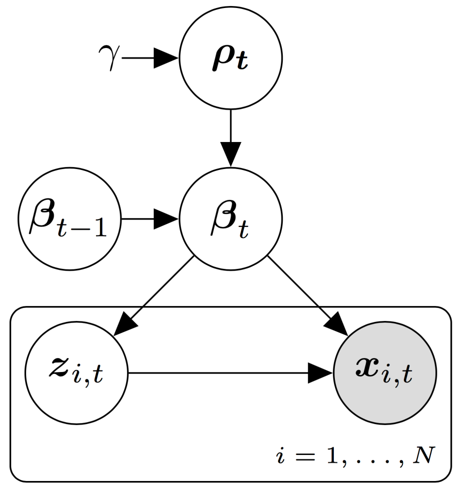
Gama et al., 2014

Our Proposal



Data Set

$\Rightarrow$



Stream of Data Batches

- **Out-of-the-box temporal extension.**
  - Global parameters β_t evolve over time.
  - Hierarchical prior modeling concept drift.
  - Closed-form Variational inference.

Related Work

Exponential Forgetting

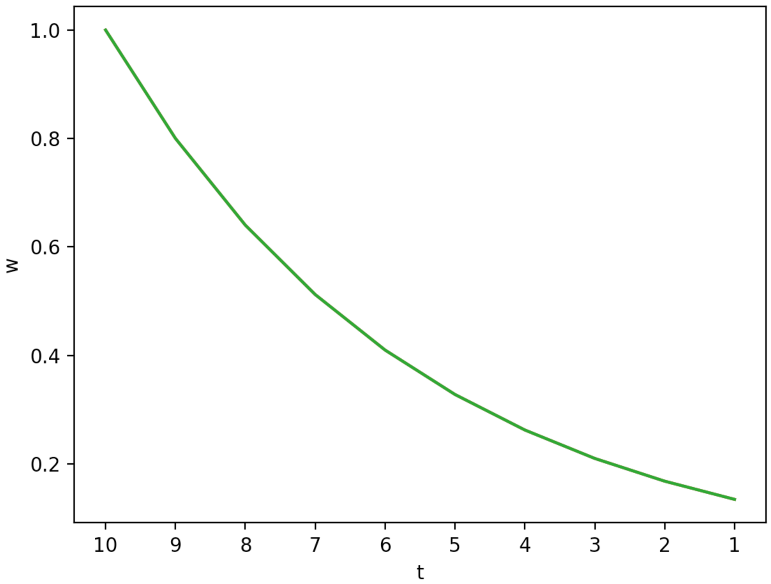
Exponential Forgetting

$$\arg\max_{\beta} \sum_{i=1}^{T} \ln p(\boldsymbol{x}_i|\boldsymbol{\beta})$$

$$\Downarrow$$

$$\arg\max_{\beta} \sum_{i=1}^{T} w_i \ln p(\boldsymbol{x}_i|\boldsymbol{\beta})$$

$$w_i = \rho^{T-i} : \rho \in (0, 1]$$



Exponential Forgetting:

- ρ is the exponential forgetting rate.
- Assign a decreasing weight to each data sample.
- Older data samples has less influence in the parameters.

Exponential Forgetting

Bayesian Learning with Exponential Forgetting:

$$p(\boldsymbol{\beta}|\boldsymbol{x}_1,\ldots,\boldsymbol{x}_T,\rho) = \frac{p(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_T|\boldsymbol{\beta},\rho)p(\boldsymbol{\beta})}{p(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_T,\rho)} = \frac{p(\boldsymbol{\beta})\prod_{i=1}^{T} p(\boldsymbol{x}_i|\boldsymbol{\beta})^{\rho^{T-i}}}{p(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_T,\rho)}$$

Variational Inference with Exponential Forgetting:

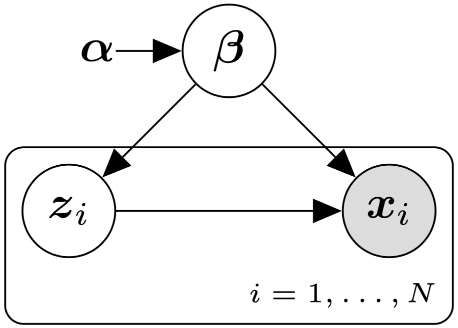
$$\mathcal{L}_\rho(\boldsymbol{\lambda}) = \mathbb{E}_q[\sum_{i=1}^{T} \rho^{T-i} \ln p(\boldsymbol{x}_i|\boldsymbol{\beta})] - KL(q(\boldsymbol{\beta}|\boldsymbol{\lambda})||p(\boldsymbol{\beta}))$$

Honkela and Valpola, 2003

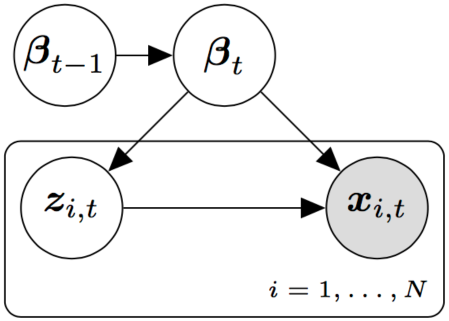
Implicit Transition Models

Kárnỳ (2014) and Özkan et al. (2013)

Explicit Transition Models



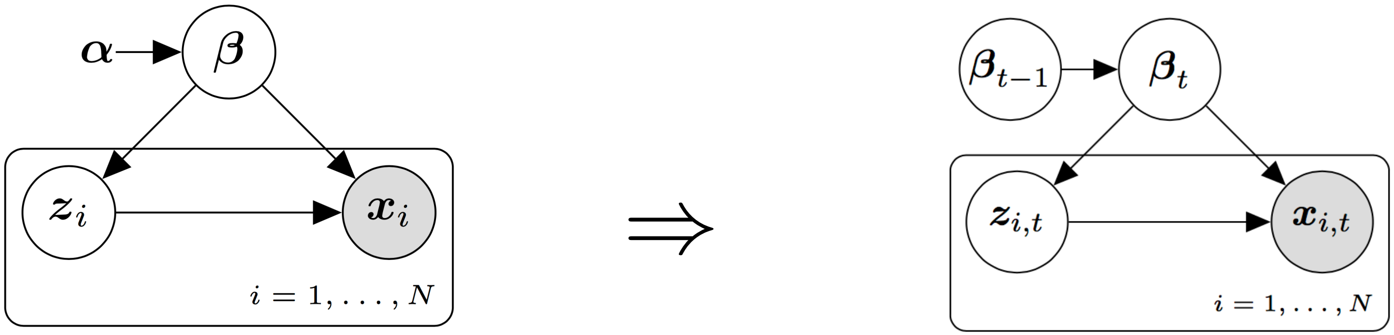
$\Rightarrow$



$$p(\boldsymbol{\beta}_t | \boldsymbol{x}_{1:t-1}) = \int p(\boldsymbol{\beta}_t | \boldsymbol{\beta}_{t-1}) p(\boldsymbol{\beta}_{t-1} | \boldsymbol{x}_{1:t-1}) d\boldsymbol{\beta}_{t-1}$$

- **Explicit Transition Models**
  - Stationary transition model with requires domain knowledge.
  - Outside conjugate exponential family.

