UNIVERSITAT ROVIRA i VIRGILI

UNIVERSITAT DE BARCELONA

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

MAI TFM

A Comparative Study Of Different Gradient Approximations For Restricted Boltzmann Machines

Lihong Ji

Supervisor: Enrique Romero Merino
co-Supervisor: Ferran Mazzanti Castrillejo

Master Degree in Artificial Intelligence
29/06/2023

Agenda

- Introduction & State of the Art
- Energy-based models & Restricted Boltzmann Machines(RBMs)
- Training RBMs
- Experimental results & Analysis
- Conclusions

1. Introduction & State of the Art

- Background
- Motivation
- Objectives
- State of the Art

Introduction -> Background

A pioneer generative neural network

First proposed in 1987[1] -> **hard to train**

Contrastive Divergence(CD) proposed in 2002 by Geoffrey Hinton[2]

Widely used in Classification, Dimension Reduction, Feature Learning etc..

The most importantly, RBMs demonstrate **the potential of the neural network**.

Introduction -> Motivation & Objectives

Motivation:

CD-series algorithms -> **bias** -> the trade-off between efficiency and precision

RBMs -> strongly probability-supported model -> **explanability**

Stochastic process in Bayesian Machine Learning -> Learn more (personally)

UNIVERSITAT ROVIRA i VIRGILI · UNIVERSITAT DE BARCELONA · UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Introduction -> Motivation & Objectives

Objectives:

Implement RBMs **from the scratch**

Manage the RBM training -> Different approximation algorithms
-> KL, NLL, Prob_sum, Entropy, Prob_distribution

Compare the approximation with the **exact results**

State of the Art

Binary RBMs:

$$E(v,h) = -(vWh^T + \alpha v^T + \beta h^T)$$

Gaussian-Bernoulli RBMs:

$$E_\theta(v,h) = \frac{1}{2}(\frac{v-\mu}{\sigma})^T(\frac{v-\mu}{\sigma}) - (\frac{v}{\sigma^2})Wh - b^Th$$

State of the Art

The exact gradient of logP(v) via Gibbs Sampling is[3]:

$$\nabla_\theta \log P(v^0) = -\sum_h P(h|v^0)\,\nabla_\theta E(v^0,h) + \mathbb{E}_{P(v^k|v^0)}\Big[\sum_h P(h|v^k)\,\nabla_\theta E(v^k,h)\Big] + \underbrace{\boxed{\mathbb{E}_{P(v^k|v^0)}[\nabla_\theta \log P(v^k)]}}_{\textbf{Bias expression}}$$

State of the Art

Based on the mathematical expression of the bias, Bengio and Fischer propose 2 sorts of the bias bound:

Bengio's bound[3]**:**

$$\mathbb{E}_{P(v^k|v^0)}[\nabla_\theta \log P(v^k)] \leq 2^m(1-2^m 2^n sig(-\alpha)^m sig(-\beta)^n)^k$$

where

$$\alpha = \max_j(\textstyle\sum_i |w_{ij}| + |b_j|) \qquad \beta = \max_i(\textstyle\sum_j |w_{ij}| + |c_j|)$$

and

m -> visible dimension, n-> hidden dimension

b -> visible layer bias c-> hidden layer bias

k -> Gibbs sampling time

State of the Art

Based on the mathematical expression of the bias, Bengio and Fischer propose 2 sorts of the bias bound:

Fischer's bound[4]:

$$\mathbb{E}_{P(v^k|v^0)}[\nabla_\theta \log P(v^k)] \leq (1 - e^{-(m+n)\triangle})^k$$

Specifically,

$$\triangle = \max(\max_{l\in(1,m)} \vartheta_l, \max_{l\in(1,n)} \zeta_l)$$

where

$$\vartheta_l = \max(|\sum_i^n I_{(w_{il}>0)} w_{il} + \alpha_l|, |\sum_i^n I_{(w_{il}<0)} w_{il} + \alpha_l|)$$

and

$$\zeta_l = \max(|\sum_j^m I_{(w_{lj}>0)} w_{lj} + \beta_l|, |\sum_j^m I_{(w_{lj}<0)} w_{lj} + \beta_l|)$$

State of the Art

Both of the bounds demonstrate the relevance of controlling the absolute values of RBM parameters.

→ increasing Gibbs sampling time k, bias -> 0

→ increasing absolute values of weights w, α, β, bias ⬆️ (weight decay works)

→ increasing the variable number m, n, bias ⬆️

2. Energy-based models & Restricted Boltzmann Machines(RBMs)

- Energy-based models
- RBMs Representation
- RBMs Inference
- RBMs Learning

What is an Energy-based model ?

Energy function → defined over the input → specify **P(v)**

For the most cases, the P(v) is an **exponential** function of energy.

$$P(v) = \frac{e^{-Energy(v)}}{Z}$$

$$\Big\Downarrow \quad \text{Add hidden variables, h}$$

$$P(v,h) = \frac{e^{-Energy(v,h)}}{Z}$$

$$\Big\Downarrow$$

$$P(v) = \frac{\sum_h e^{-Energy(v,h)}}{Z}$$

UNIVERSITAT ROVIRA i VIRGILI · UNIVERSITAT DE BARCELONA · UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Energy-based model Optimization

The idea is to assign lower energy to the observed data.

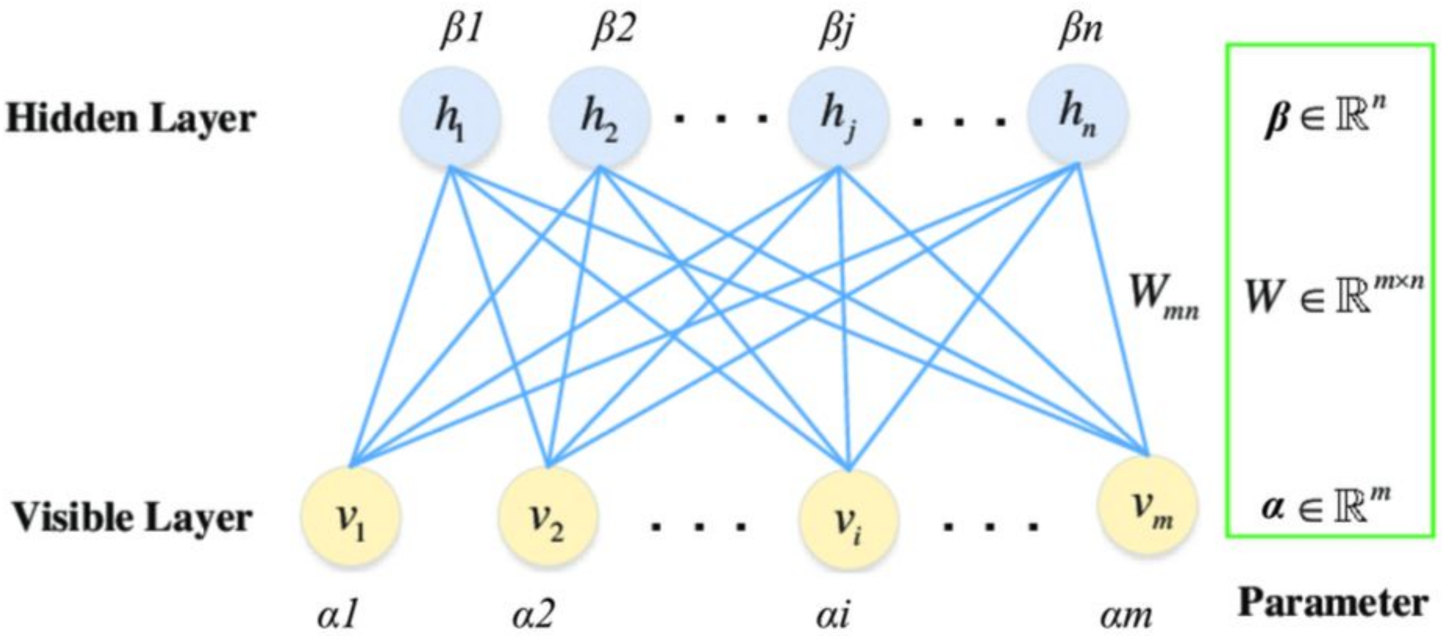
-> equivalent to minimize P(v) -> logP(v)

$$\log P(v) = \log \sum_{h} exp(-E(v,h)) - \log \sum_{v,h} exp(-E(v,h))$$

$$\Downarrow$$

$$\frac{\partial \log P(v)}{\partial \theta} = -\mathbb{E}_{P(h|\hat{v})}[\frac{\partial E(\hat{v},h)}{\partial \theta}] + \mathbb{E}_{P(v)}[\mathbb{E}_{P(h|v)}[\frac{\partial E(v,h)}{\partial \theta}]]$$

RBMs Representation



Energy function $\rightarrow E(v,h) = -(vWh^T + \alpha v^T + \beta h^T)$

RBMs Inference

$$P(h \mid v) = \frac{1}{1 + exp\{-(vW + \beta)\}}$$

$$P(v \mid h) = \frac{1}{1 + exp\{-(hW^T + \alpha)\}}$$

$$P(v) = \frac{1}{Z} exp\{\alpha^T v + \log(1 + exp\{vW + \beta\})\}$$

RBMs Learning

$$\frac{\partial \log P(v)}{\partial W_{ij}} = -\hat{v}_i P(h_j = 1 \mid \hat{v}) + \sum_v P(v) P(h_j = 1 \mid v) v_i$$

$$\frac{\partial \log P(v)}{\partial \alpha_i} = -\hat{v}_i + \sum_v P(v) v_i$$

$$\frac{\partial \log P(v)}{\partial \beta_j} = -P(h_j = 1 \mid \hat{v}) + \sum_v P(v) P(h_j = 1 \mid v)$$

$$\Downarrow$$

Gradient ascend

3. Training RBMs

- Gibbs Sampling
- Contrastive Divergence (CD)
- Persistent Contrastive Divergence (PCD)
- Weighted Contrastive Divergence (WCD)
- Weighted Persistent Contrastive Divergence(WPCD)
- Parallel Tempering

Training RBMs -> Gibbs Sampling

model expectation -> $\mathbb{E}_{P(v)}[\mathbb{E}_{P(h|v)}[\frac{\partial E(v,h)}{\partial \theta}]]$

$$\sum_{v \sim P(v)} \mathbb{E}_{P(h|v)}[\frac{\partial E(v,h)}{\partial \theta}] = \sum_{\boxed{v \sim P(v)}} \sum_{h=[0,1]} P(h|v) \frac{\partial E(v,h)}{\partial \theta}$$

