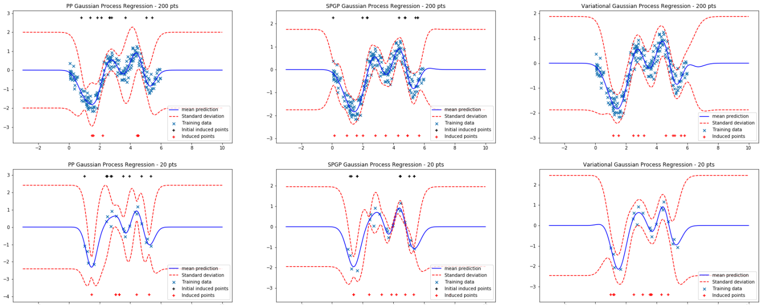
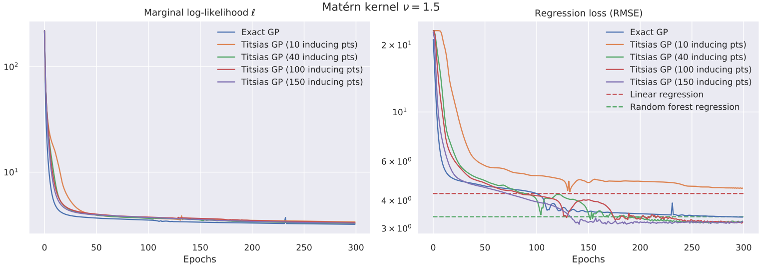

PP Gaussian Process Regression - 200 pts
SPGP Gaussian Process Regression - 200 pts
Variational Gaussian Process Regression - 200 pts
PP Gaussian Process Regression - 20 pts
SPGP Gaussian Process Regression - 20 pts
Variational Gaussian Process Regression - 20 pts
mean prediction
Standard deviation
Training data
Initial induced points
Induced points


Matérn kernel ν = 1.5
Marginal log-likelihood ℓ
Regression loss (RMSE)
Exact GP
Titsias GP (10 inducing pts)
Titsias GP (40 inducing pts)
Titsias GP (100 inducing pts)
Titsias GP (150 inducing pts)
Linear regression
Random forest regression
Epochs
10^2
10^1
2 × 10^1
6 × 10^0
4 × 10^0
3 × 10^0
0
50
100
150
200
250
300

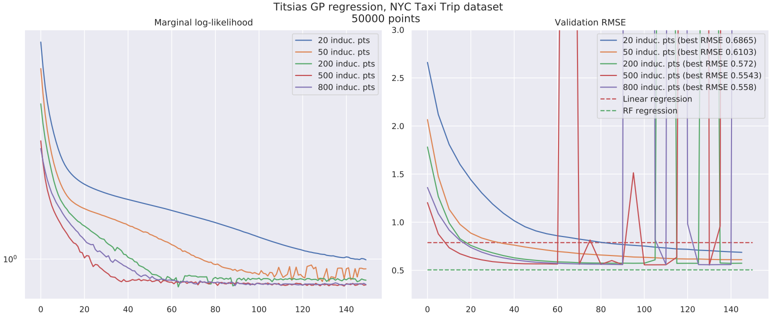


Figure: Comparison on the NYC Taxi dataset.