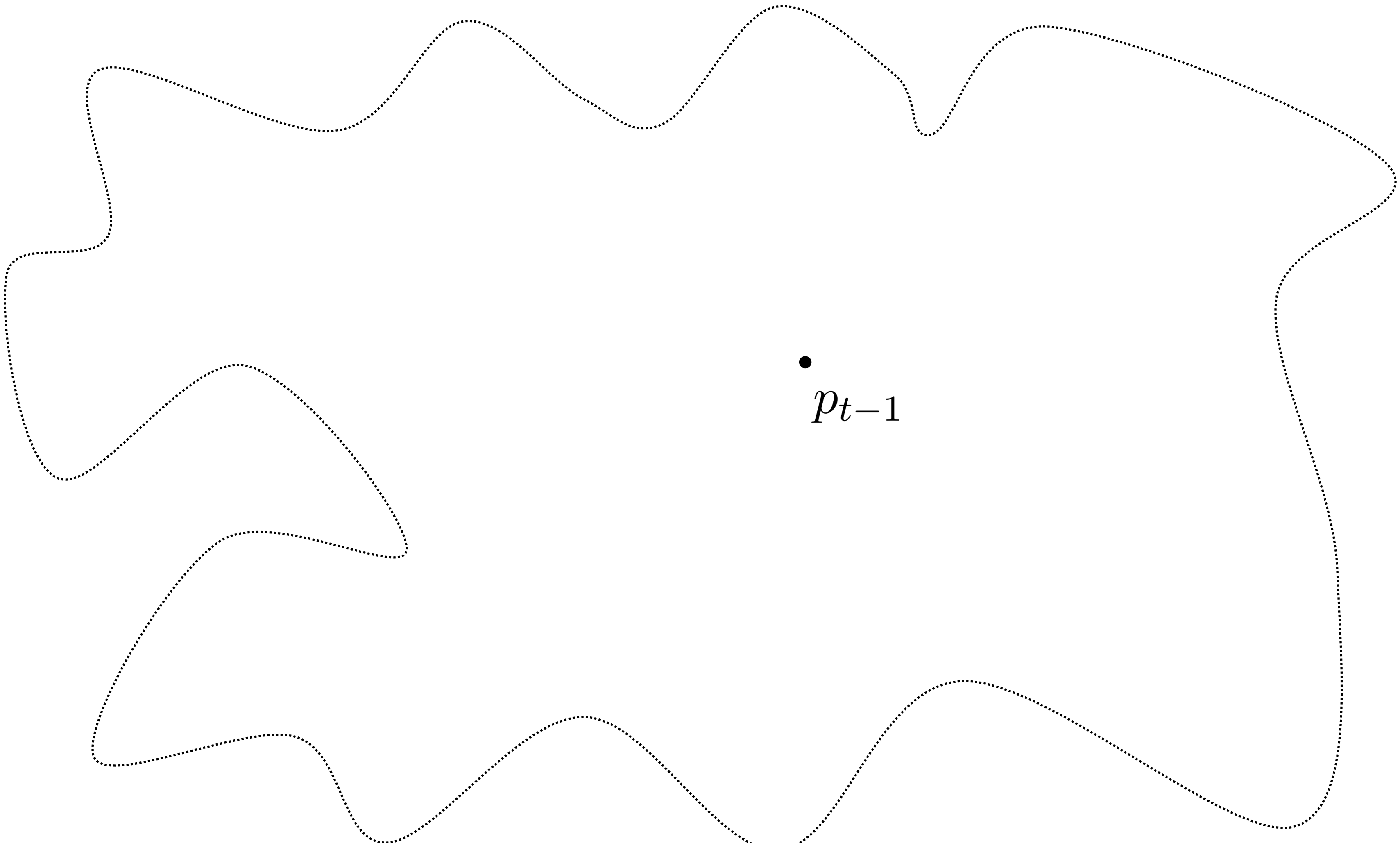
Implicit Transition Models



$$\overbrace{p(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1})}^{\hat{p}_t} = \int p(\boldsymbol{\beta}_t|\boldsymbol{\beta}_{t-1})\overbrace{p(\boldsymbol{\beta}_{t-1}|\boldsymbol{x}_{1:t-1})}^{p_{t-1}}d\boldsymbol{\beta}_{t-1}$$

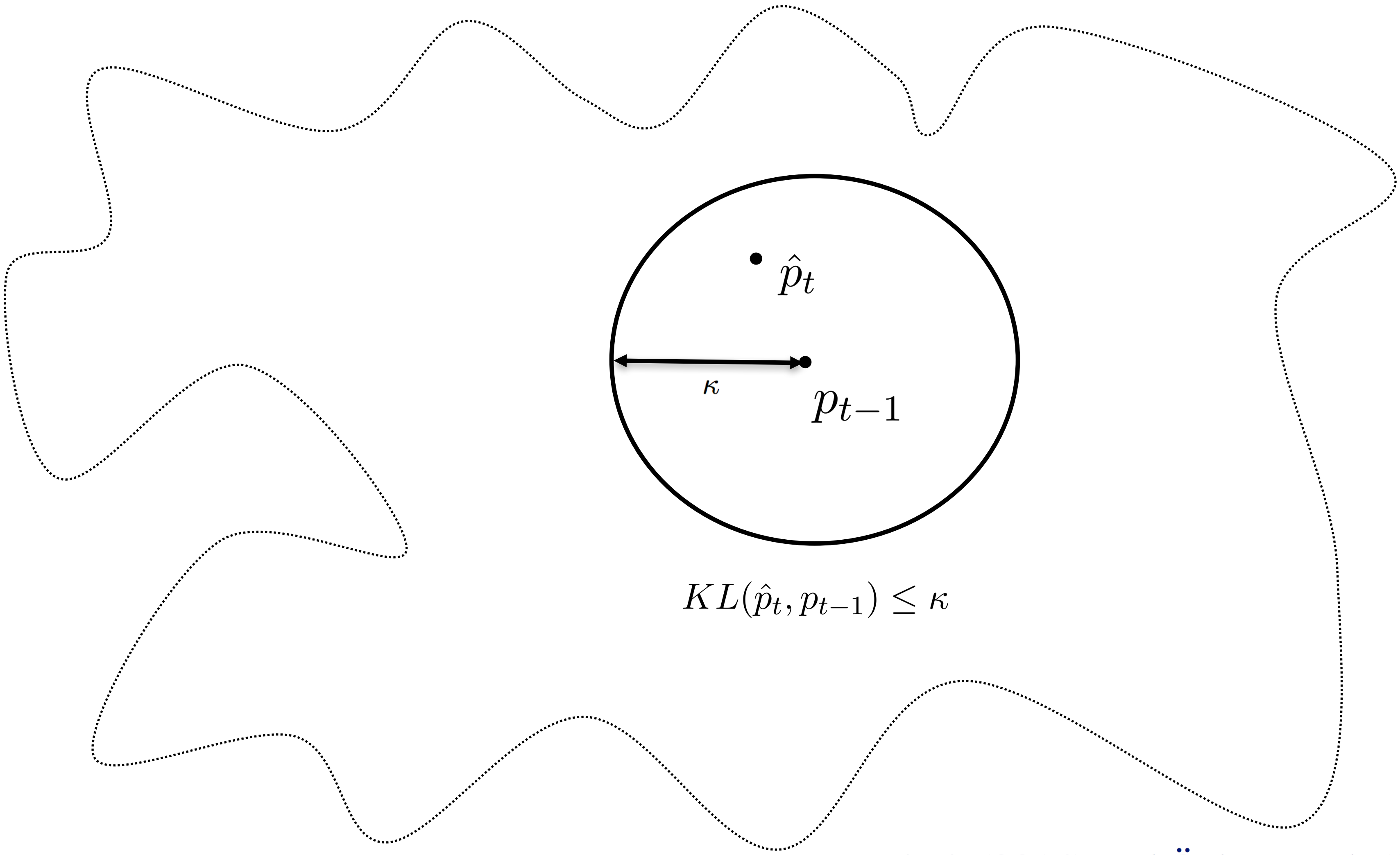
Kárný (2014) and Özkan et al. (2013)