Gibbs Sampling

$$P(v) \sim P(v|h)$$

Training RBMs -> Contrastive Divergence(CD)

Utilize Gibbs Sampling for k times -> v0, vk

$$\nabla_w - \log P(v) = P(h=1 \mid v_0)v_0 - P(h=1 \mid v_k)v_k$$

$$\nabla_\alpha - \log P(v) = v_0 - v_k$$

$$\nabla_\beta - \log P(v) = P(h=1 \mid v_0) - P(h=1 \mid v_k)$$

Training RBMs -> Persistent Contrastive Divergence(PCD)

Instead of creating new Markov chain during each updating, in PCD a persistent chain is utilized.

v0 comes from the **data**

vk comes from the **persistent chain**

Training RBMs -> Weighted Contrastive Divergence(WCD)
Weighted Persistent Contrastive Divergence(WPCD)

Recall **model expectation**:

exact gradient

$$\sum_v \boxed{P(v)} \mathbb{E}_{P(h|v)}[\frac{\partial E(v,h)}{\partial \theta}]$$

CD approximation

$$\sum_i^{N_B} \boxed{\frac{1}{N_B}} \mathbb{E}_{P(h|v_k^i)}[\frac{\partial E(v_k^i,h)}{\partial \theta}]$$

$$\Downarrow$$

$$\sum_{i=1}^{N_B} \overline{P}(v_k^i) \mathbb{E}_{P(h|v_k^i)}[\frac{\partial E(v_k^i,h)}{\partial \theta}]$$

Training RBMs -> Weighted Contrastive Divergence(WCD)
Weighted Persistent Contrastive Divergence(WPCD)

$$\overline{P}(v_i) = \frac{P(v_i)}{\sum_j^{N_B} P(v_j)} = \frac{\sum_h exp\{-E(v_i,h)\}}{\sum_j^{N_B} \sum_h \{-E(v_j,h)\}}$$

Partition function is eliminated -> efficient to compute $\overline{P}(v_i)$

WPCD = WCD + persistent chain

Training RBMs -> Parallel Tempering(PT)

Designed for more global exploration -> run multiple models at different temperatures

$$P_r(v,h) = \frac{1}{Z_r} e^{-\frac{1}{T_r} E(v,h)}$$

$\downarrow$

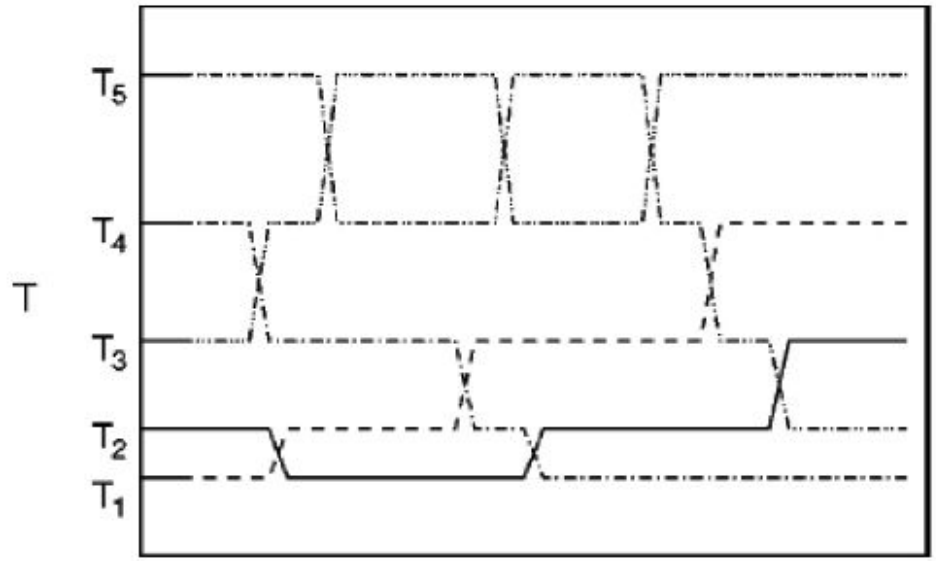
1/Tr -> [0, …., 1]

$\downarrow$

RBMs chains

Training RBMs -> Parallel Tempering(PT)

Exchange rate -> $min\left\{1, \exp\left\{(\frac{1}{T_r} - \frac{1}{T_{r-1}})(E(v_r, h_r) - E(v_{r-1}, h_{r-1}))\right\}\right\}$



In the implementation, exchange first in **odd** index and then **even** index.

4. Experimental Results & Analysis

- Datasets
- Evaluation Metrics
- Hyperparameter setup
- Results
- Analysis

UNIVERSITAT ROVIRA i VIRGILI UNIVERSITAT DE BARCELONA UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Datasets

	Size	Dimension	Type
Bars-and-Stripes-3x3	14	9	[0,1]
Bars-and-Stripes-4x4	30	16	[0,1]
Labeled-Shifter-4-11	48	11	[0,1]
Labeled-Shifter-5-13	96	13	[0,1]

Evaluation Metrics

KL divergence -> $D_{KL}(P||Q) = \sum_x P(x) \log(\frac{P(x)}{Q(x)})$

Negative Log Likelihood -> $-\frac{1}{N}\sum_n logQ(x_n)$

Probability sum -> $\sum_T P(x)$

Entropy Percentage -> $\frac{\sum_T P(x) \log P(x)}{\log \frac{1}{N}}$

Hyperparameter setup

Variable Type -> [0,1] or [-1,1]?

Inference for RBMs with [-1,1] :

$$P(h \mid v) = \frac{1}{1 + exp\left\{2 * (-vW - \beta)\right\}}$$

$$P(v \mid h) = \frac{1}{1 + exp\left\{2 * (-hW^T - \alpha)\right\}}$$

$$P(v) = \frac{1}{Z} \exp\left\{\alpha v^T + \log(exp(-vw - \beta) + exp(vw + \beta))\right\}$$

Hyperparameter setup

Variable Type -> repeat **10** times

With the weight initialization in different σ, KL are compared(CD1&WCD)

$N_h = 3N_v$

Group	$\sigma = 1$	$\sigma = 0.1$	$\sigma = 0.01$	$\sigma = 0.001$
$[0,1]\&CD1$	0.591(0.052)	0.511(0.036)	0.479(0.029)	0.505(0.052)
$[0,1]\&WCD1$	0.1416(0.052)	0.130(0.036)	0.137(0.029)	0.154(0.052)
$[-1,1]\&CD1$	$\boxed{1.709(0.366)}$	0.101(0.638)	0.066(0.032)	0.083(0.041)
$[-1,1]\&WCD1$	0.062(0.013)	0.076(0.020)	0.066(0.031)	0.059(0.025)

$N_h = 5N_v$

Group	$\sigma = 1$	$\sigma = 0.1$	$\sigma = 0.01$	$\sigma = 0.001$
$[0,1]\&CD1$	0.658(0.065)	0.548(0.060)	0.548(0.051)	0.551(0.033)
$[0,1]\&WCD1$	0.153(0.005)	0.162(0.036)	0.160(0.029)	0.150(0.052)
$[-1,1]\&CD1$	$\boxed{4.702(2.252)}$	0.087(0.032)	0.091(0.022)	0.071(0.026)
$[-1,1]\&WCD1$	$\boxed{0.001(0.001)}$	0.089(0.097)	0.14(0.032)	0.10(0.041)

Obviously, RBMs with input [0,1] are much more **stable**!

Hyperparameter setup

Variable Type -> **[0,1]**

Dimension of **hidden** layer -> **5*Nv**

Learning Rate Decay -> **linear decay** (from default value to **1e-6**)

Weight Decay -> **2.5e-5**[5]

Parameter Configuration -> repeat **30** times

BS3x3

Opt	sample-type	lr	lr-decay	gibbs-sampling-num	chain-num
CDK	gibbs-sampling	0.005	False	1	None
PCD	gibbs-sampling	0.01	True	1	None
WCD	gibbs-sampling	0.1	False	1	None
WPCD	gibbs-sampling	0.03	True	1	None
PT	parallel-tempering	0.005	True	1	2

BS4x4

Opt	sample-type	lr	lr-decay	gibbs-num	chain-num
CDK	gibbs-sampling	0.005	False	1	None
PCD	gibbs-sampling	0.01	True	1	None
WCD	gibbs-sampling	0.1	False	1	None
WPCD	gibbs-sampling	0.03	True	1	None
PT	parallel-tempering	0.005	True	1	2

Parameter Configuration -> repeat **30** times

LS4–11

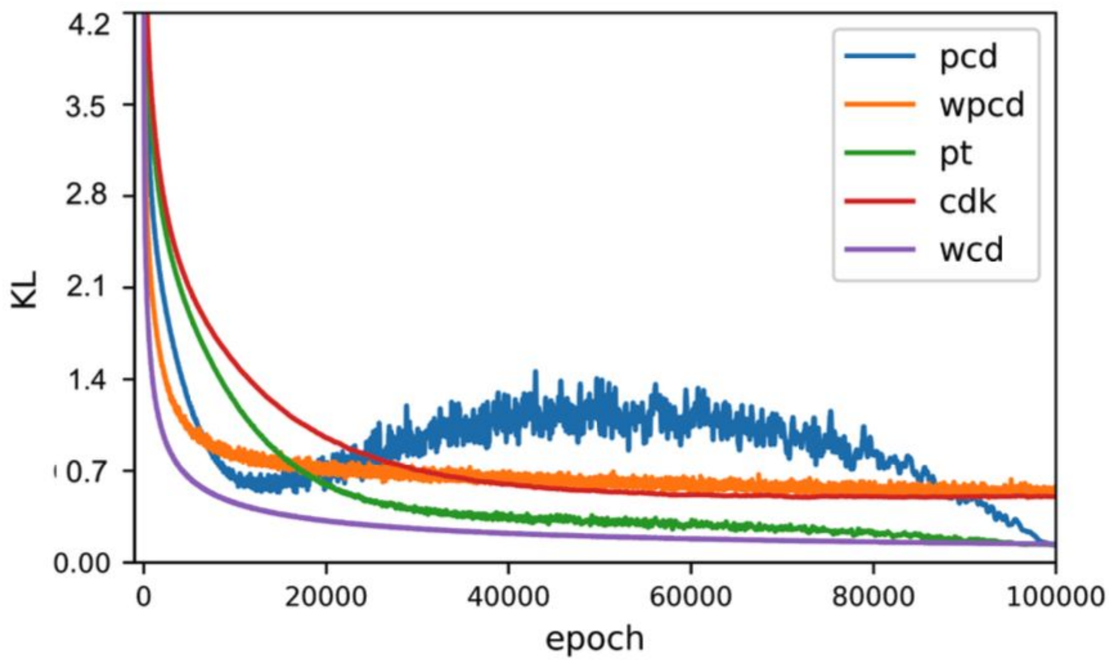
Opt	sample-type	lr	lr-decay	gibbs-num	chain-num
CDK	gibbs-sampling	0.03	False	1	None
PCD	gibbs-sampling	0.05	True	1	None
WCD	gibbs-sampling	0.2	False	1	None
$WPCD$	gibbs-sampling	0.05	True	1	None
PT	parallel-tempering	0.03	True	1	2

