Bayesian Autoregressive Distributed Lag *via* Stochastic Gradient Hamiltonian Monte Carlo

Al-Ahmadgaid B. Asaad

alasaadstat@gmail.com

School of Statistics
University of the Philippines Diliman

April 2017

Background and Motivation

Simple Linear Regression Model

Consider the following model:

$$y_i = w_0 + w_1 x_i + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2) \tag{1}$$

- In classical statistics, w_0 and w_1 are assumed to be **fixed**.
- In Bayesian statistics, w_0 and w_1 are assumed to be **random variable and follows some distribution**.

Background and Motivation

Simple Linear Regression Model

Consider the following model:

$$y_i = w_0 + w_1 x_i + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2) \tag{1}$$

- In classical statistics, w_0 and w_1 are assumed to be **fixed**.
- In Bayesian statistics, w_0 and w_1 are assumed to be **random variable and follows some distribution**.

Background and Motivation

Simple Linear Regression Model

Consider the following model:

$$y_i = w_0 + w_1 x_i + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2) \tag{1}$$

- In classical statistics, w_0 and w_1 are assumed to be **fixed**.
- In Bayesian statistics, w_0 and w_1 are assumed to be **random variable and follows some distribution**.

Background and Motivation

Simple Linear Regression Model

Consider the following model:

$$y_i = w_0 + w_1 x_i + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2) \tag{2}$$

- In classical statistics, w_0 and w_1 are estimated using **Maximum Likelihood Estimation**.
- In Bayesian statistics, w_0 and w_1 are estimated using **Bayes' theorem**.

Background and Motivation

Simple Linear Regression Model

Consider the following model:

$$y_i = w_0 + w_1 x_i + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2) \tag{2}$$

- In classical statistics, w_0 and w_1 are estimated using **Maximum Likelihood Estimation**.
- In Bayesian statistics, w_0 and w_1 are estimated using **Bayes' theorem**.

Background and Motivation

Simple Linear Regression Model

Consider the following model:

$$y_i = w_0 + w_1 x_i + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2) \tag{2}$$

- In classical statistics, w_0 and w_1 are estimated using **Maximum Likelihood Estimation**.
- In Bayesian statistics, w_0 and w_1 are estimated using **Bayes' theorem**.

Background and Motivation

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{3}$$

- $\mathbb{P}(\mathbf{w}|\mathbf{y})$ is the **a posteriori**;
- $\mathbb{P}(\mathbf{w})$ is the **a priori**;
- $\mathbb{P}(\mathbf{y}|\mathbf{w})$ is the **likelihood**;
- $\mathbb{P}(\mathbf{y})$ is the **marginal likelihood** or **model evidence**;

Background and Motivation

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{3}$$

- $\mathbb{P}(\mathbf{w}|\mathbf{y})$ is the **a posteriori**;
- $\mathbb{P}(\mathbf{w})$ is the **a priori**;
- $\mathbb{P}(\mathbf{y}|\mathbf{w})$ is the **likelihood**;
- $\mathbb{P}(\mathbf{y})$ is the **marginal likelihood** or **model evidence**;

Background and Motivation

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{3}$$

- $\mathbb{P}(\mathbf{w}|\mathbf{y})$ is the **a posteriori**;
- $\mathbb{P}(\mathbf{w})$ is the **a priori**;
- $\mathbb{P}(\mathbf{y}|\mathbf{w})$ is the **likelihood**;
- $\mathbb{P}(\mathbf{y})$ is the **marginal likelihood** or **model evidence**;

Background and Motivation

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{3}$$

- $\mathbb{P}(\mathbf{w}|\mathbf{y})$ is the **a posteriori**;
- $\mathbb{P}(\mathbf{w})$ is the **a priori**;
- $\mathbb{P}(\mathbf{y}|\mathbf{w})$ is the **likelihood**;
- $\mathbb{P}(\mathbf{y})$ is the **marginal likelihood** or **model evidence**;

Background and Motivation

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{3}$$

- $\mathbb{P}(\mathbf{w}|\mathbf{y})$ is the **a posteriori**;
- $\mathbb{P}(\mathbf{w})$ is the **a priori**;
- $\mathbb{P}(\mathbf{y}|\mathbf{w})$ is the **likelihood**;
- $\mathbb{P}(\mathbf{y})$ is the **marginal likelihood** or **model evidence**;

Markov Chain Monte Carlo

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{4}$$

- For most interesting models, the **model evidence** $\mathbb{P}(\mathbf{y})$ is often difficult to obtain.
- This is due to high-dimensional integration involved in $\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}$.
- And this is the motivation of the **Markov Chain Monte Carlo** (**MCMC**) algorithms.

Markov Chain Monte Carlo

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{4}$$

- For most interesting models, the **model evidence** $\mathbb{P}(\mathbf{y})$ is often difficult to obtain.
- This is due to high-dimensional integration involved in $\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}$.
- And this is the motivation of the **Markov Chain Monte Carlo** (**MCMC**) algorithms.

Markov Chain Monte Carlo

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{4}$$

- For most interesting models, the **model evidence** $\mathbb{P}(\mathbf{y})$ is often difficult to obtain.
- This is due to high-dimensional integration involved in $\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}$.
- And this is the motivation of the **Markov Chain Monte Carlo** (**MCMC**) algorithms.