Implicit Transition Models



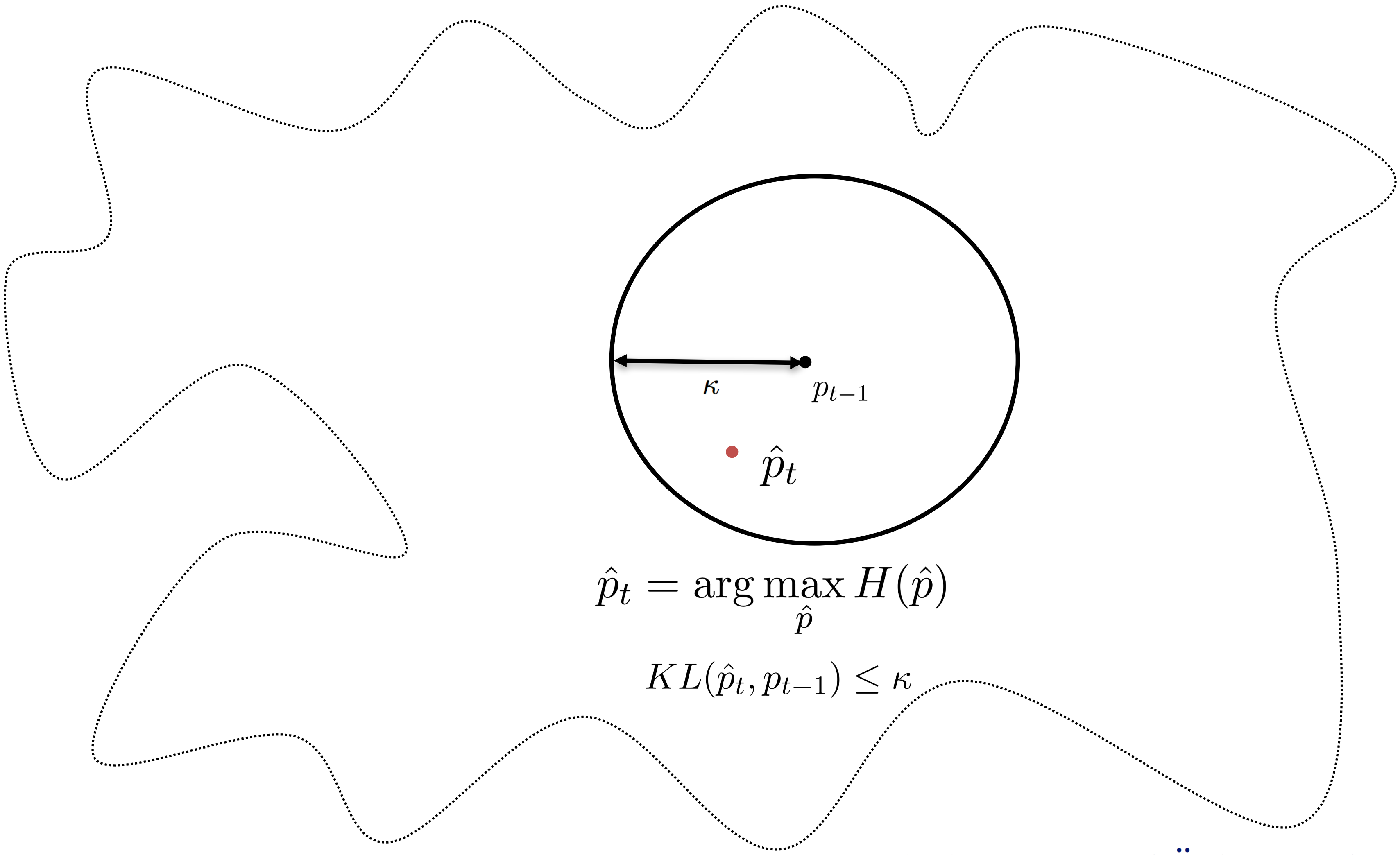
Kárnỳ (2014) and Özkan et al. (2013)

Implicit Transition Models



Kárnỳ (2014) and Özkan et al. (2013)

IMPLICIT TRANSITION MODELS



$$\hat{p}_t = \arg\max_{\hat{p}} H(\hat{p})$$

$$KL(\hat{p}_t, p_{t-1}) \leq \kappa$$

Kárnỳ (2014) and Özkan et al. (2013)

Implicit Transition Models

$$\hat{\boldsymbol{\lambda}}_t = (1-\rho)\boldsymbol{\lambda}_u + \rho\boldsymbol{\lambda}_{t-1}$$

- **Closed-form solution for the Exponential Family**
  - λ natural parameter vector.
  - $\rho \in [0,1]$ is defined by the user.
  - $\rho = 1$ equals $\kappa = 0$ (i.e. maintain all the past data).
  - $\rho = 0$ equals $\kappa = \infty$ (i.e. completely forget past data).