LS5-13

Opt	sample-type	lr	lr-decay	gibbs-num	chain-num
CDK	gibbs-sampling	0.03	False	1	None
PCD	gibbs-sampling	0.03	True	1	None
WCD	gibbs-sampling	0.2	False	1	None
$WPCD$	gibbs-sampling	0.05	True	1	None
PT	parallel-tempering	0.03	True	1	2

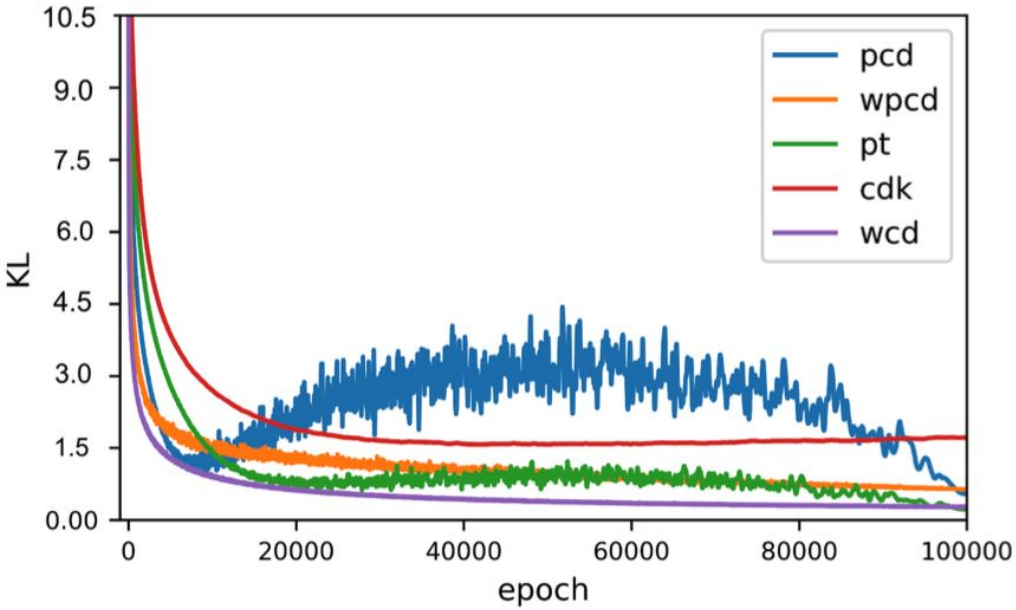
BS3x3

	KL	NLL	Prob-sum	Entropy-Percentage
$CD1$	0.45528 (0.0476)	-3.09946 (0.05310)	0.9238 (0.00189)	0.8647 (0.01495)
PCD	0.0809 (0.02586)	-2.71694 (0.01599)	0.96548 (0.00284)	0.94561 (0.00886)
WCD	0.0541(0.0029)	-2.6931(0.0029)	0.9485(0.0029)	0.9671(0.0018)
$WPCD$	0.2315 (0.00678)	-3.06958 (0.00557)	0.8098 (0.00488)	0.8785 (0.0036)
PT	0.0875(0.088)	-2.7666(0.048)	0.9782(0.002)	0.9636(0.0229)
$CD10^*$	0.0501(0.0051)	-2.6823(0.0051)	0.9732(0.0026)	0.9743(0.0024)
$WCD10^*$	0.0595(0.0072)	-2.6996(0.0072)	0.9497(0.0053)	0.9669(0.004)



UNIVERSITAT ROVIRA i VIRGILI · UNIVERSITAT DE BARCELONA · UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

BS4x4

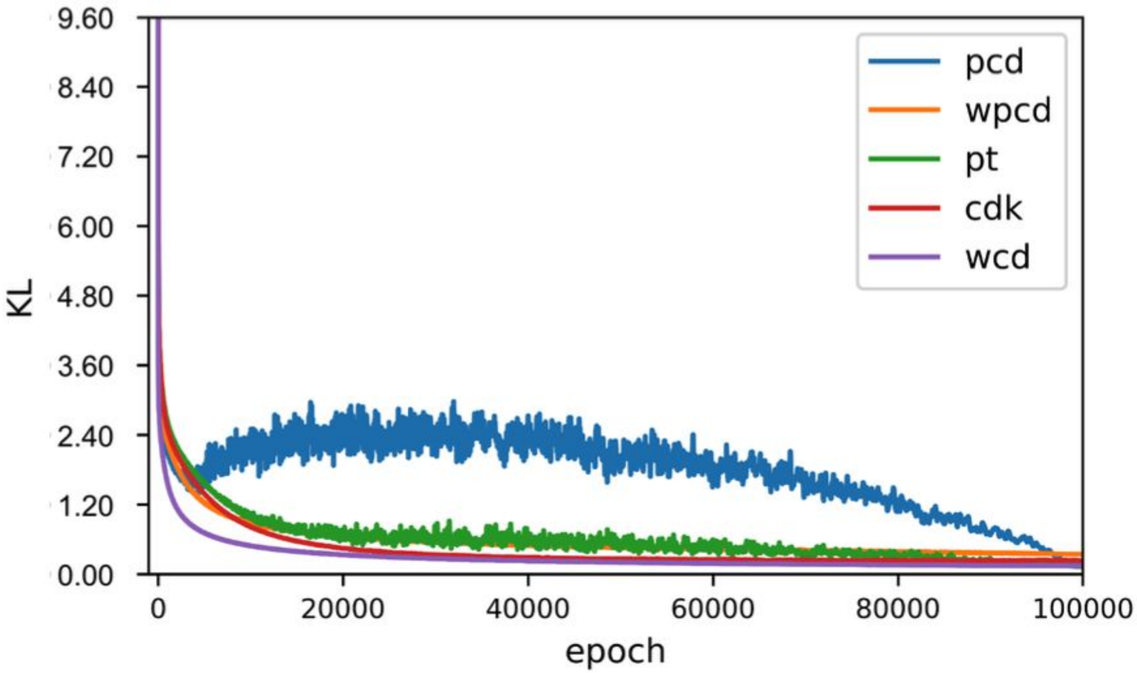


	KL	NLL	Prob-sum	Entropy-Percentage
CDK	1.5793(0.0809),	-4.9805(0.0809),	0.908(0.0072),	0.7865(0.0107)
PCD	0.6916(0.2118),	-4.0928(0.2118),	0.9718(0.0071),	0.8023(0.0555)
WCD	0.1018(0.0024),	-3.503(0.0024),	0.9088(0.0029),	0.9327(0.0017)
$WPCD$	0.3727(0.012),	-3.7739(0.012),	0.6907(0.0075),	0.7653(0.0063)
PT	0.2782(0.0691),	-3.6794(0.0691),	0.9072(0.0121),	0.8805(0.0188)
$CD10^*$	0.2743(0.0116)	-3.6755(0.0116)	0.9412(0.0032)	0.913(0.0026)
$WCD10^*$	0.102(0.0149)	-3.5032(0.0149)	0.9101(0.0107)	0.9332(0.0087)

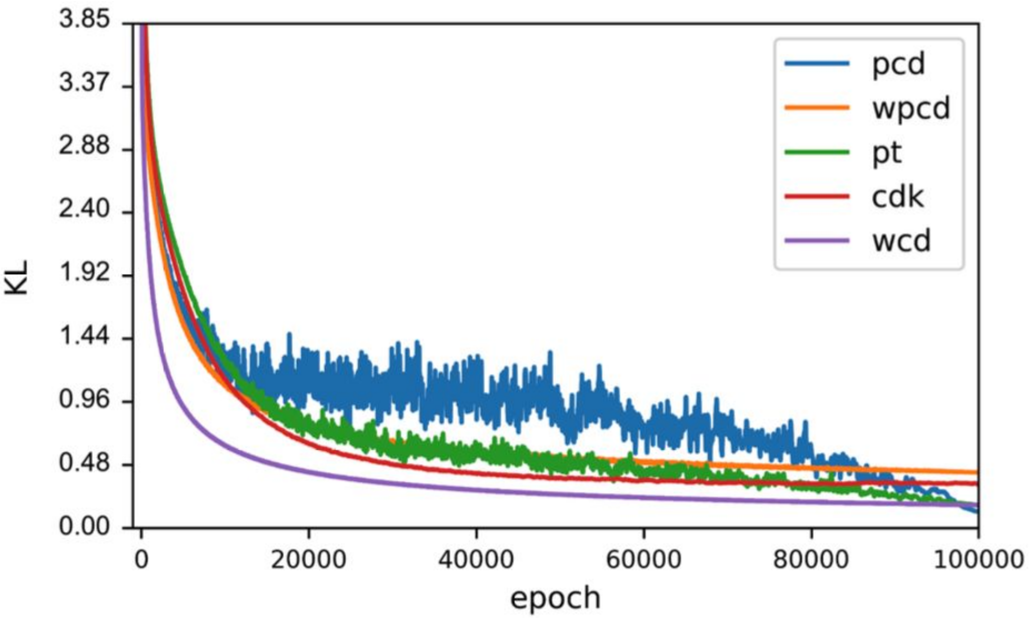
UNIVERSITAT ROVIRA i VIRGILI | UNIVERSITAT DE BARCELONA | UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

LS4-11

	KL	NLL	Prob-sum	Entropy-Percentage
CDK	0.2152(0.0205)	-4.0864(0.0205)	0.948(0.0024)	0.9216(0.0087)
PCD	0.1238(0.0332)	-3.995(0.0332)	0.9609(0.0028)	0.9511(0.0076)
WCD	0.0588(0.0004)	-3.93(0.0004)	0.9436(0.0004)	0.9576(0.0003)
$WPCD$	0.2428(0.0037)	-4.114(0.0037)	0.7865(0.0019)	0.8348(0.0017)
PT	0.099(0.0053)	-3.9702(0.0053)	0.9314(0.0012)	0.9418(0.0018)
$CD10^*$	0.0501(0.0051)	-3.9213(0.0051)	0.9732(0.0026)	0.9743(0.0024)
$WCD10^*$	0.0492(0.0015)	-3.9204(0.0015)	0.9541(0.0015)	0.9651(0.0011)