Markov Chain Monte Carlo

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{4}$$

- For most interesting models, the **model evidence** $\mathbb{P}(\mathbf{y})$ is often difficult to obtain.
- This is due to high-dimensional integration involved in $\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}$.
- And this is the motivation of the **Markov Chain Monte Carlo** (**MCMC**) algorithms.

Metropolis-Hasting

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{5}$$

- The idea is to approximate the **posterior distribution** using sampling methods without having to compute the **model evidence**, $\mathbb{P}(\mathbf{y})$.
- The popular and simplest MCMC algorithm is the **Metropolis-Hasting** (MH).

Metropolis-Hasting

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\,\mathrm{d}\,\mathbf{w}} \tag{5}$$

- The idea is to approximate the **posterior distribution** using sampling methods without having to compute the **model evidence**, $\mathbb{P}(\mathbf{y})$.
- The popular and simplest MCMC algorithm is the **Metropolis-Hasting** (MH).

Metropolis-Hasting

Bayes' Theorem

Let $\mathbf{y}$ and $\mathbf{w}$ be the data and the weights, respectively, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{P}(\mathbf{y}|\mathbf{w})\mathbb{P}(\mathbf{w})}{\int \mathbb{P}(\mathbf{y}|\mathbf{w})\, \mathrm{d}\,\mathbf{w}} \tag{5}$$

- The idea is to approximate the **posterior distribution** using sampling methods without having to compute the **model evidence**, $\mathbb{P}(\mathbf{y})$.
- The popular and simplest MCMC algorithm is the **Metropolis-Hasting** (MH).

Background and Motivation

Algorithm 3 *Metropolis-Hasting MCMC*

Initialize $\mathbf{w}_r \sim \mathbb{G}(\mathbf{w}), r = 0$
for $r \in \{1, \cdots, r_{\max}\}$ **do**
  Propose: $\mathbf{w}_{new} \sim \mathbb{G}(\mathbf{w}_{new}|\mathbf{w}_{r-1})$
  Acceptance: $\alpha(\mathbf{w}_{new}|\mathbf{w}_{r-1}) \triangleq \min\left\{1, \frac{\mathbb{P}(\mathbf{w}_{new}|\mathbf{w}_{r-1})\mathbb{G}(\mathbf{w}_{r-1}|\mathbf{w}_{new})}{\mathbb{P}(\mathbf{w}_{r-1}|\mathbf{w}_{new})\mathbb{G}(\mathbf{w}_{new}|\mathbf{w}_{r-1})}\right\}$
  Draw $x \sim \mathrm{Unif}(0, 1)$
  if $x < \alpha(\mathbf{w}_{new}|\mathbf{w}_{r-1})$ **then**
    $\mathbf{w}_r \triangleq \mathbf{w}_{new}$
  else
    $\mathbf{w}_r \triangleq \mathbf{w}_{r-1}$
  end if
end for

Metropolis-Hasting

Metropolis-Hasting Limitations

- Specification of the proposal distribution, $\mathbb{G}$, is often difficult for high dimensional parameters.
- Autocorrelations of the markov chains is often high in magnitude, hence violates the assumption of IID samples.

- Due to the limitations of the Metropolis-Hasting algorithm, Bayesians have resorted to the use of Hamiltonian Monte Carlo (HMC).

Metropolis-Hasting

Metropolis-Hasting Limitations

- Specification of the proposal distribution, $\mathbb{G}$, is often difficult for high dimensional parameters.
- Autocorrelations of the markov chains is often high in magnitude, hence violates the assumption of IID samples.

- Due to the limitations of the Metropolis-Hasting algorithm, Bayesians have resorted to the use of Hamiltonian Monte Carlo (HMC).

Hybrid Monte Carlo

Hamiltonian Monte Carlo

- Originally known as **Hybrid Monte Carlo** in the paper by Duane et al. 1987, addresses the issue in the Metropolis-Hasting by considering **auxiliary variable** for describing the physical system in drawing samples from the **target distribution**.
- HMC is based on Hamiltonian dynamics.

- Hamiltonian dynamics describe the system using *location parameter* notated as $\mathbf{w}$ and *momentum parameter* $\mathbf{p}$.

Hybrid Monte Carlo

Hamiltonian Monte Carlo

- Originally known as **Hybrid Monte Carlo** in the paper by Duane et al. 1987, addresses the issue in the Metropolis-Hasting by considering **auxiliary variable** for describing the physical system in drawing samples from the **target distribution**.
- HMC is based on Hamiltonian dynamics.

- Hamiltonian dynamics describe the system using *location parameter* notated as $\mathbf{w}$ and *momentum parameter* $\mathbf{p}$.

Hybrid Monte Carlo

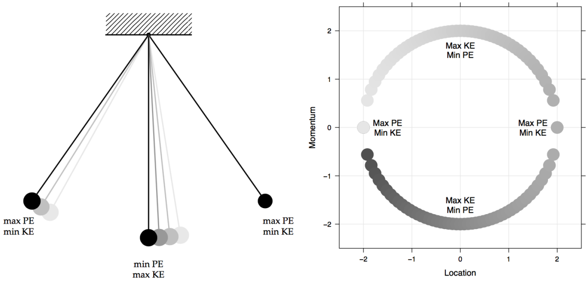
Hamiltonian Monte Carlo

- Originally known as **Hybrid Monte Carlo** in the paper by Duane et al. 1987, addresses the issue in the Metropolis-Hasting by considering **auxiliary variable** for describing the physical system in drawing samples from the **target distribution**.
- HMC is based on Hamiltonian dynamics.

