

# Bayesian computation project

Charles Dufour

EPFL

Spring semester 2019

# Table of contents

- 1 Framework
- 2 Data
- 3 Models
- 4 Methods used
  - Gradient descent and variant
  - Metropolis Hastings and variant
- 5 Comparison
  - Model accuracy
  - Method comparison
- 6 Conclusion


- 1 Framework
- 2 Data
- 3 Models
- 4 Methods used
- 5 Comparison
- 6 Conclusion

# Framework implementations and limits<sup>1</sup>

## Optimization

- gradient descent
- linear search gd
- Wolfe cond gd
- Newton gd (slow)
- Stochastic gd

---

<sup>1</sup>more information can be found in the  repository

# Framework implementations and limits<sup>1</sup>


## Optimization

- gradient descent
- linear search gd
- Wolfe cond gd
- Newton gd (slow)
- Stochastic gd

## Approximation

- Laplace
- GVA (need to reparametrize)

---

<sup>1</sup>more information can be found in the  repository

# Framework implementations and limits<sup>1</sup>

## Optimization

- gradient descent
- linear search gd
- Wolfe cond gd
- Newton gd (slow)
- Stochastic gd


## Approximation

- Laplace
- GVA (need to reparametrize)

## Sampling

- MH random walk
- MALA
- IS, RS
- Gibbs
- ~~MH within Gibbs~~

---

<sup>1</sup>more information can be found in the  repository

- 1 Framework
- 2 Data**
- 3 Models
- 4 Methods used
- 5 Comparison
- 6 Conclusion

# Structure

Figure: Hourly wage and features in the USA, May 1985

ED	SOUTH	NONWH	HISP	FE	MARR	MARRFE	EX	EXSQ	UNION	LNWAGE	AGE	MANUF	CONSTR	MANAG	SALES	CLER	SERV	PROF
10	0	0	0	0	1	0	27	729	0	2.1972	43	0	1	0	0	0	0	0
12	0	0	0	0	1	0	20	400	0	1.7047	38	0	0	0	1	0	0	0
12	0	0	0	1	0	0	4	16	0	1.3350	22	0	0	0	1	0	0	0
12	0	0	0	1	1	1	29	841	0	2.3514	47	0	0	0	0	1	0	0
12	0	0	0	0	1	0	40	1600	1	2.7080	58	0	1	0	0	0	0	0



# Structure

Figure: Hourly wage and features in the USA, May 1985

ED	SOUTH	NONWH	HISP	FE	MARR	MARRFE	EX	EXSQ	UNION	LNWAGE	AGE	MANUF	CONSTR	MANAG	SALES	CLER	SERV	PROF
10	0	0	0	0	1	0	27	729	0	2.1972	43	0	1	0	0	0	0	0
12	0	0	0	0	1	0	20	400	0	1.7047	38	0	0	0	1	0	0	0
12	0	0	0	1	0	0	4	16	0	1.3350	22	0	0	0	1	0	0	0
12	0	0	0	1	1	1	29	841	0	2.3514	47	0	0	0	0	1	0	0
12	0	0	0	0	1	0	40	1600	1	2.7080	58	0	1	0	0	0	0	0

## Purpose

- predict exactly the revenue
- predict if revenue above mean

# Structure

Figure: Hourly wage and features in the USA, May 1985

ED	SOUTH	NONWH	HISP	FE	MARR	MARRFE	EX	EXSQ	UNION	LNWAGE	AGE	MANUF	CONSTR	MANAG	SALES	CLER	SERV	PROF
10	0	0	0	0	1	0	27	729	0	2.1972	43	0	1	0	0	0	0	0
12	0	0	0	0	1	0	20	400	0	1.7047	38	0	0	0	1	0	0	0
12	0	0	0	1	0	0	4	16	0	1.3350	22	0	0	0	1	0	0	0
12	0	0	0	1	1	1	29	841	0	2.3514	47	0	0	0	0	1	0	0
12	0	0	0	0	1	0	40	1600	1	2.7080	58	0	1	0	0	0	0	0