Deep Neural Network as Gaussian Processes

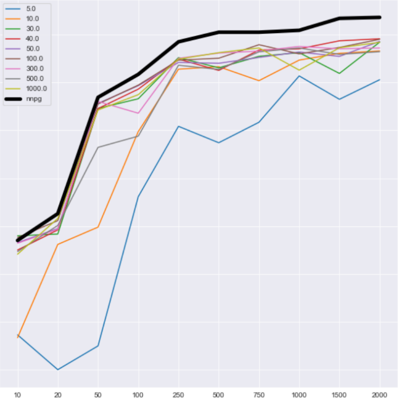
Review of Lee et al. article

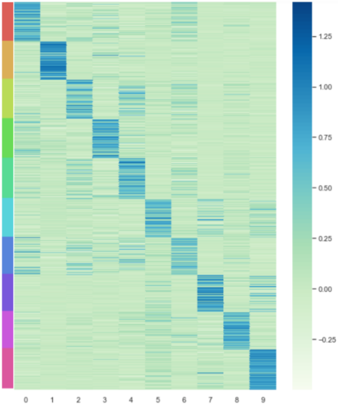
Delanoue Pierre, Guillo Clement

ENS Paris Saclay, MVA Master

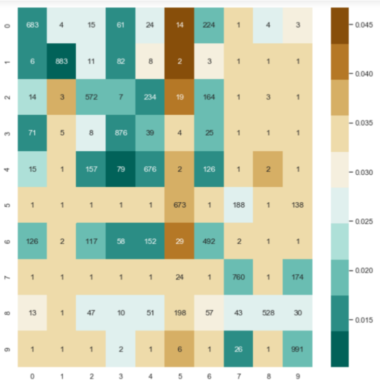
Mars 13, 2020

Accuracy and Posterior Mean

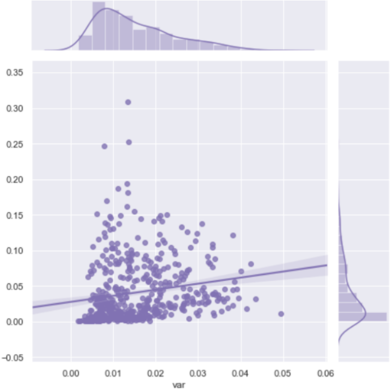




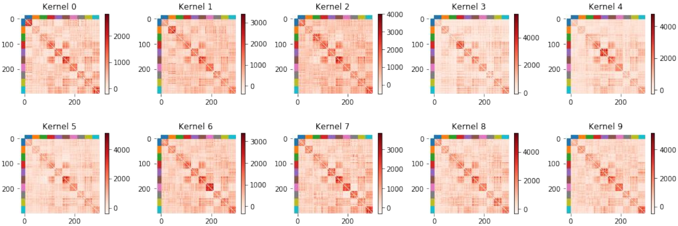
Posterior Variance



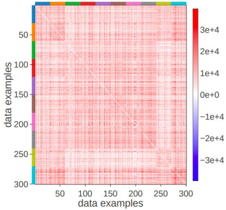
	0	1	2	3	4	5	6	7	8	9
0	683	4	15	61	24	14	224	1	4	3
1	6	883	11	82	8	2	3	1	1	1
2	14	3	572	7	234	19	164	1	3	1
3	71	5	8	876	39	4	25	1	1	1
4	15	1	157	79	676	2	126	1	2	1
5	1	1	1	1	1	673	1	188	1	138
6	126	2	117	58	152	29	492	2	1	1
7	1	1	1	1	1	24	1	760	1	174
8	13	1	47	10	51	198	57	43	528	30
9	1	1	1	2	1	6	1	26	1	991



Approximate Inference Turns Deep Networks Into Gaussian Processes

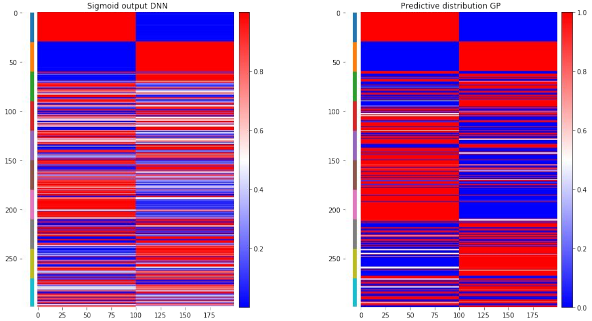


Class-specific kernels trained on MNIST.