- Hamiltonian dynamics describe the system using *location parameter* notated as $\mathbf{w}$ and *momentum parameter* $\mathbf{p}$.

Example of Hamiltonian Dynamics: Pendulum

- As an example, consider a ball attached to a frictionless pendulum swinging on a vertical plane.

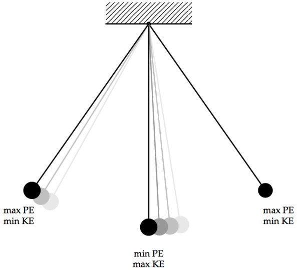


(a) Energies in Physical Pendulum. (b) Energies in Phase Space.

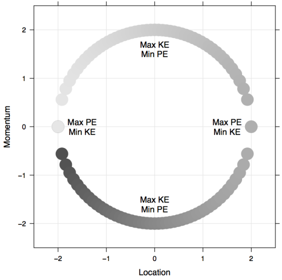
Figure: Conversion of Energies in Physical Pendulum and Phase Space.

Example of Hamiltonian Dynamics: Pendulum

- For each location of the ball given by $\mathbf{w}$, there is a corresponding *potential energy* (PE), denoted by $\mathbb{U}(\mathbf{w})$.



(a) Energies in Physical Pendulum.

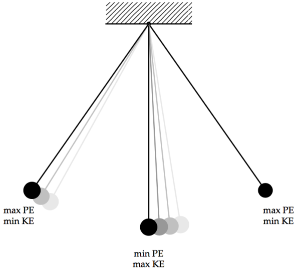


(b) Energies in Phase Space.

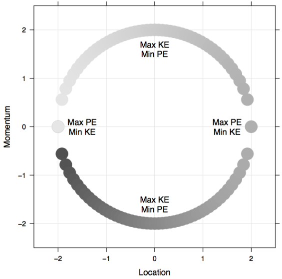
Figure: Conversion of Energies in Physical Pendulum and Phase Space.

Example of Hamiltonian Dynamics: Pendulum

- And for each momentum $\mathbf{p}$, there is an associated *kinetic energy* (KE) $\mathbb{K}(\mathbf{p})$.



(a) Energies in Physical Pendulum.

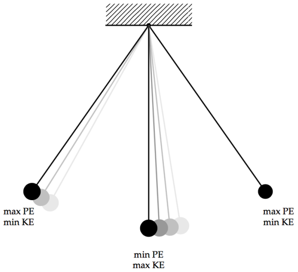


(b) Energies in Phase Space.

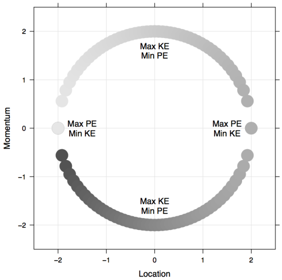
Figure: Conversion of Energies in Physical Pendulum and Phase Space.

Example of Hamiltonian Dynamics: Pendulum

- So that at the extreme trajectory of the pendulum, the PE is maximum and KE is minimum;



(a) Energies in Physical Pendulum.

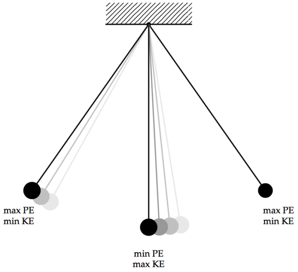


(b) Energies in Phase Space.

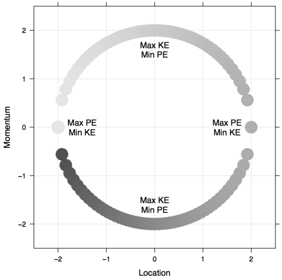
Figure: Conversion of Energies in Physical Pendulum and Phase Space.

Example of Hamiltonian Dynamics: Pendulum

- And at the equilibrium point, the KE is maximum and PE is minimum.



(a) Energies in Physical Pendulum.

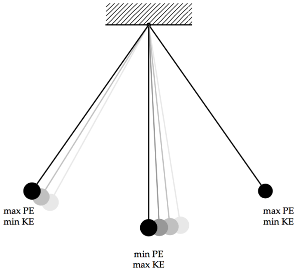


(b) Energies in Phase Space.

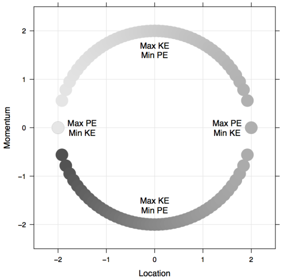
Figure: Conversion of Energies in Physical Pendulum and Phase Space.

Example of Hamiltonian Dynamics: Pendulum

- The system is a function of time, hence the Hamiltonian dynamics evolve in a continuous space called **phase space**.



(a) Energies in Physical Pendulum.

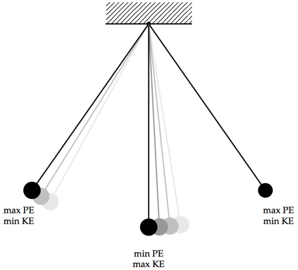


(b) Energies in Phase Space.