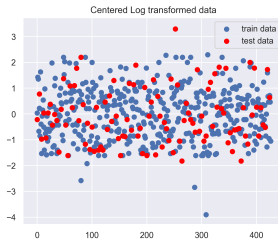
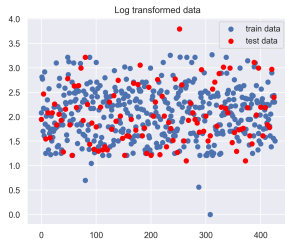
## Purpose

- predict exactly the revenue
- predict if revenue above mean

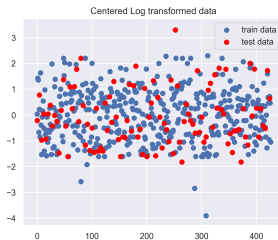
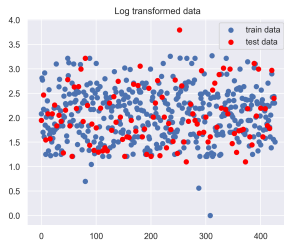
## Features dropped due to high correlation

- AGE
- EXSQ

# Visualization



# Visualization



- 1 Framework
- 2 Data
- 3 Models**
- 4 Methods used
- 5 Comparison
- 6 Conclusion

# Models

3 models implemented:

- Gaussian model

$$Y|\beta, \sigma \sim \mathcal{N}(X\beta, \sigma^2) \quad \beta \sim \mathcal{N}_d(\vec{0}, 3^2 I), \quad \sigma \sim \exp(2)$$

# Models

3 models implemented:

- Gaussian model

$$Y|\beta, \sigma \sim \mathcal{N}(X\beta, \sigma^2) \quad \beta \sim \mathcal{N}_d(\vec{0}, 3^2 I), \quad \sigma \sim \exp(2)$$

- Student model

$$Y|\beta, \nu \sim X\beta + t_\nu \quad \beta \sim \mathcal{N}_d(\vec{0}, 3^2 I), \quad \nu \sim \Gamma(2, 4)$$

# Models

3 models implemented:

- Gaussian model

$$Y|\beta, \sigma \sim \mathcal{N}(X\beta, \sigma^2) \quad \beta \sim \mathcal{N}_d(\vec{0}, 3^2 I), \quad \sigma \sim \exp(2)$$

- Student model

$$Y|\beta, \nu \sim X\beta + t_\nu \quad \beta \sim \mathcal{N}_d(\vec{0}, 3^2 I), \quad \nu \sim \Gamma(2, 4)$$

- Logistic regression

$$\mathbb{P}(Y = 1|X, \beta) = \frac{e^{X^T \beta}}{1 + e^{X^T \beta}}, \quad \beta \sim \mathcal{N}_d(0, 3^2)$$



- 1 Framework
- 2 Data
- 3 Models
- 4 Methods used**
  - Gradient descent and variant
  - Metropolis Hastings and variant
- 5 Comparison
- 6 Conclusion

# MAP (Maximum at posteriori) estimator

$$f(\theta) = -\log(\tilde{f}(\theta|d))$$

$$\theta_c = \theta - \eta * \nabla_{\theta} f(\theta)$$

# MAP (Maximum at posteriori) estimator

$$f(\theta) = -\log(\tilde{f}(\theta|d))$$

$$\theta_c = \theta - \eta * \nabla_{\theta} f(\theta)$$

- Vanilla gradient descent

# MAP (Maximum at posteriori) estimator

$$f(\theta) = -\log(\tilde{f}(\theta|d))$$

$$\theta_c = \theta - \eta * \nabla_{\theta} f(\theta)$$

- Vanilla gradient descent
- Line search gd ( $0 < \beta < 1$ )  
accept if :

$$f(\theta_c) < f(\theta) - \eta \varepsilon ||\nabla_{\theta} f(\theta)||_2^2$$

else:

$$\eta = \beta \eta$$

# MAP (Maximum at posteriori) estimator

$$f(\theta) = -\log(\tilde{f}(\theta|d))$$

$$\theta_c = \theta - \eta * \nabla_{\theta} f(\theta)$$

- Vanilla gradient descent
- Line search gd ( $0 < \beta < 1$ )  
accept if :

$$f(\theta_c) < f(\theta) - \eta \epsilon \|\nabla_{\theta} f(\theta)\|_2^2$$

else:

$$\eta = \beta \eta$$

- Wolfe condition ( $0 < c_1 < c_2 < 1$ )  
decrease  $\eta$  if

$$f(\theta_c) \geq f(\theta) + c_1 \eta \|\nabla_{\theta} f(\theta)\|_2^2$$

increase  $\eta$  if

$$-\nabla_{\theta} f(\theta_c)^T \nabla_{\theta} f(\theta) \geq -c_2 \|\nabla_{\theta} f(\theta)\|_2^2$$