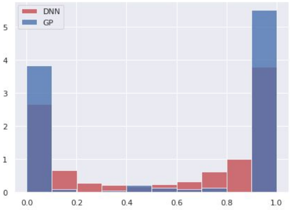


Kernel trained on CIFAR-10

DNN vs. GP in the face of overconfidence



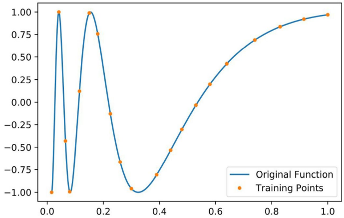
Predictions of DNN vs GP, trained on binary MNIST, on out-of-class samples



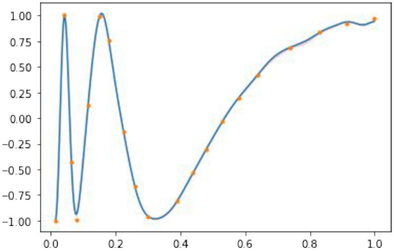
Distribution of predictions: DNN vs. GP, on out-of-class samples

Supervised learning

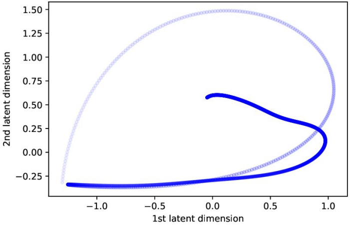
Original function



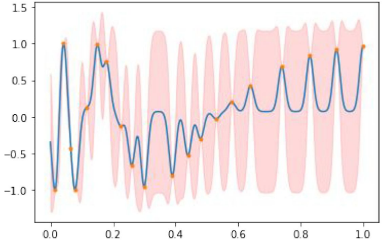
DGP



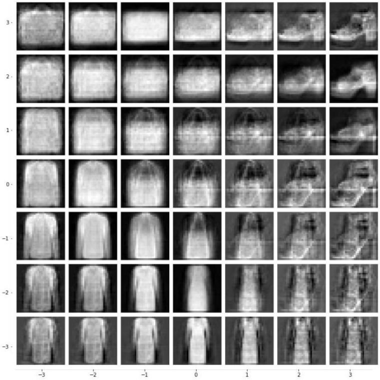
Latent space

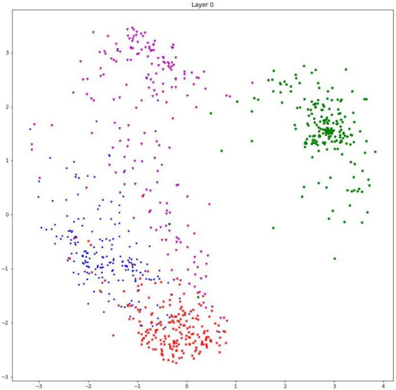


GP

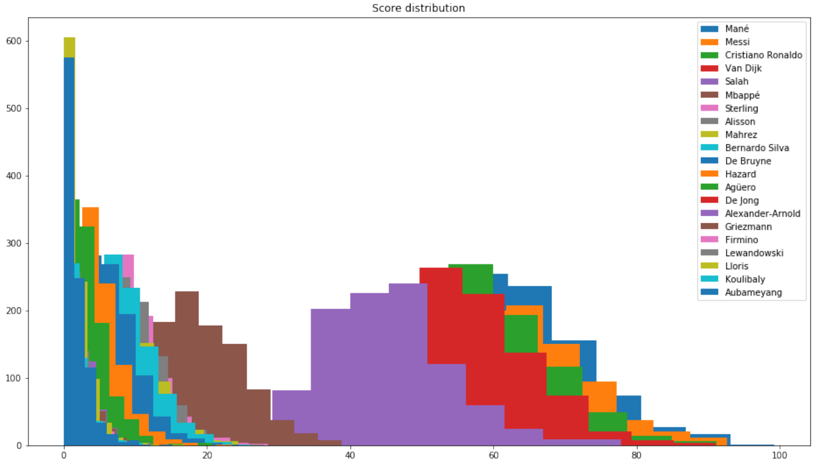


Unsupervised learning





Player score distribution



Film time-dependant scores

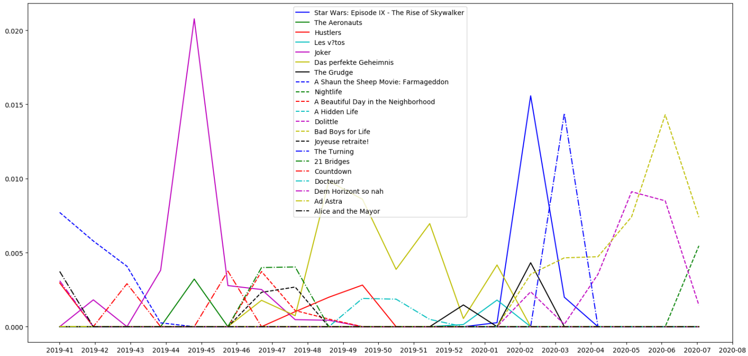


Figure: Evolution of the weights $w_{t,k}$

KPLS: from non-Bayesian to Bayesian

Thibault Lahire and Samuel Gruffaz

Presentation

Friday, 13 March 2020

KPLS (Kriging using Partial Least Squares)

Reference article:

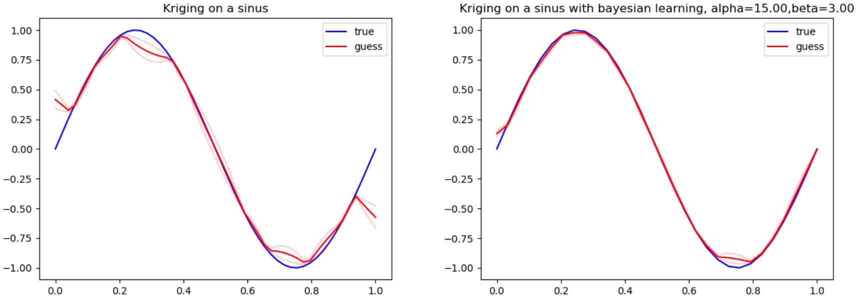
- Title: Improving Kriging surrogates of high-dimensional design models by Partial Least Squares dimension reduction
- Authors: Mohamed Amine Bouhlel, Nathalie Bartoli, Abdelkader Otsmane, Joseph Morlier
- Date: 2015

⟶ Main contribution = use of PLS to reduce the number of hyper-parameters
⟶ Underlying theory is fully **frequentist**

Experiments

Introducing $\Delta = (\mathbf{y} - \mathbf{F}\boldsymbol{\beta})^\top \mathbf{R}^{-1} (\mathbf{y} - \mathbf{F}\boldsymbol{\beta})$, we have:

$$\hat{\sigma}^2_{MP} = \frac{2\delta + \Delta}{2(\alpha - 1) + n} \quad \text{and} \quad \hat{\sigma}^2_{MAP} = \frac{2\delta + \Delta}{2(\alpha + 1) + n}$$



Kriging on a sinus

Kriging on a sinus with bayesian learning, alpha=15.00,beta=3.00

Figure: Example where $n = 13$ points

Data and Training

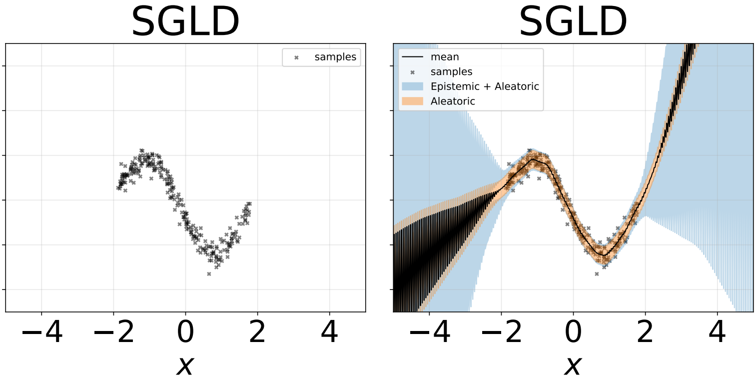


Figure: On the right: The data to be learned / On the left: The learned distribution of the data