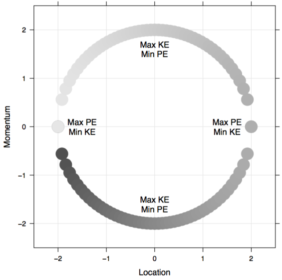
Figure: Conversion of Energies in Physical Pendulum and Phase Space.

Example of Hamiltonian Dynamics: Pendulum

- The total energy of the system is characterized by the Hamiltonian $\mathbb{H}(\mathbf{w}, \mathbf{p}) \triangleq \mathbb{U}(\mathbf{w}) + \mathbb{K}(\mathbf{p})$.



(a) Energies in Physical Pendulum.

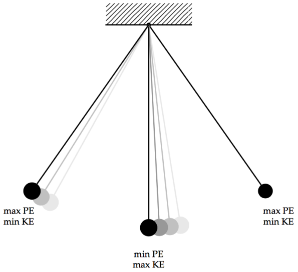


(b) Energies in Phase Space.

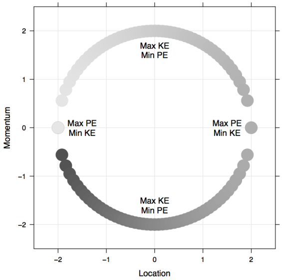
Figure: Conversion of Energies in Physical Pendulum and Phase Space.

Example of Hamiltonian Dynamics: Pendulum

- And therefore describes the conversion of the two energies as the object moves throughout a system in time.



(a) Energies in Physical Pendulum.



(b) Energies in Phase Space.

Figure: Conversion of Energies in Physical Pendulum and Phase Space.

Hamiltonian Equations

- So that the following are the Hamiltonian equations:

$$\begin{aligned}\frac{\mathrm{d}\,\mathbf{w}}{\mathrm{d}\,t} &= \frac{\partial \mathbb{H}(\mathbf{w},\mathbf{p})}{\partial \mathbf{p}} = \frac{\mathrm{d}\,\mathbb{K}(\mathbf{p})}{\mathrm{d}\,\mathbf{p}} \\ \frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} &= -\frac{\partial \mathbb{H}(\mathbf{w},\mathbf{p})}{\partial \mathbf{w}} = -\frac{\mathrm{d}\,\mathbb{U}(\mathbf{w})}{\mathrm{d}\,\mathbf{w}}.\end{aligned} \tag{6}$$

Properties of Hamiltonian Dynamics

Three Properties of Hamiltonian Dynamics

There are three properties that makes Hamiltonian dynamics good for sampling methods:

- *Conservation* of the energy;
- The system *preserves* the volume of the phase space. This follows from *Liouville's theorem*;
- And the last property is *reversibility*.

Time Discretization for Hamiltonian Dynamics

- Since the phase space changes over time which is a continuous variable, then in order to simulate the Hamiltonian dynamics under numerical computations, the time has to be discretized.
- And there are several ways to do this, one such solution is to consider the leapfrog method.

Time Discretization for Hamiltonian Dynamics

- Since the phase space changes over time which is a continuous variable, then in order to simulate the Hamiltonian dynamics under numerical computations, the time has to be discretized.
- And there are several ways to do this, one such solution is to consider the leapfrog method.

Time Discretization for Hamiltonian Dynamics

Leap Frog Method

Let $\mathbf{w}, \mathbf{p}, \mathbb{U}, \mathbb{K}$ and γ be the location, momentum, potential, kinetic and step size parameters, respectively, then

$$\mathbf{p}(t+\gamma/2) = \mathbf{p}(t) - (\gamma/2)\frac{\partial \mathbb{U}(\mathbf{w}(t))}{\partial \mathbf{w}(t)} \tag{7}$$

$$\mathbf{w}(t+\gamma) = \mathbf{w}(t) + \gamma\frac{\partial \mathbb{K}(\mathbf{p}(t))}{\partial \mathbf{p}(t)}, \tag{8}$$

$$\mathbf{p}(t+\gamma) = \mathbf{p}(t+\gamma/2) - (\gamma/2)\frac{\partial \mathbb{U}(\mathbf{w}(t+\gamma))}{\partial \mathbf{w}(t)} \tag{9}$$

Hamiltonian Dynamics for MCMC

So how is Hamiltonian dynamics linked to MCMC?

Canonical Distribution

The total energy is related to the probability distribution of the parameter of interest using the concept of **canonical distribution** from the Statistical Mechanics. That is,

$$\mathbb{P}(\mathbf{w}) = \frac{1}{Z} \exp\left[-E(\mathbf{w})\right], \tag{10}$$

where E is the total energy.

- Therefore, E in this case, is $\mathbb{H}(\mathbf{w}, \mathbf{p})$.
- And using the three properties of Hamiltonian dynamics mentioned earlier, the canonical distribution is *invariant*.

Hamiltonian Dynamics for MCMC

So how is Hamiltonian dynamics linked to MCMC?

Canonical Distribution

The total energy is related to the probability distribution of the parameter of interest using the concept of **canonical distribution** from the Statistical Mechanics. That is,

$$\mathbb{P}(\mathbf{w}) = \frac{1}{Z}\exp\left[-E(\mathbf{w})\right], \tag{10}$$

where E is the total energy.