# MH with random walk

## Theory

```
1: for  $i = 1$  to  $N$  do  
2:   draw  $\eta \sim \mathcal{N}_d(0, 1)$   
3:    $\theta_c = \theta_n + \varepsilon \eta$   
4:    $R = f(\theta_c|d)/f(\theta_n|d)$   
5:   if  $U(0, 1) \leq R$  then  
6:      $\theta_{n+1} = \theta_c$   
7:   else  
8:      $\theta_{n+1} = \theta_n$   
9:   end if  
10: end for
```

## Practice

- set  $\varepsilon$  such that the acceptance rate of the proposal is between 10 and 50 percent.
- compute everything using `expsumlog`
- check visually the chain to determine the burn-in
- test different initialization to detect potential silent failure

# MH with Langevin correction (MALA)

## Theory

As for the random walk MH algorithm except for

- the proposal:

$$\theta_c = \theta_n + \tau \nabla \log f(\theta_n | d) + \sqrt{2\tau} \eta$$

- the acceptance ratio:

$$R = \frac{f(\theta_c | d) q(\theta_n | \theta_c)}{f(\theta_n | d) q(\theta_c | \theta_n)}$$

## Practice

- as before but be more careful with tuning the step size  $\tau$



$$q(x, x') \propto \exp \left( \frac{\|x' - x - \tau \nabla \log f(x | d)\|_2^2}{-4\tau} \right)$$

- biggest challenge: implement computation of gradient in efficient manner

- 1 Framework
- 2 Data
- 3 Models
- 4 Methods used
- 5 Comparison**
  - Model accuracy
  - Method comparison
- 6 Conclusion



# Comparison of accuracy on test set

## Gaussian MAE

	error on test	error on train
gd	0.685706	0.613765
line_search_gd	0.685706	0.613765
Wolfe_cond_gd	0.738525	0.692682
MH_vanilla_mean	0.687246	0.614395
MH_vanilla_median	0.688279	0.614266
MH_Langevin_mean	0.687873	0.613893
MH_Langevin_median	0.688071	0.614006

# Comparison of accuracy on test set

## Gaussian MAE

	error on test	error on train
gd	0.685706	0.613765
line_search_gd	0.685706	0.613765
Wolfe_cond_gd	0.738525	0.692682
MH_vanilla_mean	0.687246	0.614395
MH_vanilla_median	0.688279	0.614266
MH_Langevin_mean	0.687873	0.613893
MH_Langevin_median	0.688071	0.614006

## Student MAE

	error on test	error on train
gd	0.685292	0.609907
line_search_gd	0.685159	0.611209
Wolfe_cond_gd	0.685284	0.609814
MH_vanilla_mean	0.687430	0.611630
MH_vanilla_median	0.685471	0.612177
MH_Langevin_mean	0.685110	0.611450
MH_Langevin_median	0.684638	0.611450

# Comparison of accuracy on test set

## Gaussian MAE

	error on test	error on train
gd	0.685706	0.613765
line_search_gd	0.685706	0.613765
Wolfe_cond_gd	0.738525	0.692682
MH_vanilla_mean	0.687246	0.614395
MH_vanilla_median	0.688279	0.614266
MH_Langevin_mean	0.687873	0.613893
MH_Langevin_median	0.688071	0.614006