LS5-13



	KL	NLL	Prob-sum	Entropy-Percentage
$CD1$	0.3257(0.0629)	-4.89(0.0629)	0.926(0.0052)	0.8918(0.0134)
PCD	0.1353(0.0165)	-4.6996(0.0165)	0.9135(0.0039)	0.9228(0.005)
WCD	0.0725(0.0001)	-4.6368(0.0001)	0.9307(0.0001)	0.9452(0.0001)
$WPCD$	0.2808(0.0032)	-4.8451(0.0032)	0.7566(0.0023)	0.8025(0.002)
PT	0.1039(0.0333)	-4.6682(0.0333)	0.9652(0.0018)	0.9577(0.0071)
$CD10^*$	0.0717(0.0059)	-4.636(0.0059)	0.9624(0.0024)	0.9636(0.0022)
$WCD10^*$	0.0583(0.003)	-4.6225(0.0005)	0.9488(0.0003)	0.95856(0.0002)

According to the experimental results,

-> Generally, **WCD** performs the best

-> PCD is really **sensitive** with the learning rate

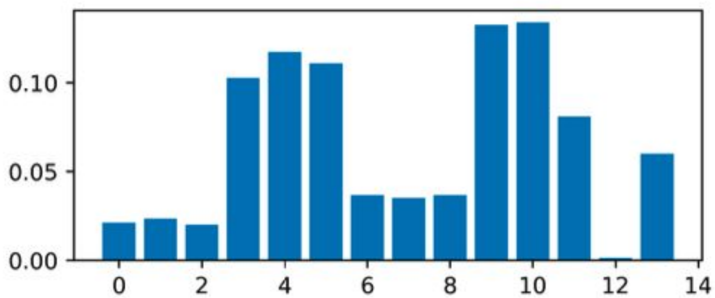
-> **Nh increasing** to around 10, CD10 begins to perform better than WCD1