- Therefore, E in this case, is $\mathbb{H}(\mathbf{w}, \mathbf{p})$.
- And using the three properties of Hamiltonian dynamics mentioned earlier, the canonical distribution is *invariant*.

Hamiltonian Dynamics for MCMC

So how is Hamiltonian dynamics linked to MCMC?

Canonical Distribution

The total energy is related to the probability distribution of the parameter of interest using the concept of **canonical distribution** from the Statistical Mechanics. That is,

$$\mathbb{P}(\mathbf{w}) = \frac{1}{Z} \exp\left[-E(\mathbf{w})\right], \tag{10}$$

where E is the total energy.

- Therefore, E in this case, is $\mathbb{H}(\mathbf{w}, \mathbf{p})$.
- And using the three properties of Hamiltonian dynamics mentioned earlier, the canonical distribution is *invariant*.

Hamiltonian Dynamics for MCMC

So that the equation $\mathbb{P}(\mathbf{w}) = \frac{1}{Z}\exp\left[-E(\mathbf{w})\right]$, becomes

$$\begin{aligned}\mathbb{P}(\mathbf{w},\mathbf{p}) &\propto \exp\left[-\mathbb{H}(\mathbf{w},\mathbf{p})\right]\\ &= \exp\left[-\mathbb{U}(\mathbf{w}) - \mathbb{K}(\mathbf{p})\right]\\ &= \exp\left[-\mathbb{U}(\mathbf{w})\right]\exp\left[-\mathbb{K}(\mathbf{p})\right]\\ &\propto \mathbb{P}(\mathbf{w})\mathbb{P}(\mathbf{p}).\end{aligned}$$

Therefore the joint canonical distribution of the location parameter $\mathbf{w}$ and the momentum parameter $\mathbf{p}$ factors into the products of its marginal density, implying independence.

Hamiltonian Dynamics for MCMC

So that the equation $\mathbb{P}(\mathbf{w}) = \frac{1}{Z}\exp\left[-E(\mathbf{w})\right]$, becomes

$$\begin{aligned}
\mathbb{P}(\mathbf{w}, \mathbf{p}) &\propto \exp\left[-\mathbb{H}(\mathbf{w}, \mathbf{p})\right] \\
&= \exp\left[-\mathbb{U}(\mathbf{w}) - \mathbb{K}(\mathbf{p})\right] \\
&= \exp\left[-\mathbb{U}(\mathbf{w})\right]\exp\left[-\mathbb{K}(\mathbf{p})\right] \\
&\propto \mathbb{P}(\mathbf{w})\mathbb{P}(\mathbf{p}).
\end{aligned}$$

Therefore the joint canonical distribution of the location parameter **w** and the momentum parameter **p** factors into the products of its marginal density, implying independence.

Hamiltonian Dynamics for MCMC

So that the parameter of interest $\mathbf{w}$ has the following target distribution,

$$\begin{aligned} \mathbb{U}(\mathbf{w}) &= -\log \mathbb{P}(\mathbf{w}|\mathbf{y}) && (11) \\ &= -\log[\mathbb{P}(\mathbf{w})\mathcal{L}(\mathbf{w}|\mathbf{y})] - \mathcal{C}, && (12) \end{aligned}$$

where $\mathcal{C} \triangleq \log \mathbb{P}(\mathbf{y})$.

Hamiltonian Dynamics for MCMC

The kinetic energy is often assumed to be standard Gaussian distributed, and thus

$$\mathbb{K}(\mathbf{p}, \boldsymbol{\mu} \triangleq \mathbf{0}, \boldsymbol{\Sigma} \triangleq \mathbf{I}) = \frac{(\mathbf{p} - \boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{p} - \boldsymbol{\mu})}{2} = \frac{\mathbf{p}^{\mathsf{T}}\mathbf{p}}{2}. \tag{13}$$

Hamiltonian Dynamics for MCMC

Algorithm 5 *Hamiltonian MCMC*

1: Initialize Leap Frog parameters: γ and τ;
2: Set initial location $\mathbf{w}^{(r=0)}(t=0)$;
3: **for** $r \in \{0, \cdots, r_{\max}\}$ **do**
4: Draw initial momentum, $\mathbf{p}^{(r)}(t=0) \sim \frac{\exp[-\mathbb{K}(\mathbf{p})]}{Z}$;
5: Compute $\mathbb{H}(\mathbf{w}^{(r)}(0), \mathbf{p}^{(r)}(0)) \triangleq \mathbb{U}(\mathbf{w}^{(r)}(0)) + \mathbb{K}(\mathbf{p}^{(r)}(0))$;
6: Simulate Hamiltonian dynamics using Leap Frog:
7: **for** $t \in \{0, \cdots, \tau\}$ **do**

$$\mathbf{p}^{(r)}(t+\gamma/2) \triangleq \mathbf{p}^{(r)}(t) - (\gamma/2)\frac{\partial \mathbb{U}(\mathbf{w}^{(r)}(t))}{\partial \mathbf{w}^{(r)}(t)} \tag{4.7}$$

$$\mathbf{w}^{(r)}(t+\gamma) \triangleq \mathbf{w}^{(r)}(t) + \gamma\frac{\partial \mathbb{K}(\mathbf{p}^{(r)}(t+1))}{\partial \mathbf{p}^{(r)}(t+1)}, \tag{4.8}$$

$$\mathbf{p}^{(r)}(t+\gamma) \triangleq \mathbf{p}^{(r)}(t+\gamma/2) - (\gamma/2)\frac{\partial \mathbb{U}(\mathbf{w}^{(r)}(t+\gamma))}{\partial \mathbf{w}^{(r)}(t+\gamma)} \tag{4.9}$$