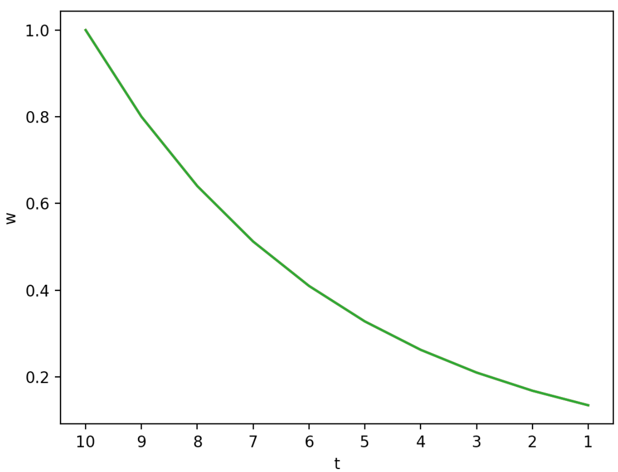
Kárnỳ (2014) and Özkan et al. (2013)

Implicit Transition Models

Exponential Forgetting

$$\ln p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T | \boldsymbol{\beta}, \rho) = \sum_{i=1}^{T} w_i \ln p(\boldsymbol{x}_i | \boldsymbol{\beta})$$

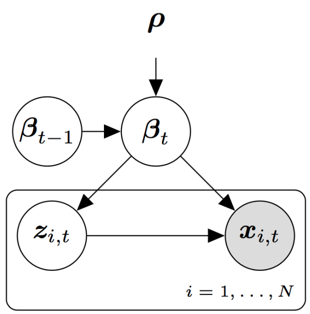
$$w_i = \rho^{T-i} : \rho \in (0, 1]$$



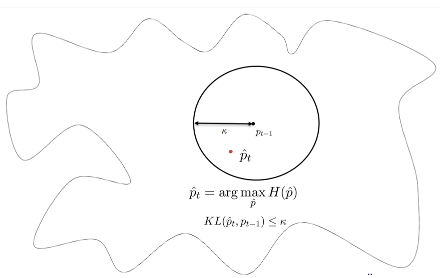
$\Leftrightarrow$

Masegosa et al. 2018

Implicit Transition Models



$$\hat{\boldsymbol{\lambda}}_t = (1-\rho)\boldsymbol{\lambda}_u + \rho\boldsymbol{\lambda}_{t-1}$$



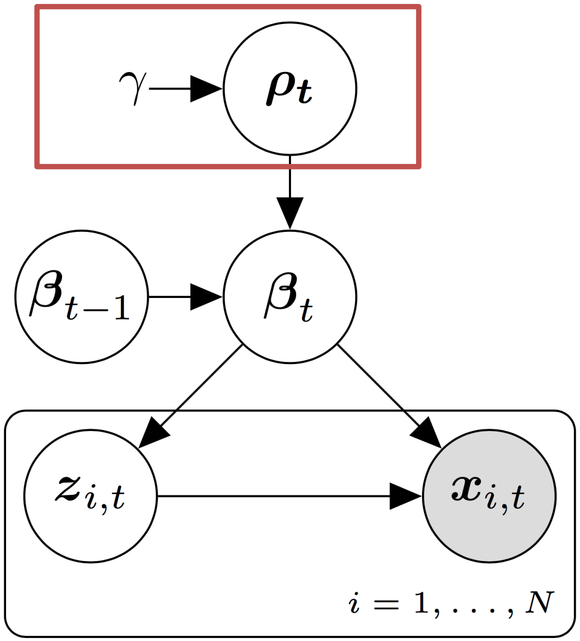
How to choose ρ?

- ρ defines the degree of forgetting.
- Optimal ρ is time dependent.

Hierarchical Power Priors

Masegosa et al. 2018

Hierarchical Power Priors



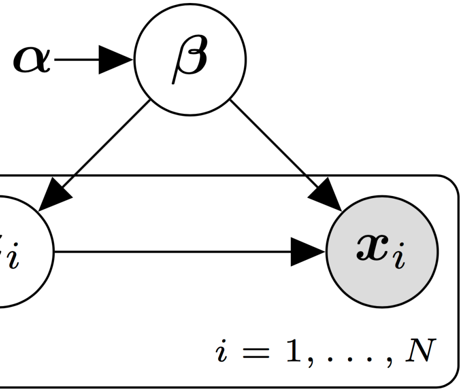
- $\rho_t \sim TrucantedExponential(\gamma)$, $\Omega(\rho_t) = [0, 1]$.
  - ρ_t close to 1 $\rightarrow$ No Drift at time t (i.e. $\beta_{t-1} \approx \beta_t$).
  - ρ_t close to 0 $\rightarrow$ Drift at time t (i.e. $\beta_{t-1} \not\approx \beta_t$).
- $p(\rho_t | \mathbf{x}_{1:t})$ tracks concept drift.

Masegosa et al. 2018

Variational Inference with HPPs

Masegosa et al. 2018

Previous Knowledge

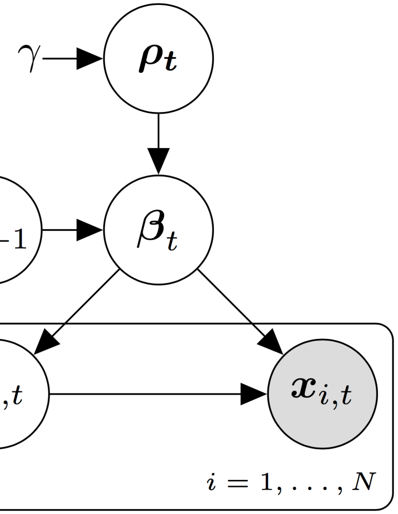


$$q(\boldsymbol{\beta}, \boldsymbol{z} | \boldsymbol{\lambda}, \boldsymbol{\phi}) \approx p(\boldsymbol{\beta}, \boldsymbol{z} | \boldsymbol{x})$$

- **Variational Inference in plain LVMs**
  - $(\boldsymbol{\lambda}^\star, \boldsymbol{\phi}^\star) = \arg\max_{\boldsymbol{\lambda}, \boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\phi} | \mathbf{x}, \boldsymbol{\alpha})$
  - Closed-form gradients for CEF models.

Winn & Bishop, 2005 Hoffman et al., 2013

Hierarchical Power Priors



$$q(\boldsymbol{\beta}_t, \boldsymbol{z}_t, \rho_t | \boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \omega_t) \approx p(\boldsymbol{\beta}_t, \boldsymbol{z}_t, \rho_t | \boldsymbol{x}_1, \ldots, \boldsymbol{x}_t)$$

- **Variational Inference in temporal LVMs**
  - $(\boldsymbol{\lambda}_t^\star, \boldsymbol{\phi}_t^\star, \omega_t^\star) = \arg\max_{\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \omega_t} \mathcal{L}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \omega_t | \mathbf{x}_t, \boldsymbol{\lambda}_{t-1})$
  - **No closed-form gradients.**

Masegosa et al. 2018

HIERARCHICAL POWER PRIORS

Algorithm 1 SVB with Hierarchical Power Priors and Truncated Exponential (SVB-HPP-Exp)

Input: A data batch $\boldsymbol{x}_t$, the variational posterior in previous time step $\boldsymbol{\lambda}_{t-1}$.