-> **Larger** datasets lead to WCD10 outperforms CD10 and WCD1

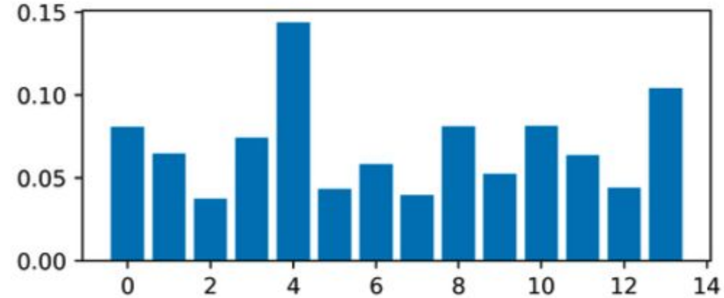
-> Persistent chain **suppresses** WCD

UNIVERSITAT ROVIRA i VIRGILI | UNIVERSITAT DE BARCELONA | UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

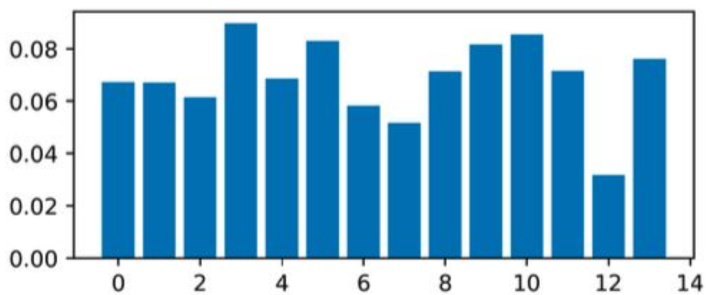
BS3x3



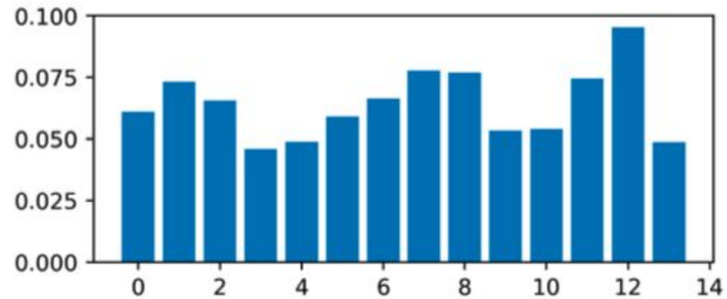
CD1



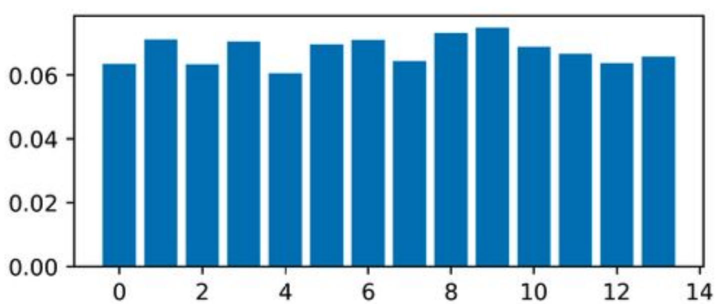
PCD



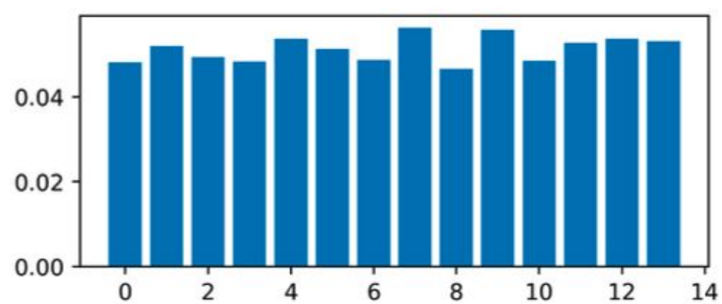
CD10



PT



WCD1



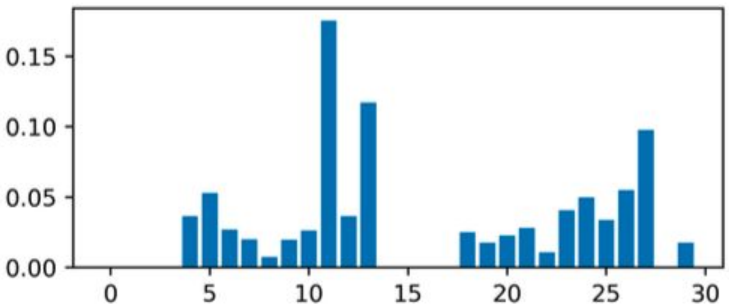
WPCD

UNIVERSITAT ROVIRA i VIRGILI

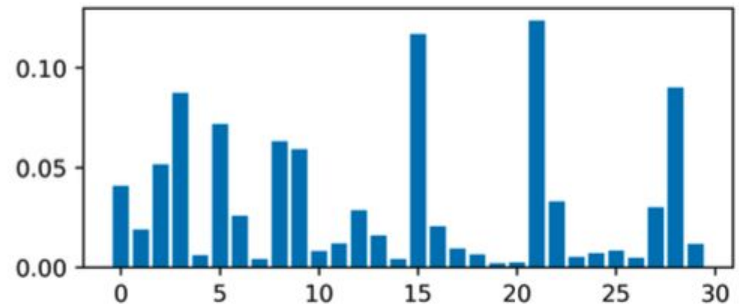
UNIVERSITAT DE BARCELONA

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

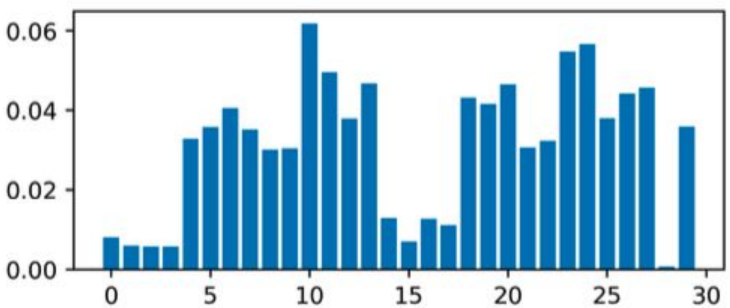
BS4x4



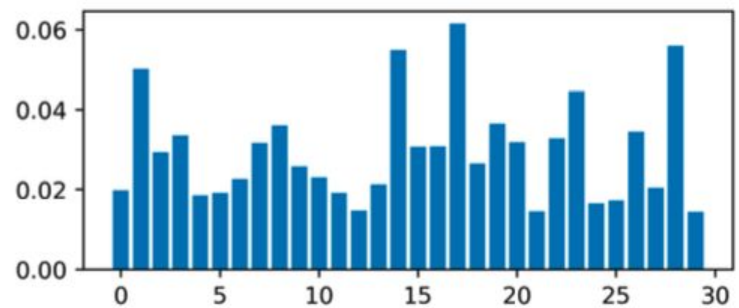
CD1



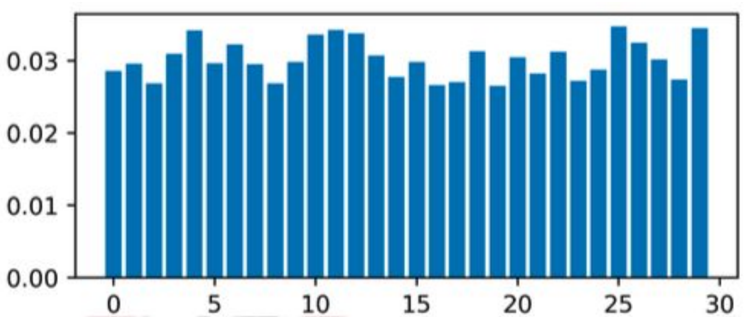
PCD



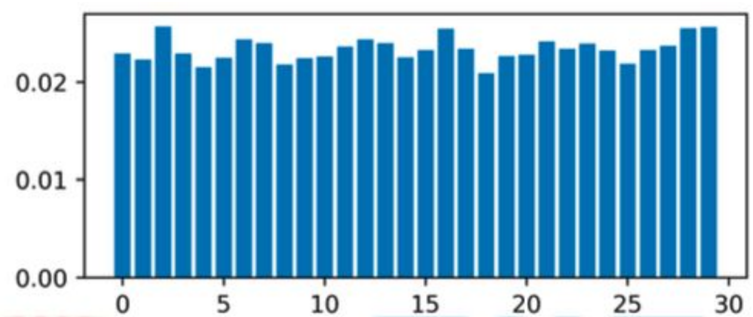
CD10



PT

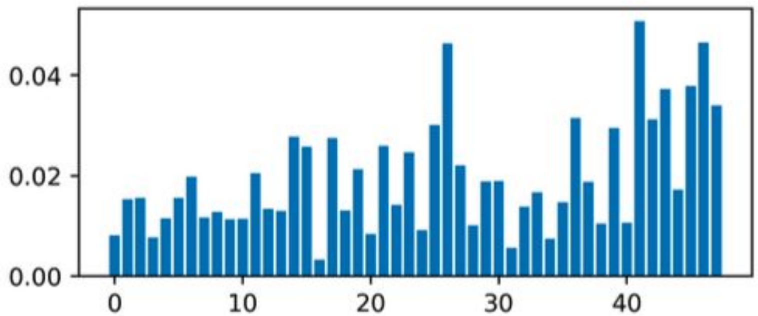