8: **end for**
9: **if** $\Delta\mathbb{H} < 0$ **then**
10: $\mathbf{w}^{(r+1)}(0) \triangleq \mathbf{w}^{(r)}(\tau+\gamma)$
11: **else**
12: **if** $a < \exp(\Delta\mathbb{H}), a \sim \text{Unif}(0,1)$ **then**
13: $\mathbf{w}^{(r+1)}(0) \triangleq \mathbf{w}^{(r)}(\tau+\gamma)$
14: **else**
15: $\mathbf{w}^{(r+1)}(0) \triangleq \mathbf{w}^{(r)}(0)$
16: **end if**
17: **end if**

Langevin Dynamics

- The Stochastic Gradient HMC works by considering Langevin dynamics on its momentum.
- The said dynamics extend the idea of the Newton's second law of motion.
- Originally, the second law proceeds as follows: let $\mathbf{f}$ be the force, $\mathbf{p}$ be the momentum, m be the mass, $\mathbf{v}$ be the veclocity, and $\mathbf{a}$ be the acceleration, then

$$\mathbf{f} = \frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} = \frac{\mathrm{d}(m\mathbf{v})}{\mathrm{d}\,t} = m\frac{\mathrm{d}\,\mathbf{v}}{\mathrm{d}\,t} = m\mathbf{a}. \tag{14}$$

Langevin Dynamics

- The Stochastic Gradient HMC works by considering Langevin dynamics on its momentum.
- The said dynamics extend the idea of the Newton's second law of motion.
- Originally, the second law proceeds as follows: let $\mathbf{f}$ be the force, $\mathbf{p}$ be the momentum, m be the mass, $\mathbf{v}$ be the veclocity, and $\mathbf{a}$ be the acceleration, then

$$\mathbf{f} = \frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} = \frac{\mathrm{d}(m\mathbf{v})}{\mathrm{d}\,t} = m\frac{\mathrm{d}\,\mathbf{v}}{\mathrm{d}\,t} = m\mathbf{a}. \tag{14}$$

Langevin Dynamics

- The Stochastic Gradient HMC works by considering Langevin dynamics on its momentum.
- The said dynamics extend the idea of the Newton's second law of motion.
- Originally, the second law proceeds as follows: let $\mathbf{f}$ be the force, $\mathbf{p}$ be the momentum, m be the mass, $\mathbf{v}$ be the veclocity, and $\mathbf{a}$ be the acceleration, then

$$\mathbf{f} = \frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} = \frac{\mathrm{d}(m\mathbf{v})}{\mathrm{d}\,t} = m\frac{\mathrm{d}\,\mathbf{v}}{\mathrm{d}\,t} = m\mathbf{a}. \tag{14}$$

Langevin Dynamics

- The Stochastic Gradient HMC works by considering Langevin dynamics on its momentum.
- The said dynamics extend the idea of the Newton's second law of motion.
- Originally, the second law proceeds as follows: let $\mathbf{f}$ be the force, $\mathbf{p}$ be the momentum, m be the mass, $\mathbf{v}$ be the veclocity, and $\mathbf{a}$ be the acceleration, then

$$\mathbf{f} = \frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} = \frac{\mathrm{d}(m\mathbf{v})}{\mathrm{d}\,t} = m\frac{\mathrm{d}\,\mathbf{v}}{\mathrm{d}\,t} = m\mathbf{a}. \tag{14}$$

Langevin Dynamics

- The idea of Langevin dynamics is to take into account or at least approximate the effect of neglected degrees of freedom;
- and this is achieved by adding two force terms: one represents the frictional force, $\eta\mathbf{v}^{\star}$; and the other represents the random force, $\mathbf{e}$.
- So that the Langevin equation is given below:

$$\frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} - \eta\mathbf{v}^{\star} + \mathbf{e} = m\mathbf{a}, \tag{15}$$

where the random force is assumed to have zero mean and is uncorrelated, i.e. $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \xi\mathbf{I})$.

Langevin Dynamics

- The idea of Langevin dynamics is to take into account or at least approximate the effect of neglected degrees of freedom;
- and this is achieved by adding two force terms: one represents the frictional force, $\eta \mathbf{v}^{\blacklozenge}$; and the other represents the random force, $\mathbf{e}$.
- So that the Langevin equation is given below:

$$\frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} - \eta \mathbf{v}^{\blacklozenge} + \mathbf{e} = m\mathbf{a}, \tag{15}$$

where the random force is assumed to have zero mean and is uncorrelated, i.e. $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \xi \mathbf{I})$.

Langevin Dynamics

- The idea of Langevin dynamics is to take into account or at least approximate the effect of neglected degrees of freedom;
- and this is achieved by adding two force terms: one represents the frictional force, $\eta\mathbf{v}^{\star}$; and the other represents the random force, $\mathbf{e}$.
- So that the Langevin equation is given below:

$$\frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} - \eta\mathbf{v}^{\star} + \mathbf{e} = m\mathbf{a}, \tag{15}$$

where the random force is assumed to have zero mean and is uncorrelated, i.e. $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \xi\mathbf{I})$.