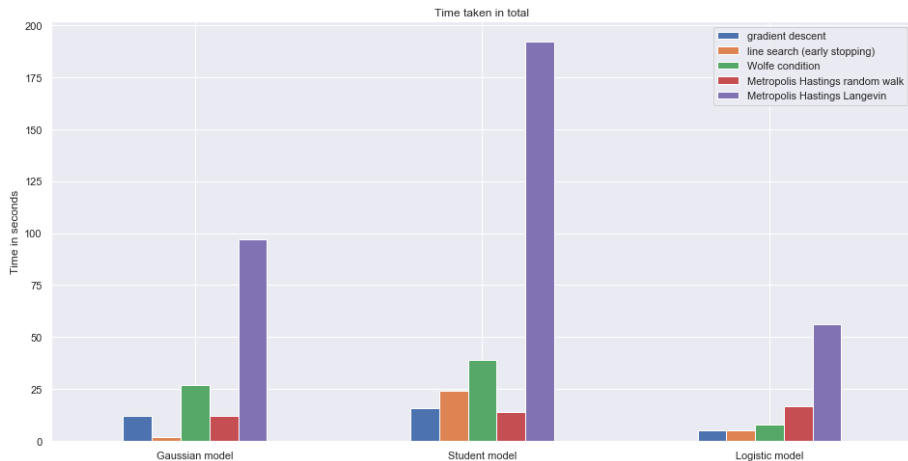
## Student MAE

	error on test	error on train
gd	0.685292	0.609907
line_search_gd	0.685159	0.611209
Wolfe_cond_gd	0.685284	0.609814
MH_vanilla_mean	0.687430	0.611630
MH_vanilla_median	0.685471	0.612177
MH_Langevin_mean	0.685110	0.611450
MH_Langevin_median	0.684638	0.611450

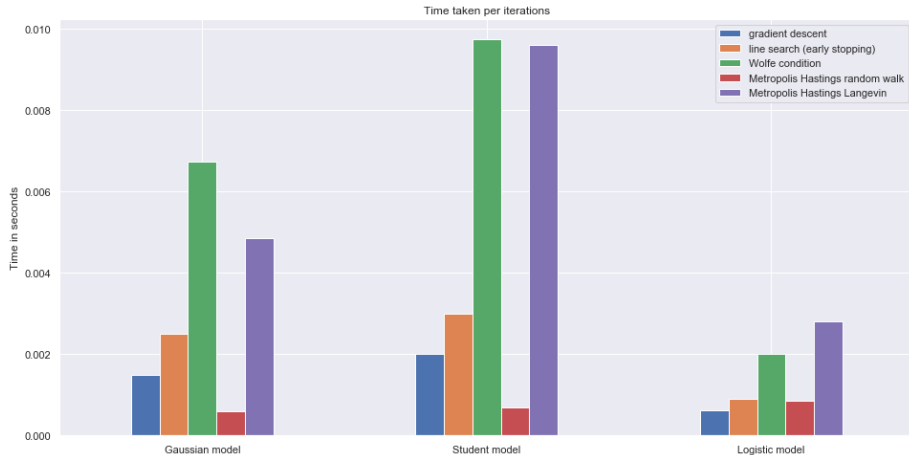
## Logistic error rate

	error on test	error on train
gd	0.448598	0.271663
line_search_gd	0.467290	0.269321
Wolfe_cond_gd	0.467290	0.269321
MH_vanilla_mean	0.467290	0.274005
MH_vanilla_median	0.467290	0.271663
MH_Langevin_mean	0.467290	0.271663
MH_Langevin_median	0.467290	0.274005

# Comparison in term of total time



# Comparison in term of time per iteration



- 1 Framework
- 2 Data
- 3 Models
- 4 Methods used
- 5 Comparison
- 6 Conclusion**

# Conclusion

## Modelization

- Simpler methods and models performed the best
- Relationship highly non-linear

# Conclusion

## Modelization

- Simpler methods and models performed the best
- Relationship highly non-linear

## Improvement

- Tuning of hyper-parameters
- Feature engineering



# Conclusion

## Modelization

- Simpler methods and models performed the best
- Relationship highly non-linear

## Improvement

- Tuning of hyper-parameters
- Feature engineering
- Gamma model
- Classification in multiple ordered classes

# Conclusion




## Modelization

- Simpler methods and models performed the best
- Relationship highly non-linear

## Improvement

- Tuning of hyper-parameters
- Feature engineering
- Gamma model
- Classification in multiple ordered classes
- More robust and faster module to use more advanced techniques

# References

-  Guillaume Dehaene  
*Lecture Notes, Bayesian computation MATH-435, 2019.*
-  E. R. Berndt  
*The practice of econometrics : classic and contemporary. Addison-Wesley Pub. Co., 1991*
-  GitHub repository  
[https://github.com/dufourc1/Bayesian computation](https://github.com/dufourc1/Bayesian_computation)