WCD1

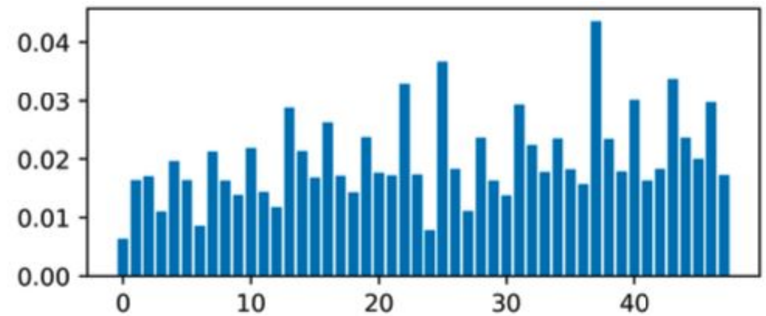


WPCD

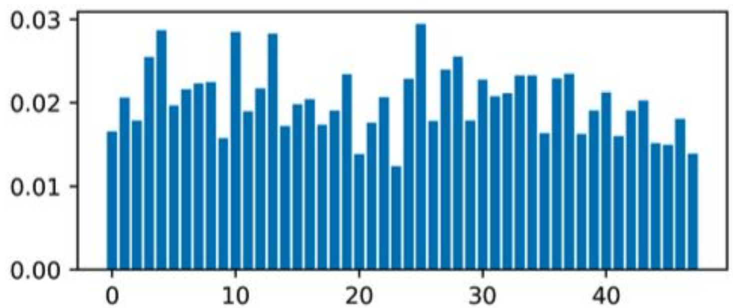
LS4-11



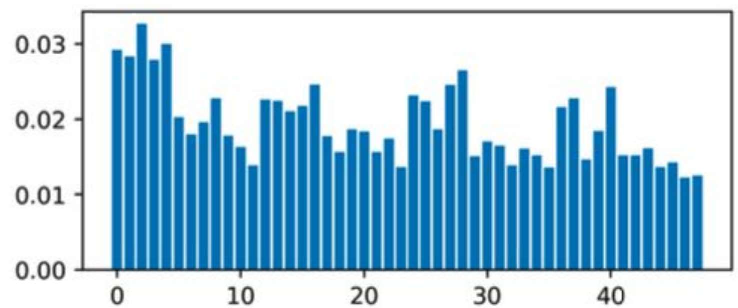
CD1



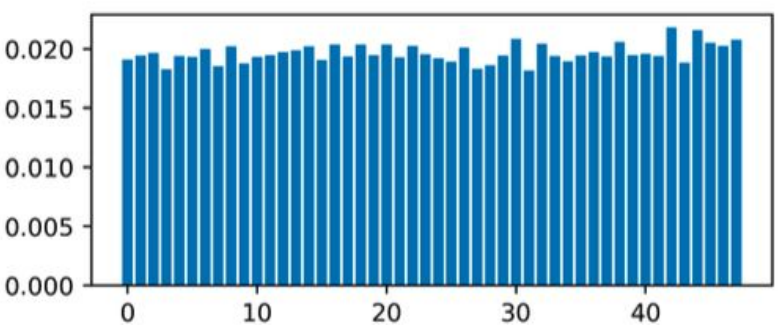
PCD



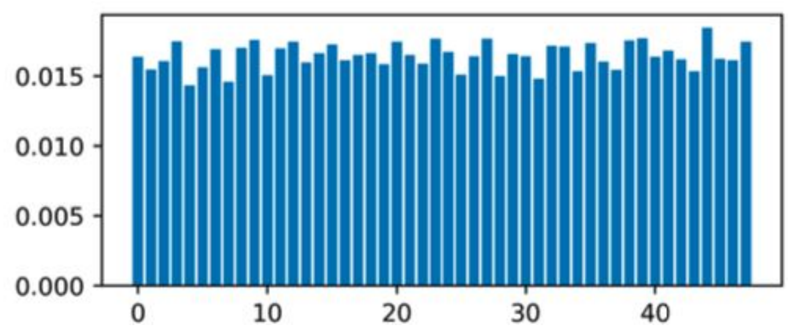
CD10



PT

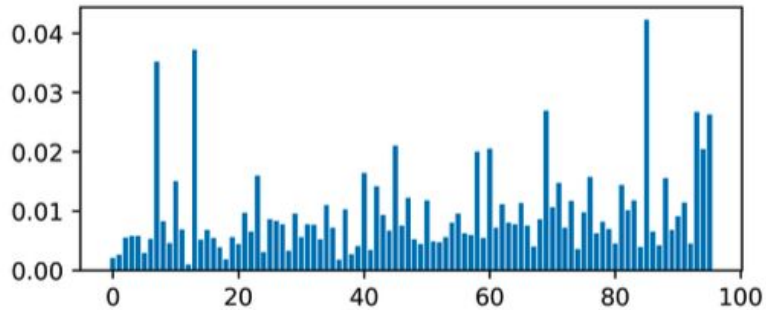


WCD1

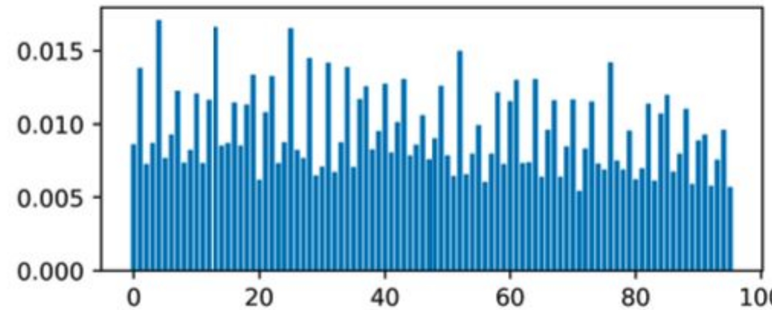


WPCD

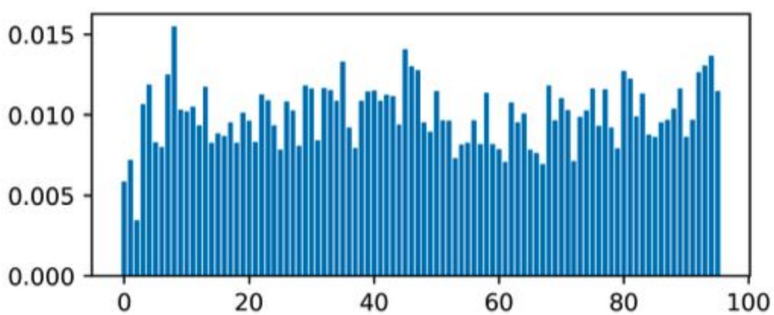
LS5-13



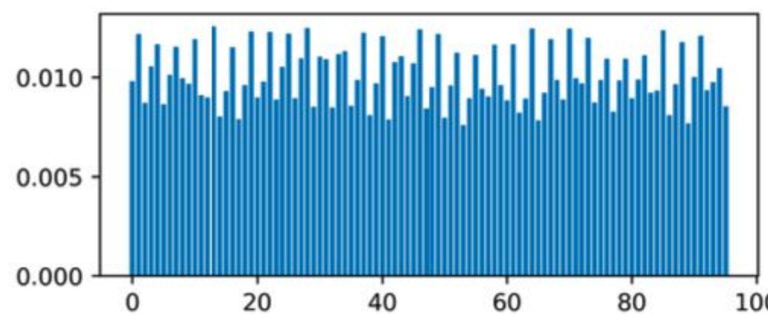
CD1



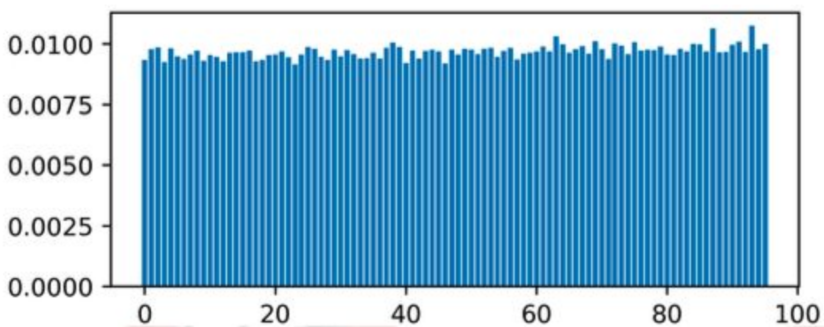
PCD



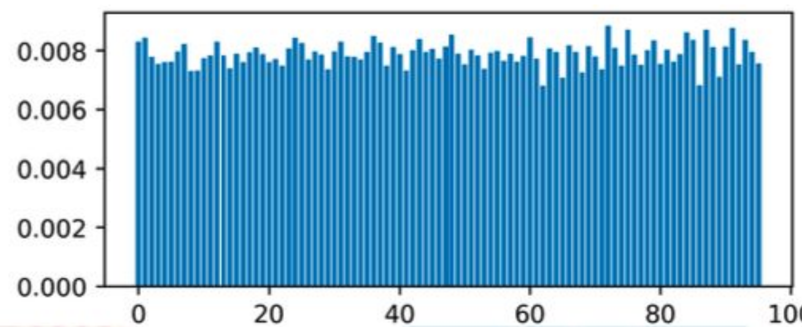
CD10



PT



WCD1



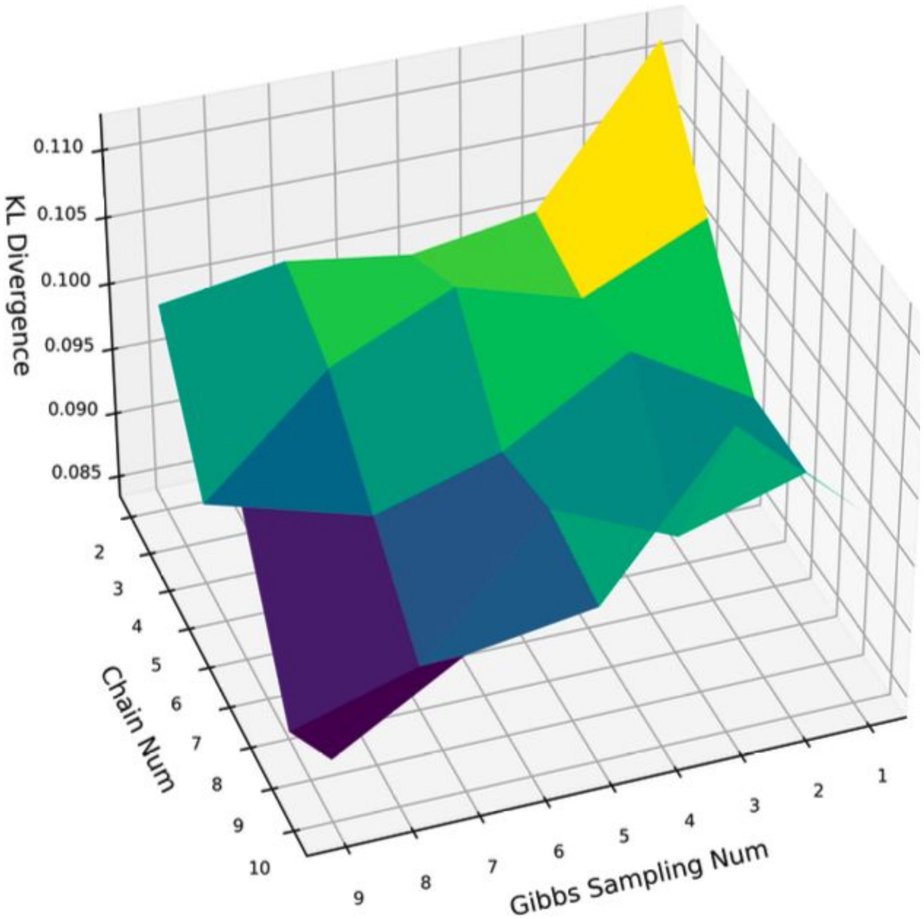
WPCD

According to the experimental results,

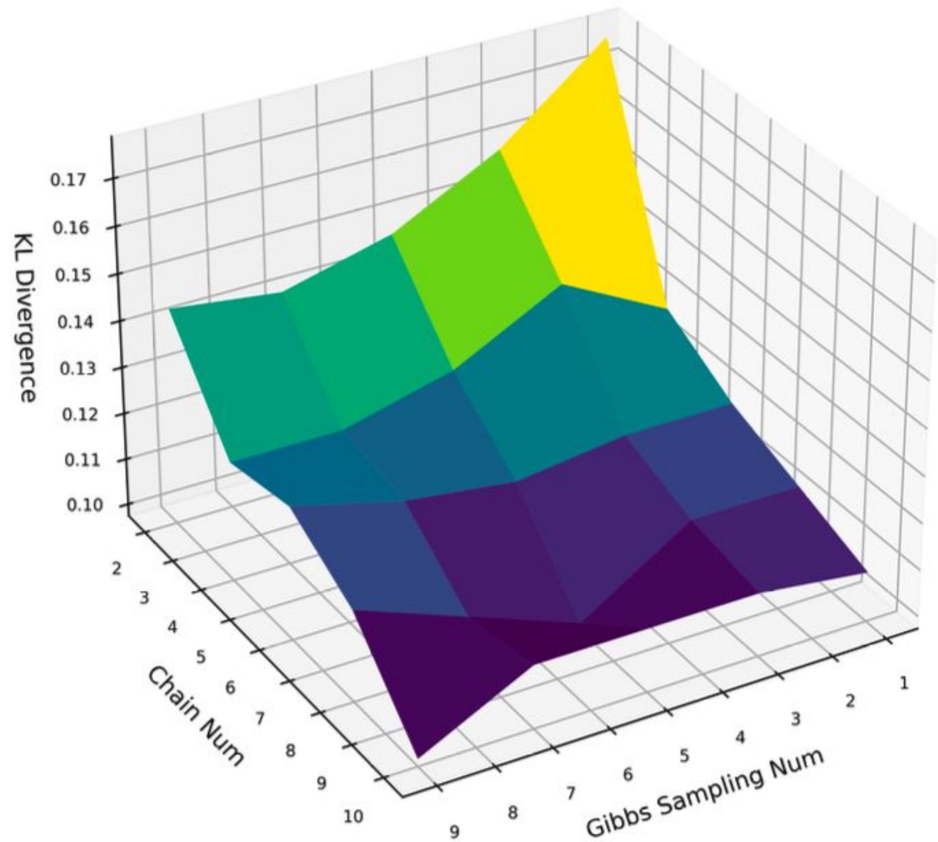
-> WCD produces the least bias.

UNIVERSITAT ROVIRA i VIRGILI · UNIVERSITAT DE BARCELONA · UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Parallel Tempering Parameter Exploration -> Chain Num & Gibbs sampling Num