Langevin Dynamics

- The idea of Langevin dynamics is to take into account or at least approximate the effect of neglected degrees of freedom;
- and this is achieved by adding two force terms: one represents the frictional force, $\eta \mathbf{v}^{\blacklozenge}$; and the other represents the random force, $\mathbf{e}$.
- So that the Langevin equation is given below:

$$\frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} - \eta \mathbf{v}^{\blacklozenge} + \mathbf{e} = m\mathbf{a}, \tag{15}$$

where the random force is assumed to have zero mean and is uncorrelated, i.e. $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \xi \mathbf{I})$.

Mathematical Programming

- The idea behind **Stochastic Gradient Hamiltonian Monte Carlo** (SGHMC) is based on speeding up the numerical computations in optimization problems;
- In optimization, the popular algorithm for minimzing a function is the **Gradient Descent**;
- In classical statistics, this is also called **Batch Gradient Descent** when minimizing the residual sum of square;
- It is called "**Batch**", since it uses all data points in computing the gradient.

Mathematical Programming

- The idea behind **Stochastic Gradient Hamiltonian Monte Carlo** (SGHMC) is based on speeding up the numerical computations in optimization problems;
- In optimization, the popular algorithm for minimzing a function is the **Gradient Descent**;
- In classical statistics, this is also called **Batch Gradient Descent** when minimizing the residual sum of square;
- It is called "**Batch**", since it uses all data points in computing the gradient.

Mathematical Programming

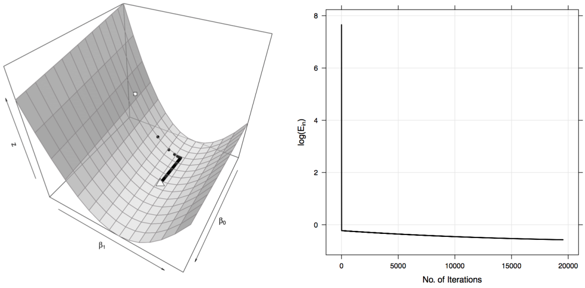
- The idea behind **Stochastic Gradient Hamiltonian Monte Carlo** (SGHMC) is based on speeding up the numerical computations in optimization problems;
- In optimization, the popular algorithm for minimzing a function is the **Gradient Descent**;
- In classical statistics, this is also called **Batch Gradient Descent** when minimizing the residual sum of square;
- It is called "**Batch**", since it uses all data points in computing the gradient.

Mathematical Programming

- The idea behind **Stochastic Gradient Hamiltonian Monte Carlo** (SGHMC) is based on speeding up the numerical computations in optimization problems;
- In optimization, the popular algorithm for minimzing a function is the **Gradient Descent**;
- In classical statistics, this is also called **Batch Gradient Descent** when minimizing the residual sum of square;
- It is called "**Batch**", since it uses all data points in computing the gradient.

Batch Gradient Descent

Consider the following error surface function of the simple linear regression model:



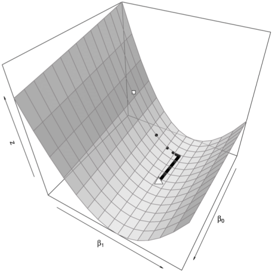
(a) BGD on Loss Function Surface.

(b) Loss Function under BGD.

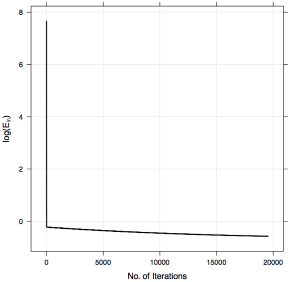
Figure: Batch Gradient Descent on SLR Loss Function.

Gradient Descent

Batch gradient descent can be very computationally expensive especially for large dataset.



(a) BGD on Loss Function Surface.



(b) Loss Function under BGD.

Figure: Batch Gradient Descent on SLR Loss Function.

Stochastic Gradient Descent

- One might suggest that instead of using **all observations**, would it be feasible to just use **one** or **sample of observations**?
- The answer is **Yes**! and that is the idea behind **Stochastic Gradient Descent** (SGD).
- SGD updates the parameter using only one observation for every iteration, which is a lot faster.
- Sometimes it is called **Minibatch Gradient Descent** (MGD) if it uses **samples of observations**.
- MGD can take advantage of vectorization in computation and hence even faster than SGD;

Stochastic Gradient Descent

- One might suggest that instead of using **all observations**, would it be feasible to just use **one** or **sample of observations**?
- The answer is **Yes**! and that is the idea behind **Stochastic Gradient Descent** (SGD).
- SGD updates the parameter using only one observation for every iteration, which is a lot faster.
- Sometimes it is called **Minibatch Gradient Descent** (MGD) if it uses **samples of observations**.
- MGD can take advantage of vectorization in computation and hence even faster than SGD;

Stochastic Gradient Descent

- One might suggest that instead of using **all observations**, would it be feasible to just use **one** or **sample of observations**?
- The answer is **Yes**! and that is the idea behind **Stochastic Gradient Descent** (SGD).
- SGD updates the parameter using only one observation for every iteration, which is a lot faster.
- Sometimes it is called **Minibatch Gradient Descent** (MGD) if it uses **samples of observations**.
- MGD can take advantage of vectorization in computation and hence even faster than SGD;