Visualizing ϵ_c

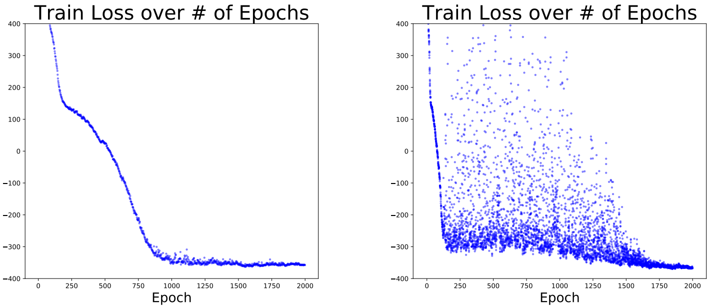


Figure: Comparing the training with a singleton batch or five batches

Latent Dirichlet Allocation graphical model

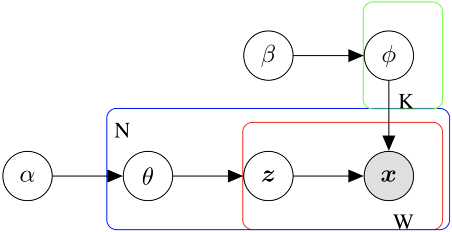
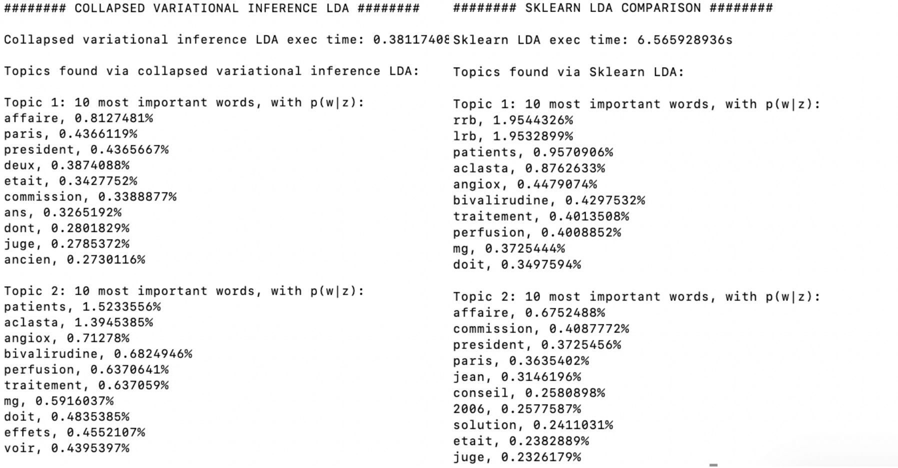


Figure: LDA graphical model

$$\begin{aligned} \theta_j &\sim \text{Dirichlet}(\alpha) \\ \theta_j &\sim \text{Dirichlet}(\alpha) \\ z_{ij} &\sim \text{Multinomial}(\theta_j) \\ x_{ij} &\sim \text{Multinomial}(\phi_{z_{ij}}) \end{aligned}$$

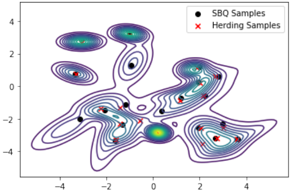
$$\begin{aligned} 1 \leq j \leq N \\ 1 \leq k \leq K \\ 1 \leq i \leq W \end{aligned}$$

Experiments: CVB-LDA script on Sequoia dataset

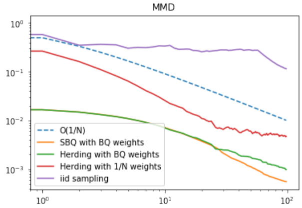


```
######## COLLAPSED VARIATIONAL INFERENCE LDA ########

Collapsed variational inference LDA exec time: 0.38117408

Topics found via collapsed variational inference LDA:

Topic 1: 10 most important words, with p(w|z):
affaire, 0.8127481%
paris, 0.4366119%
president, 0.4365667%
deux, 0.3874088%
etait, 0.3427752%
commission, 0.3388877%
ans, 0.3265192%
dont, 0.2801829%
juge, 0.2785372%
ancien, 0.2730116%

Topic 2: 10 most important words, with p(w|z):
patients, 1.5233556%
aclasta, 1.3945385%
angiox, 0.71278%
bivalirudine, 0.6824946%
perfusion, 0.6370641%
traitement, 0.637059%
mg, 0.5916037%
doit, 0.4835385%
effets, 0.4552107%
voir, 0.4395397%
```