Output: $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t)$, a new update of the variational posterior.

1: $\boldsymbol{\lambda}_t \leftarrow \boldsymbol{\lambda}_{t-1}$.

2: $\mathbb{E}_q[\rho_t] \leftarrow 0.5$.

3: Randomly initialize $\boldsymbol{\phi}_t$.

4: **repeat**

5: $\quad (\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t) = \arg\min_{\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t} \mathcal{L}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t | \boldsymbol{x}_t, \mathbb{E}_q[\rho_t]\boldsymbol{\lambda}_{t-1} + (1 - \mathbb{E}[\rho_t])\boldsymbol{\alpha}_u)$

6: $\quad \omega_t = KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_u(\boldsymbol{\beta}_t)\right) - KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})\right) + \gamma$

7: $\quad \mathbb{E}_q[\rho_t] = \frac{1}{(1-e^{-\omega_t})} - \frac{1}{\omega_t}$

8: **until** convergence

9: **return** $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \omega_t)$

$$\mathcal{L}_{HPP} \geq \hat{\mathcal{L}}_{HPP}$$

Masegosa et al. 2018

HIERARCHICAL POWER PRIORS

Algorithm 1 SVB with Hierarchical Power Priors and Truncated Exponential (SVB-HPP-Exp)

Input: A data batch $\boldsymbol{x}_t$, the variational posterior in previous time step $\boldsymbol{\lambda}_{t-1}$.

Output: $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t)$, a new update of the variational posterior.

```
λ_t ← λ_{t-1}.
```

$\boldsymbol{\lambda}_t \leftarrow \boldsymbol{\lambda}_{t-1}$.

$\mathbb{E}_q[\rho_t] \leftarrow 0.5$.

Randomly initialize $\boldsymbol{\phi}_t$.

repeat

$\quad (\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t) = \arg\min_{\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t} \mathcal{L}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t | \boldsymbol{x}_t, \mathbb{E}_q[\rho_t]\boldsymbol{\lambda}_{t-1} + (1 - \mathbb{E}[\rho_t])\boldsymbol{\alpha}_u)$

$\quad \boxed{\omega_t = KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)\,||\,p_u(\boldsymbol{\beta}_t)\right) - KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)\,||\,p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})\right) + \gamma}$

$\quad \mathbb{E}_q[\rho_t] = \frac{1}{(1-e^{-\omega_t})} - \frac{1}{\omega_t}$

until convergence

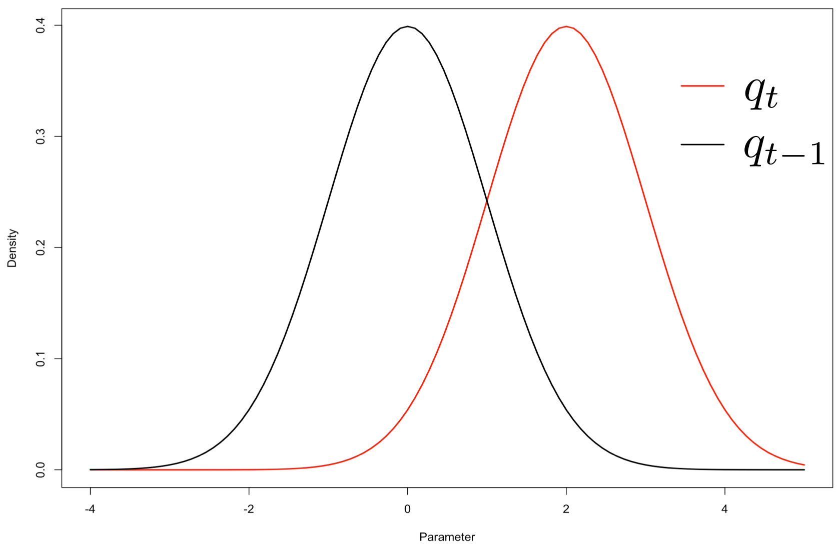
return $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \omega_t)$

$$\mathcal{L}_{HPP} \geq \hat{\mathcal{L}}_{HPP}$$

Masegosa et al. 2018

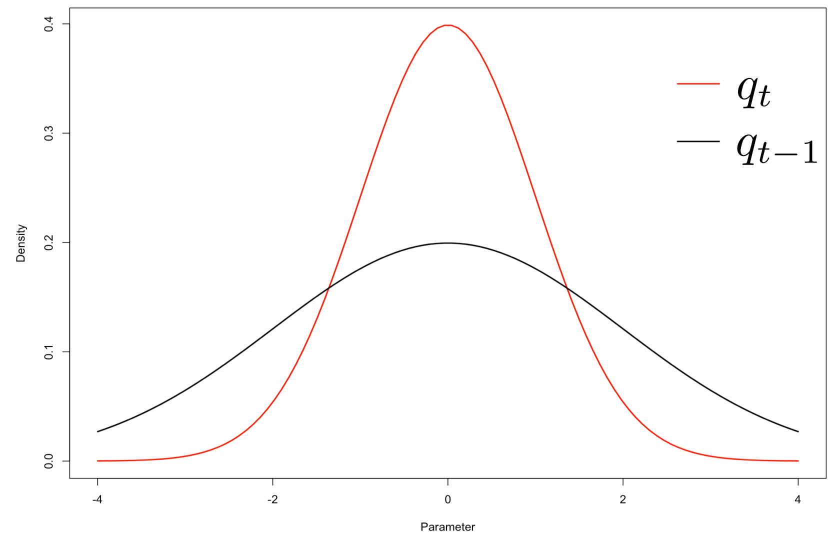
Measuring concept drift

Drift



$KL(q_t, p_u) + \gamma < KL(q_t, q_{t-1})$

No Drift



$KL(q_t, p_u) + \gamma > KL(q_t, q_{t-1})$

A measure of concept drift:

$$\omega_t = KL(q_t, p_u) - KL(q_t, q_{t-1}) + \gamma$$

Masegosa et al. 2018

Hierarchical Power Priors

Algorithm 1 SVB with Hierarchical Power Priors and Truncated Exponential (SVB-HPP-Exp)

Input: A data batch $\boldsymbol{x}_t$, the variational posterior in previous time step $\boldsymbol{\lambda}_{t-1}$.

Output: $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t)$, a new update of the variational posterior.