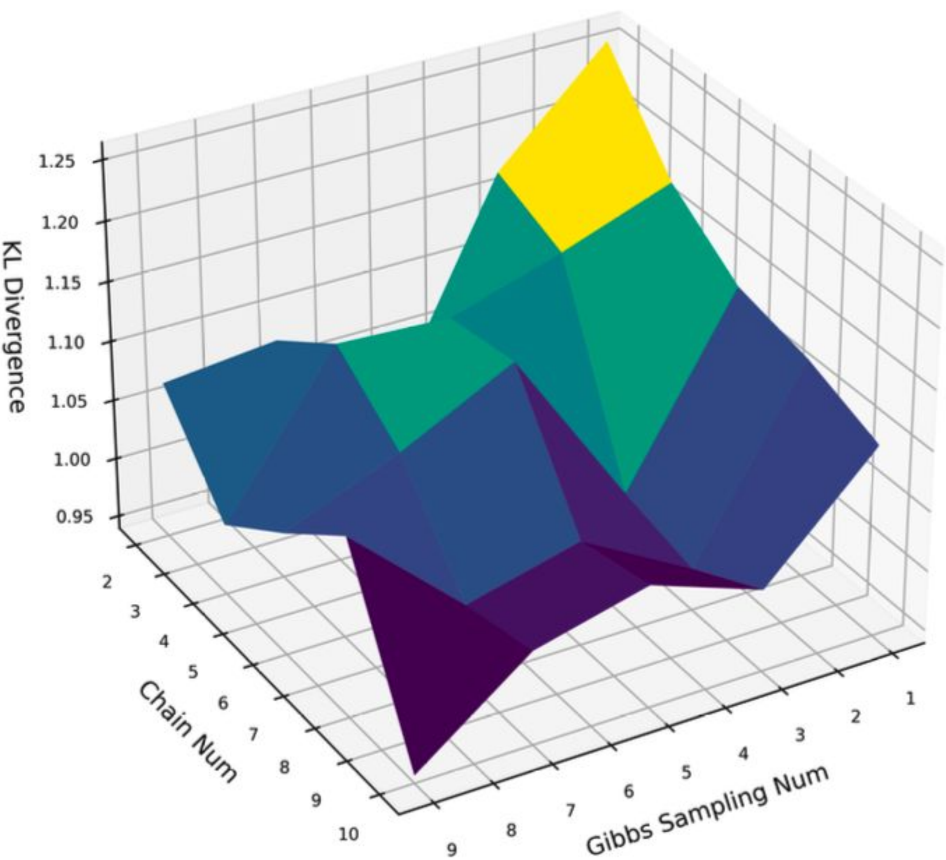


BS3x3

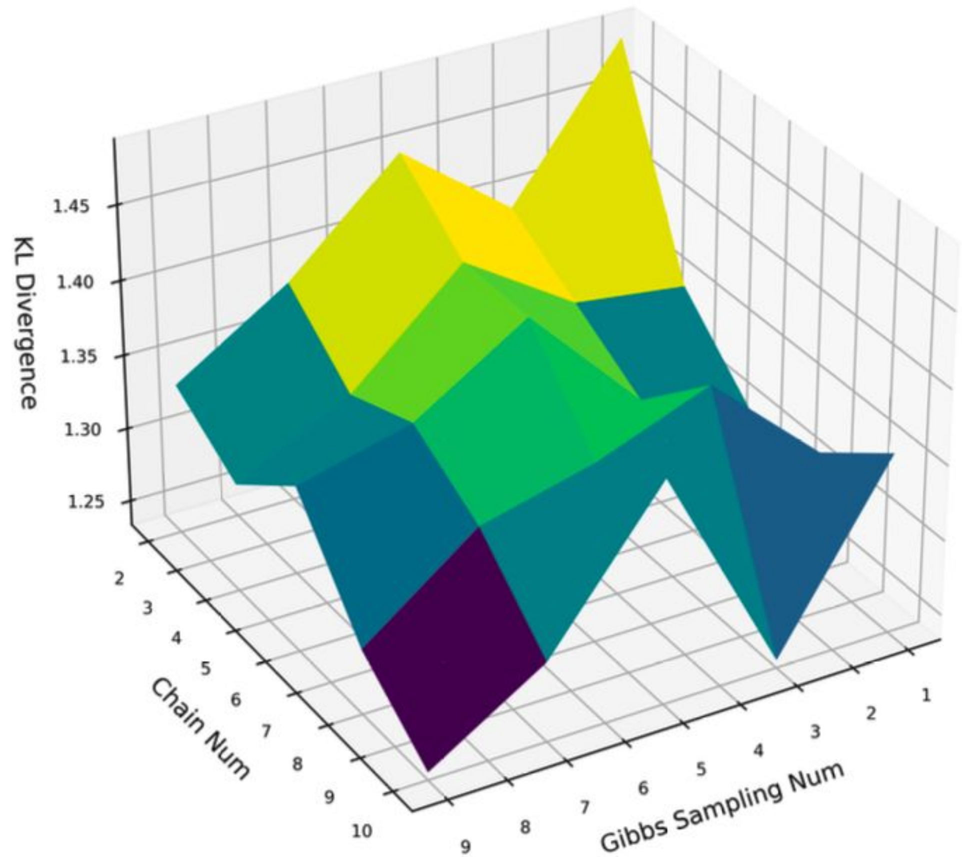


BS4x4

Parallel Tempering Parameter Exploration -> Chain Num & Gibbs sampling Num



LS4-11



LS5-13

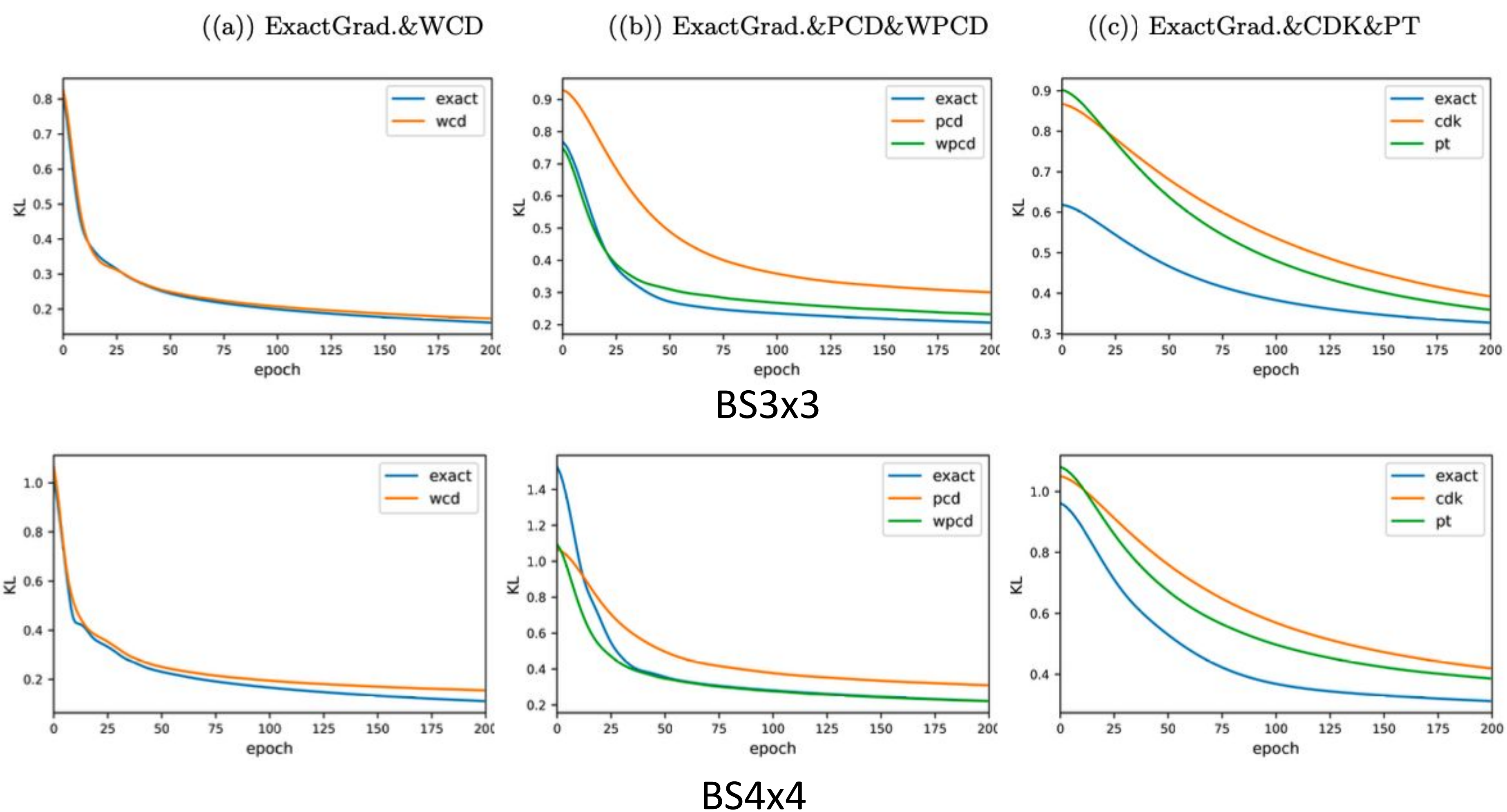
According to the experimental results,

-> With Nv increasing, PT improves more stably

-> Adding **Chain Num** and **Gibbs sampling Num** is **compatible**

-> Adding **Chain Num** is more **effective** than Adding **Gibbs sampling Num**

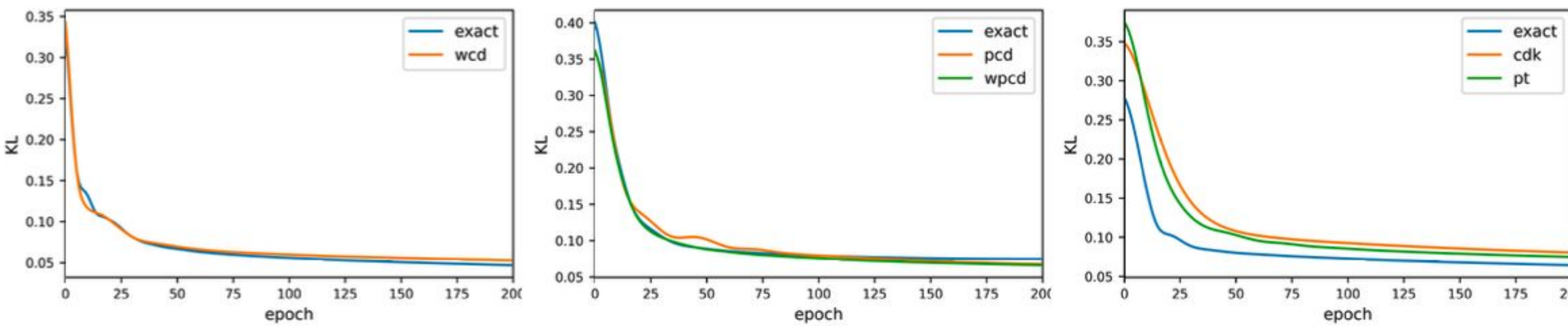
Analysis -> KL: ExactGrad. & ApproximationGrad.



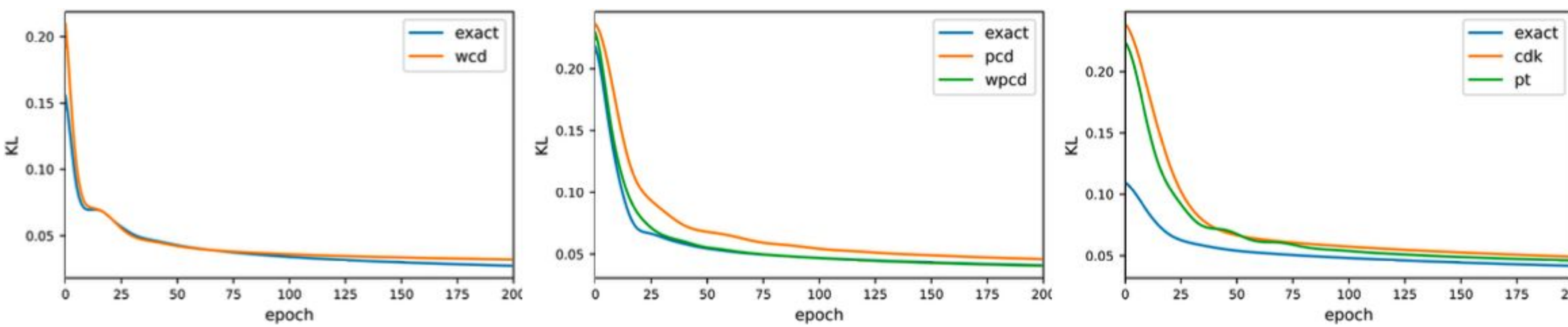
UNIVERSITAT ROVIRA i VIRGILI | UNIVERSITAT DE BARCELONA | UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Analysis -> KL: ExactGrad. & ApproximationGrad.

((a)) ExactGrad.&WCD ((b)) ExactGrad.&PCD&WPCD ((c)) ExactGrad.&CDK&PT



LS4-11



LS5-13

Analysis -> cos_sim: ExactGrad. & ApproximationGrad.

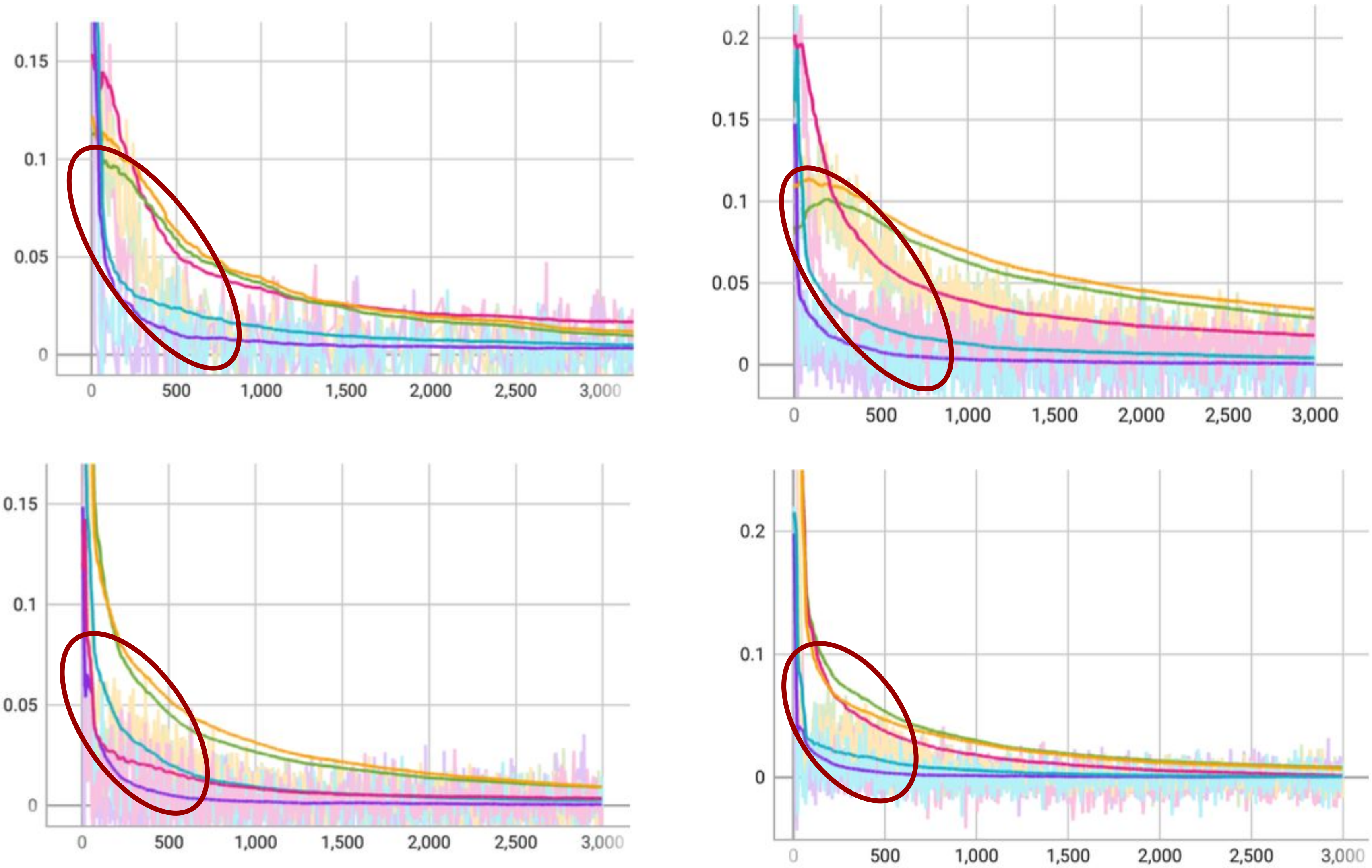
$$sim(A,B) = \frac{<A,B> + \epsilon}{\|A\| * \|B\| + \epsilon}$$