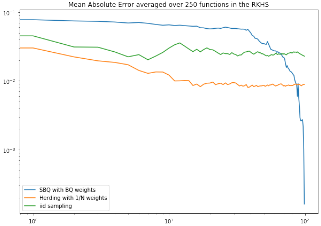
```
######## SKLEARN LDA COMPARISON ########

Sklearn LDA exec time: 6.565928936s

Topics found via Sklearn LDA:

Topic 1: 10 most important words, with p(w|z):
rrb, 1.9544326%
lrb, 1.9532899%
patients, 0.9570906%
aclasta, 0.8762633%
angiox, 0.4479074%
bivalirudine, 0.4297532%
traitement, 0.4013508%
perfusion, 0.4008852%
mg, 0.3725444%
doit, 0.3497594%

Topic 2: 10 most important words, with p(w|z):
affaire, 0.6752488%
commission, 0.4087772%
president, 0.3725456%
paris, 0.3635402%
jean, 0.3146196%
conseil, 0.2580898%
2006, 0.2577587%
solution, 0.2411031%
etait, 0.2382889%
juge, 0.2326179%
```



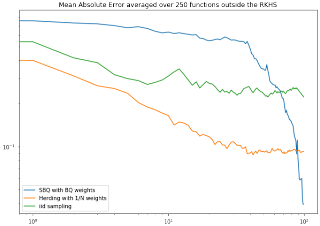
(a) 20 first samples of SBQ and Herding



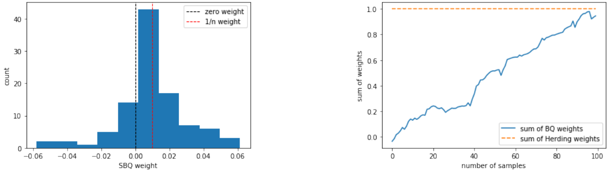
(b) MMD



(c) Mean Absolute Error for 250 random functions within the RKHS

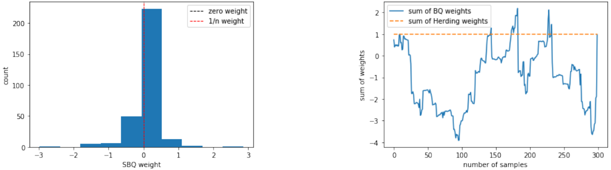


(d) Mean Absolute Error for 250 random functions outside of the RKHS



(e) Distribution of weights for 100 samples

(f) Sum of the weights for 100 samples



(g) Distribution of weights for 300 samples

(h) Sum of the weights for 300 samples

Plan of the Presentation

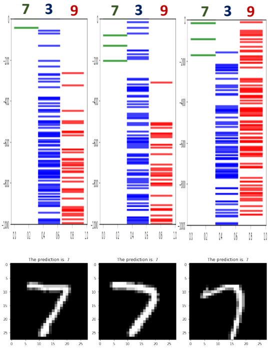
- Introduction
- Origin of the model
    - Stochastic Gradient Descent (SGD)
    - Stochastic Weight Averaging (SWA)
    - Assumptions
- Presentation of SWAG
    - Theoretical presentation of SWAG
    - SWAG Diagonal
    - SWAG Low Rank + Diagonal Structure
- Experimental Results
- Conclusion