$\boldsymbol{\lambda}_t \leftarrow \boldsymbol{\lambda}_{t-1}$.

$\mathbb{E}_q[\rho_t] \leftarrow 0.5$.

Randomly initialize $\boldsymbol{\phi}_t$.

repeat

  $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t) = \arg\min_{\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t} \mathcal{L}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t | \boldsymbol{x}_t, \mathbb{E}_q[\rho_t]\boldsymbol{\lambda}_{t-1} + (1 - \mathbb{E}[\rho_t])\boldsymbol{\alpha}_u)$

  $\omega_t = KL\,(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)\,||\,p_u(\boldsymbol{\beta}_t)) - KL\,(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)\,||\,p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})) + \gamma$

  $\boxed{\mathbb{E}_q[\rho_t] = \frac{1}{(1-e^{-\omega_t})} - \frac{1}{\omega_t}}$

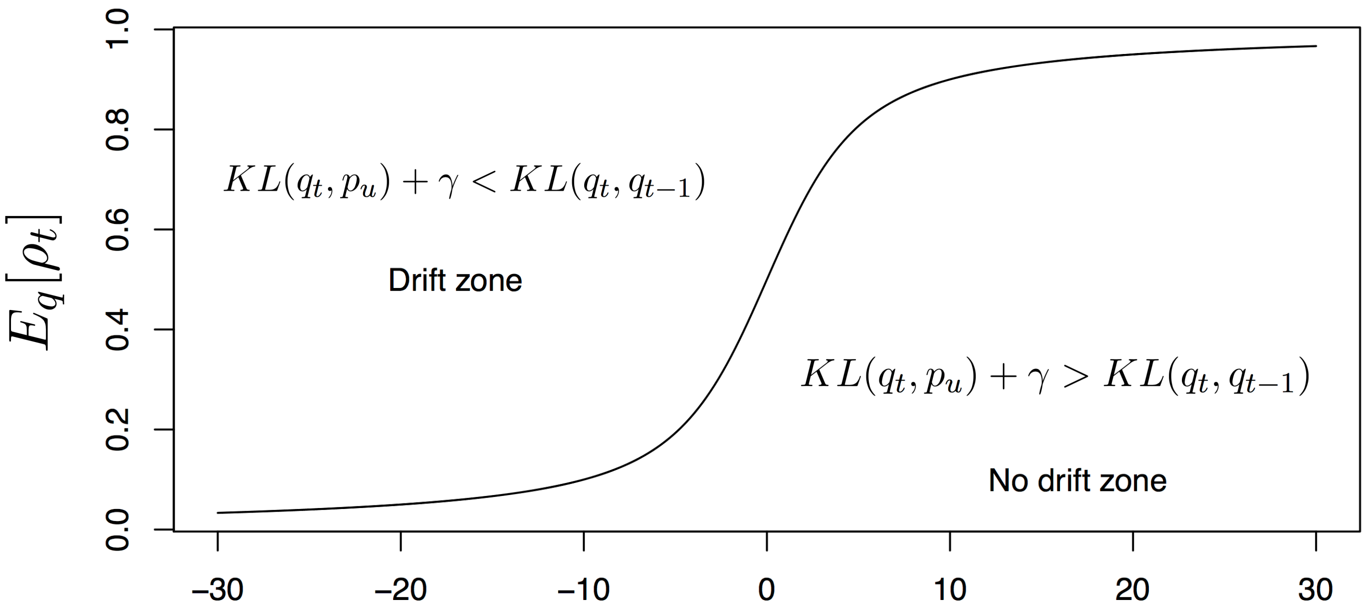
until convergence

return $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \omega_t)$

$$\mathcal{L}_{HPP} \geq \hat{\mathcal{L}}_{HPP}$$

Masegosa et al. 2018

Measuring concept drift

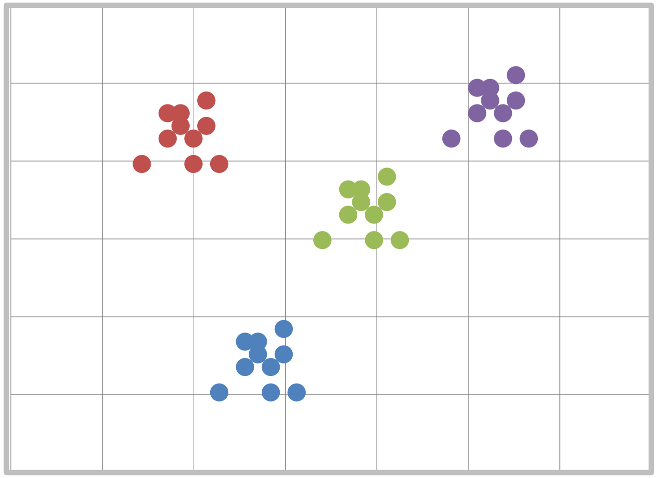


$$\mathbb{E}_q[\rho_t] = \frac{1}{(1-e^{-\omega_t})} - \frac{1}{\omega_t}$$

Masegosa et al. 2018

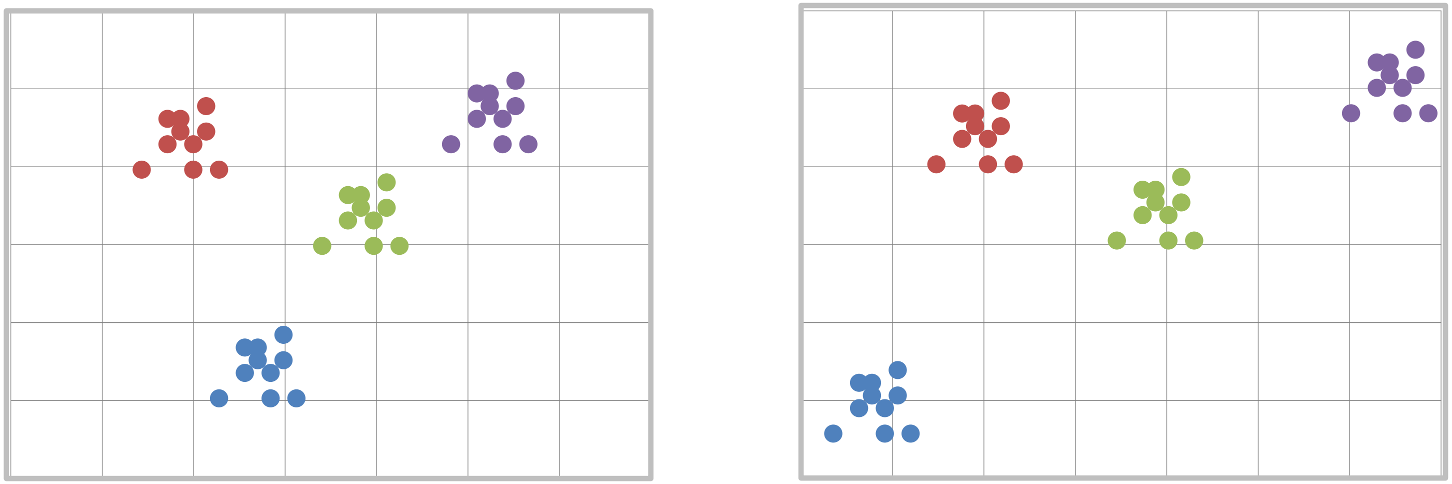
What if only part of the data drifts?