ϵ=1e-12

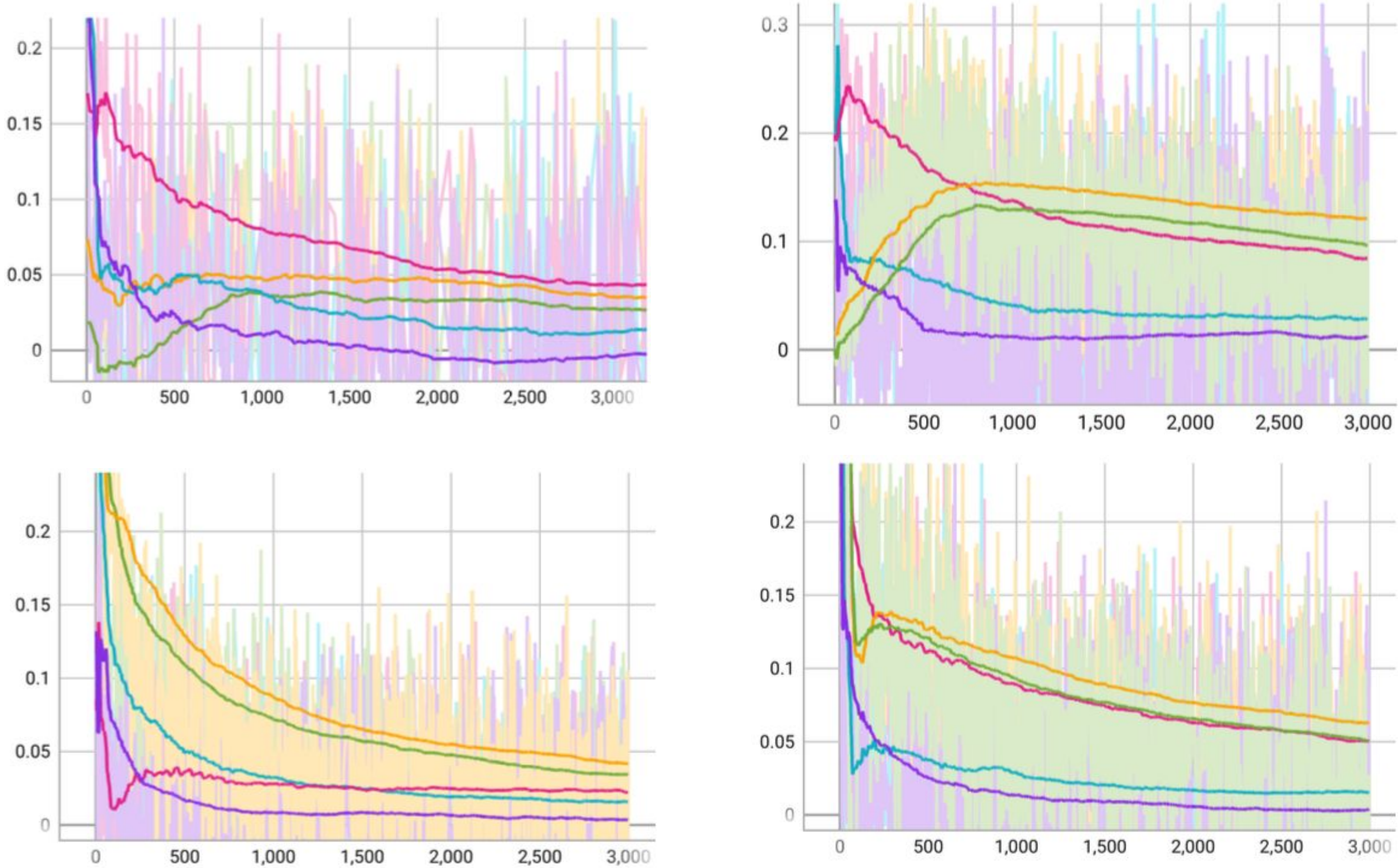
- dvb_cdk
- dvb_pcd
- dvb_pt
- dvb_wcd
- dvb_wpcd

BS3x3	BS4x4
LS4-11	LS5-13

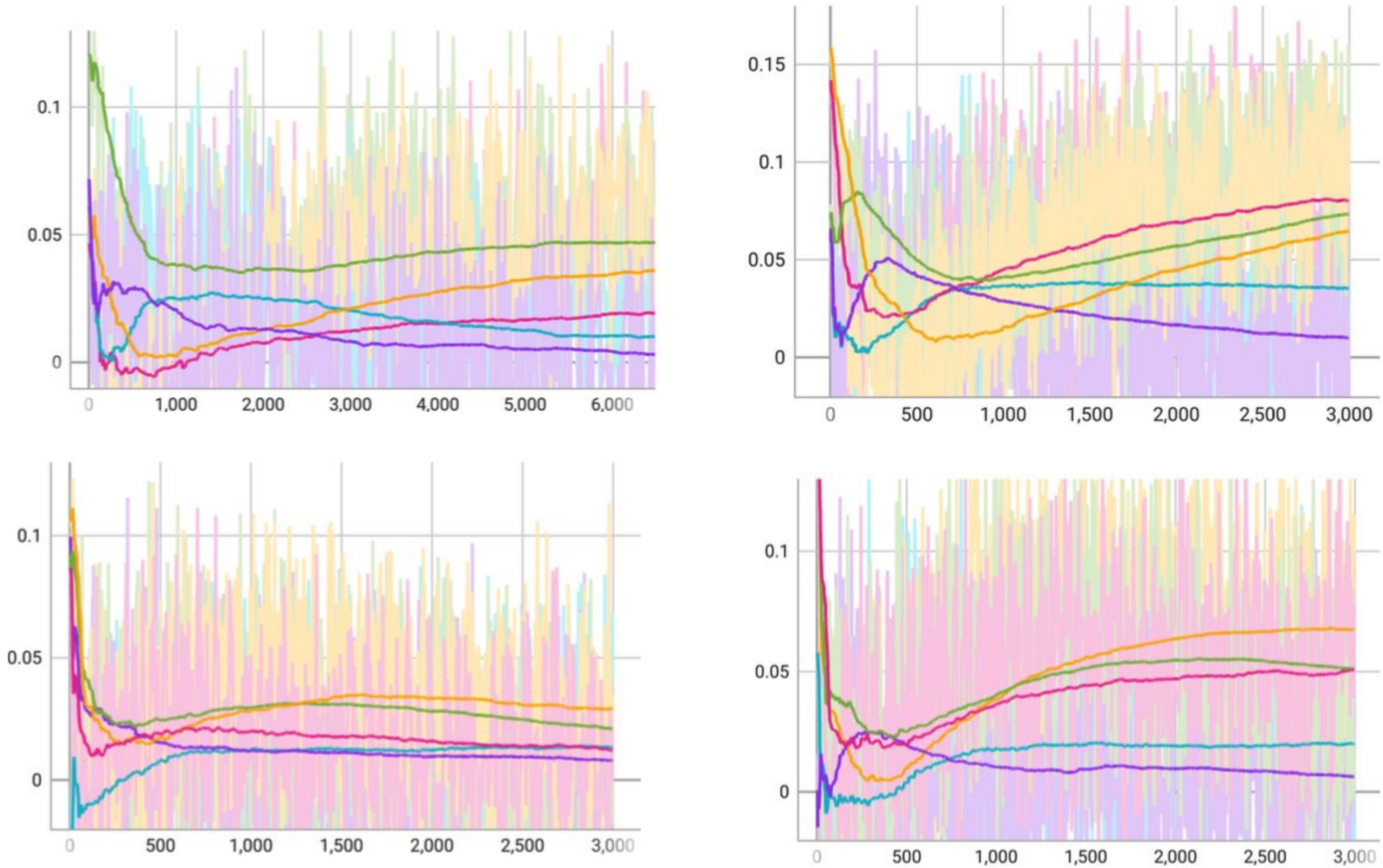
Analysis -> cos_sim: ExactGrad. & ApproximationGrad.



Analysis -> cos_sim: ExactGrad. & ApproximationGrad.



Analysis -> cos_sim: ExactGrad. & ApproximationGrad.



UNIVERSITAT ROVIRA i VIRGILI · UNIVERSITAT DE BARCELONA · UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

According to the experimental results,

-> All approximation cosine similarities converge to 0, always shift

-> WCD,WPCD are heterogeneous with CDK,PCD,PT

5. Conclusions

RBMs with **[0,1]** input perform more **stably**.

Increasing **Gibbs Sampling num**, **Chain num** and adding a **persistent chain enhance** model **performance** and **mitigate** the effect of **bias**.

WCD is **not compatible** with the **persistent chain**.

WCD-series algorithms and **CD**-series algorithms are **heterogeneous**.

Generally, considering **model performance** and **bias effect**, **WCD** is the **prior** choice.

Reference

[1]Paul Smolensky. Information processing in dynamical systems: Foundations of harmony theory. techreport. Colorado Univ at Boulder Dept of Computer Science, 1986.
[2]Geoffrey E Hinton. “Training products of experts by minimizing contrastive divergence”. inNeural computation: 14.8 (2002), pages 1771–1800.
[3]Yoshua Bengio and Olivier Delalleau. “Justifying and generalizing contrastive divergence”. inNeural computation: 21.6 (2009), pages 1601–1621.
[4]Asja Fischer and Christian Igel. “Bounding the bias of contrastive divergence learning”. inNeural computation: 23.3 (2011), pages 664–673.
[5]Tijmen Tieleman. “Training restricted Boltzmann machines using approximations to the likeli- hood gradient”. inProceedings of the 25th international conference on Machine learning: 2008, pages 1064–1071.