A Simple Baseline for Bayesian Uncertainty

Algorithm 1 Continuous learning of SWAG model

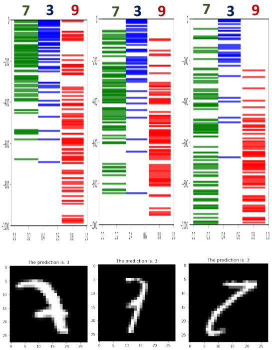
```
$\theta_{\text{pre}}$ : pretrained weights ; $\eta$ : learning rate ; $T$ : number of steps ; $c$ : moment update frequency ; $K$ : required rank ; $S$ : number of samples
  $\overline{\theta} \leftarrow \theta_0, \quad \overline{\theta^2} \leftarrow \theta_0^2$                                  {Initialize moments}
  for $i \leftarrow 1, 2, ..., T$ do
    $\theta_i \leftarrow \texttt{SGD}(\theta_{i-1})$                                        {Perform SGD update}
    if update_time = True then
      $n \leftarrow i/c$                                                  {Number of models}
      $\overline{\theta} \leftarrow \frac{n\overline{\theta} + \theta_i}{n+1}, \quad \overline{\theta^2} \leftarrow \frac{n\overline{\theta^2} + \theta_i^2}{n+1}$                 {Update moments}
      if nbr_stored_param = $K$ then
        forget_first_param
        store_new_param($\theta_i - \overline{\theta}$)                          {Store deviation}
    if estimate_time = True then
      for $i \leftarrow 1, 2, ..., S$ do
        Draw $\widetilde{\theta}_i \sim \mathcal{N}\left(\theta_{\text{SWA}}, \frac{1}{2}\Sigma_{\text{diag}} + \frac{\widehat{D}\widehat{D}^\top}{2(K-1)}\right)$
        $p(y^*|\text{Data}) \mathrel{+}= \frac{1}{S} p(y^*|\widetilde{\theta}_i)$
  return $p(y^*|\text{Data})$
```

Results obtained on the classification task



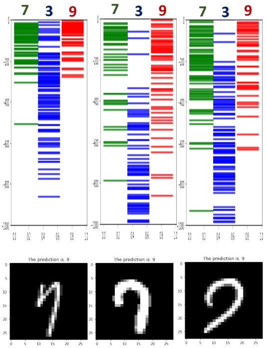
σ = 4.43 σ = 25.51 σ = 19.35

Prediction: 7
CORRECT



σ = 145.7 σ = 159.1 σ = 120.6

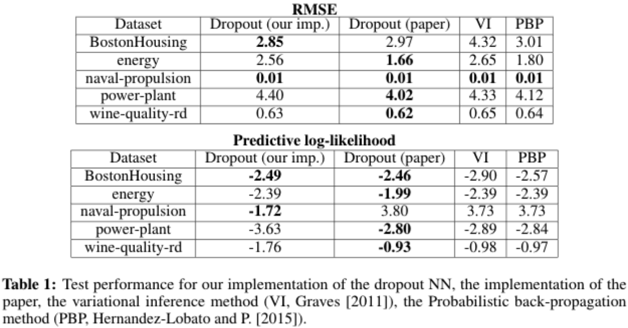
Prediction: 3
INCORRECT



σ = 67.4 σ = 203.1 σ = 153.4

Prediction: 9
INCORRECT

Results obtained on the regression task



RMSE

Dataset	Dropout (our imp.)	Dropout (paper)	VI	PBP
BostonHousing	**2.85**	2.97	4.32	3.01
energy	2.56	**1.66**	2.65	1.80
naval-propulsion	**0.01**	**0.01**	**0.01**	**0.01**
power-plant	4.40	**4.02**	4.33	4.12
wine-quality-rd	0.63	**0.62**	0.65	0.64

Predictive log-likelihood

Dataset	Dropout (our imp.)	Dropout (paper)	VI	PBP
BostonHousing	**-2.49**	**-2.46**	-2.90	-2.57
energy	-2.39	**-1.99**	-2.39	-2.39
naval-propulsion	**-1.72**	3.80	3.73	3.73
power-plant	-3.63	**-2.80**	-2.89	-2.84
wine-quality-rd	-1.76	**-0.93**	-0.98	-0.97

Table 1: Test performance for our implementation of the dropout NN, the implementation of the paper, the variational inference method (VI, Graves [2011]), the Probabilistic back-propagation method (PBP, Hernandez-Lobato and P. [2015]).