Multiple HPPs



$$\mathbf{x}_{t-1}$$

Multiple HPPs



$\mathbf{x}_{t-1}$ $\mathbf{x}_t$

- **Multiple HPPs**
  - Place independent $\rho_{k,t}$ for each parameter of the model.
  - Closed-form Variational inference.

Masegosa et al. 2018

Experimental Evaluation

Experimental Evaluation

Different Domains and Different Models:

- Energy Data Set with a Linear Regression Model.
- Finance Data Set with a MoG Model.
- GPS Data Set with a MoG Model.
- Text Data Set with a LDA Model.

Compare with SOTA Methods:

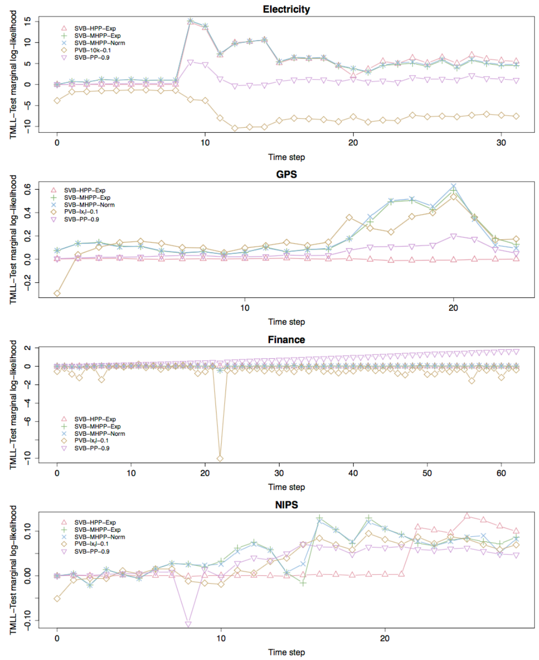
- SVB (Broderick et al. (2013)): Incremental Bayesian Updating
- SVB-PP (Broderick et al. (2013),Gaber et al. (2005)): Bayesian Updating with (fixed) Exponential Forgetting.
- PVB (McInerney et al. (2015)): Population Variational Bayes.

EXPERIMENTAL EVALUATION

Test Marginal Log-Likelihood (Perplexity)

DATA SET	SVB	PVB				SVB-PP		SVB-HPP	SVB-MHPP	
		(1)	(2)	(3)	(4)	$\rho = 0.9$	$\rho = 0.99$	EXP	EXP	NORM
ELECTRICITY	-44.91	-51.01	-52.19	-51.11	-61.70	-43.92	-44.80	-40.05	-40.02	**-39.91**
GPS	-1.98	-2.10	-2.77	-1.97	-4.49	-1.94	-1.97	-1.97	**-1.86**	**-1.86**
FINANCE	-19.84	-22.29	-22.57	-20.40	-20.73	**-19.05**	-19.78	-19.83	-19.83	-19.82
NIPS	-4.07	-4.04*	-4.21*	-4.01	-4.12	-4.02	-4.06	-4.01	**-4.00**	**-4.00**

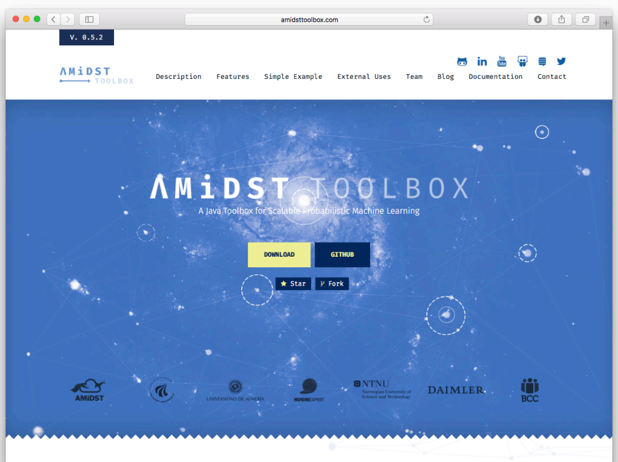
- **Summary of the evaluation:**
  - SVB-MHPP is the most robust approach.
  - Adaptive forgetting mechanisms are usually needed.
  - Concept drift usually affects only a part of the model.



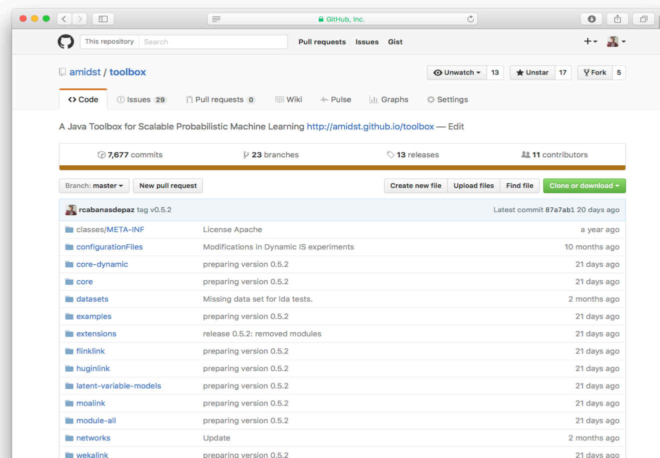
- **Summary of the evaluation:**
  - SVB-MHPP is the most robust approach.
  - Adaptive forgetting mechanisms are usually needed.
  - Concept drift usually affects only a part of the model.

AMIDST TOOLBOX

A Java Toolbox for Scalable Probabilistic Machine Learning



www.amidsttoolbox.com



github.com/amidst/toolbox

Apache
License 2.0

Bayesian Modeling of Concept Drift in Deep Learning

Bayesian Reasoning:

- Mainly Conjugate and Linear Models.
- Complex Inference.
- Unified Framework for model building, inference, prediction and decision making.
- Explicit accounting for uncertainty and variability of outcomes.
- Robust to overfitting; tools for model selection and composition.

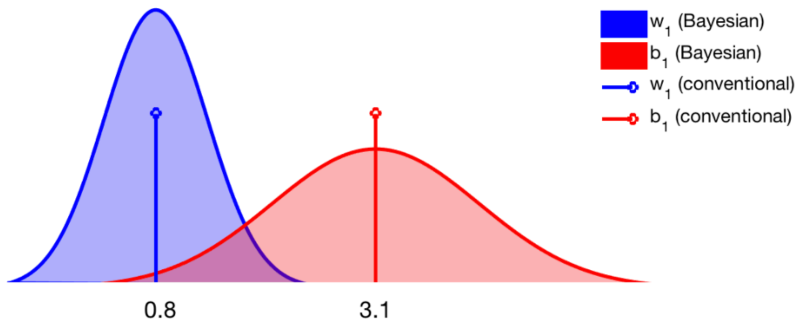
Deep Learning:

- Rich non-linear models for classification and sequence prediction.
- Scalable learning using stochastic approximation and conceptually simple.
- Easily composable with other gradient-based methods.
- Only point estimates.
- Hard to score models, do selection and complexity penalisation.

Bayesian Deep Learning

$$p(\boldsymbol{w}|\boldsymbol{y},\boldsymbol{x}) = \frac{p(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})p(\boldsymbol{w})}{\int p(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})p(\boldsymbol{w})d\boldsymbol{w}}$$

[Using Modern Variational Inference Methods]

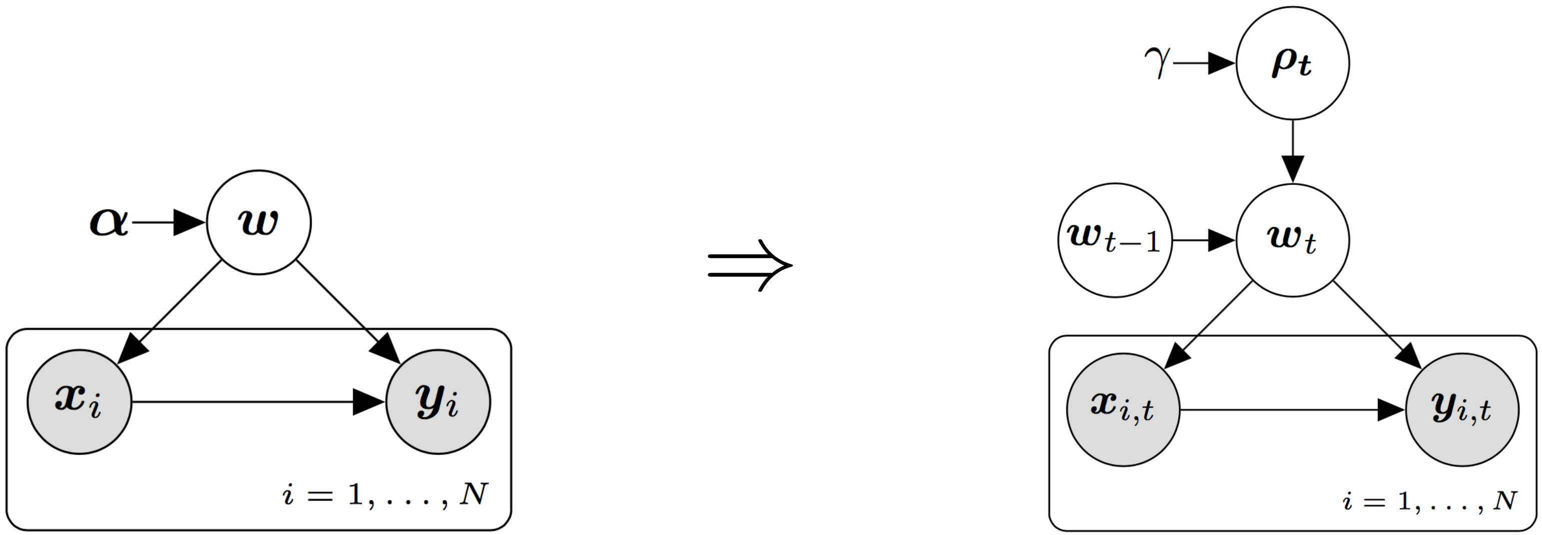


Variational Deep Learning

$$p(\boldsymbol{w}|\boldsymbol{y}, \boldsymbol{x}) \approx \prod_i q(w_i|\lambda_i)$$

[q is inside exponential family]

BAYESIAN DEEP LEARNING



Bayesian Modeling of Concept Drift:

- Learn DNNs from non-stationary data streams.
- Address Catastrophical Forgetting.
- Help in Domain Adaptation Problems.

Hierarchical Power Priors

Algorithm 1 SVB with Hierarchical Power Priors and Truncated Exponential (SVB-HPP-Exp)

Input: A data batch $\boldsymbol{x}_t$, the variational posterior in previous time step $\boldsymbol{\lambda}_{t-1}$.
Output: $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t)$, a new update of the variational posterior.

1: $\boldsymbol{\lambda}_t \leftarrow \boldsymbol{\lambda}_{t-1}$.
2: $\mathbb{E}_q[\rho_t] \leftarrow 0.5$.
3: Randomly initialize $\boldsymbol{\phi}_t$.
4: **repeat**
5: $\quad (\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t) = \arg\min_{\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t} \mathcal{L}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t | \boldsymbol{x}_t, \mathbb{E}_q[\rho_t]\boldsymbol{\lambda}_{t-1} + (1 - \mathbb{E}[\rho_t])\boldsymbol{\alpha}_u)$
6: $\quad \omega_t = KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_u(\boldsymbol{\beta}_t)\right) - KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})\right) + \gamma$
7: $\quad \mathbb{E}_q[\rho_t] = \frac{1}{(1-e^{-\omega_t})} - \frac{1}{\omega_t}$
8: **until** convergence
9: **return** $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \omega_t)$

$$\mathcal{L}_{HPP} \geq \hat{\mathcal{L}}_{HPP}$$

Masegosa et al. 2018

Thanks for your attention

@ andresmasegosa@ual.es

https://github.com/andresmasegosa/slides

AMiDST TOOLBOX

Exponential Forgetting

$$\arg\max_{\beta} \sum_{i=1}^{T} \ln p(\boldsymbol{x}_i|\boldsymbol{\beta}) \quad \Longrightarrow \quad ESS_{ML} = \sum_{i=1}^{T} 1 = T$$

$$\arg\max_{\beta} \sum_{i=1}^{T} \rho^{T-i} \ln p(\boldsymbol{x}_i|\boldsymbol{\beta}) \quad \Longrightarrow \quad ESS_{\rho} = \sum_{i=1}^{T} \rho^{T-i} = \frac{1-\rho^T}{1-\rho} \longrightarrow \frac{1}{1-\rho}$$

Exponential Forgetting:

- Equivalent Sample size (ESS) depends only on ρ.
- Sliding window considering the last $\frac{1}{1-\rho}$ samples of the stream.