Stochastic Gradient Descent

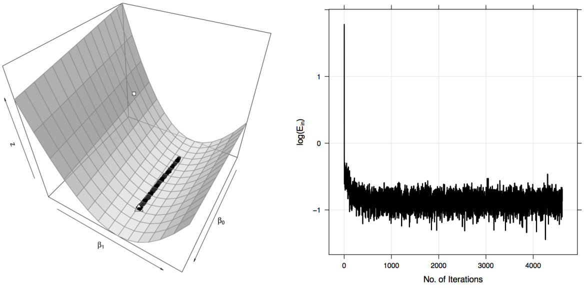
- One might suggest that instead of using **all observations**, would it be feasible to just use **one** or **sample of observations**?
- The answer is **Yes**! and that is the idea behind **Stochastic Gradient Descent** (SGD).
- SGD updates the parameter using only one observation for every iteration, which is a lot faster.
- Sometimes it is called **Minibatch Gradient Descent** (MGD) if it uses **samples of observations**.
- MGD can take advantage of vectorization in computation and hence even faster than SGD;

Stochastic Gradient Descent

- One might suggest that instead of using **all observations**, would it be feasible to just use **one** or **sample of observations**?
- The answer is **Yes**! and that is the idea behind **Stochastic Gradient Descent** (SGD).
- SGD updates the parameter using only one observation for every iteration, which is a lot faster.
- Sometimes it is called **Minibatch Gradient Descent** (MGD) if it uses **samples of observations**.
- MGD can take advantage of vectorization in computation and hence even faster than SGD;

Stochastic Gradient Descent

Consider again the following error surface function of the simple linear regression model:



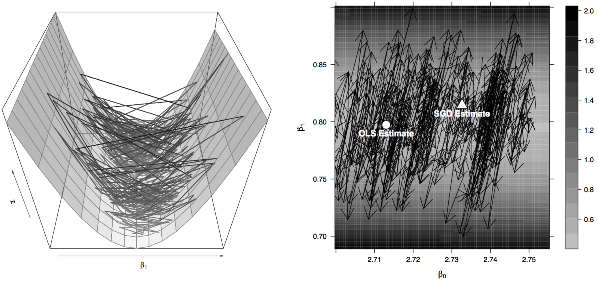
(a) SGD on Loss Function Surface.

(b) Loss Function under SGD.

Figure: Stochastic Batch Gradient Descent on SLR Loss Function.

Stochastic Gradient Descent

Zoom into the SGD gradient vectors.



(a) SGD on Loss Function Surface. (b) SGD on Loss Function Contour.

Figure: A Closer Look at SGD Gradient Vectors.

Stochastic Gradient Hamiltonian Monte Carlo

- To take advantage of the features of SGD, Bayesians decided to apply this algorithm to MCMC methods involving gradient computation.
- Such MCMCs are the **HMC** and **LMC**.
- For HMC in particular, Chen, Fox, and Guestrin 2014 were the pioneers for marrying the two methods.
- Their work has been inspired by Ahn, Korattikara, and Welling 2012; and Welling and Teh 2011.

Stochastic Gradient Hamiltonian Monte Carlo

- To take advantage of the features of SGD, Bayesians decided to apply this algorithm to MCMC methods involving gradient computation.
- Such MCMCs are the **HMC** and **LMC**.
- For HMC in particular, Chen, Fox, and Guestrin 2014 were the pioneers for marrying the two methods.
- Their work has been inspired by Ahn, Korattikara, and Welling 2012; and Welling and Teh 2011.

Stochastic Gradient Hamiltonian Monte Carlo

- To take advantage of the features of SGD, Bayesians decided to apply this algorithm to MCMC methods involving gradient computation.
- Such MCMCs are the **HMC** and **LMC**.
- For HMC in particular, Chen, Fox, and Guestrin 2014 were the pioneers for marrying the two methods.
- Their work has been inspired by Ahn, Korattikara, and Welling 2012; and Welling and Teh 2011.

Stochastic Gradient Hamiltonian Monte Carlo

- To take advantage of the features of SGD, Bayesians decided to apply this algorithm to MCMC methods involving gradient computation.
- Such MCMCs are the **HMC** and **LMC**.
- For HMC in particular, Chen, Fox, and Guestrin 2014 were the pioneers for marrying the two methods.
- Their work has been inspired by Ahn, Korattikara, and Welling 2012; and Welling and Teh 2011.

Stochastic Gradient Hamiltonian Monte Carlo

- Although Chen, Fox, and Guestrin 2014 formulated the theory of the SGHMC, there is no discussion as to how well is the mixing of the samples;
- There is also no empirical analysis on the convergence of the algorithm to the posterior distribution.
- Further, the fact that the SGHMC was published only a couple of years ago, there is no theoretical results yet for simple model estimated using the said algorithm;
- For example for the case of linear regression models.

Stochastic Gradient Hamiltonian Monte Carlo

- Although Chen, Fox, and Guestrin 2014 formulated the theory of the SGHMC, there is no discussion as to how well is the mixing of the samples;
- There is also no empirical analysis on the convergence of the algorithm to the posterior distribution.
- Further, the fact that the SGHMC was published only a couple of years ago, there is no theoretical results yet for simple model estimated using the said algorithm;
- For example for the case of linear regression models.

Stochastic Gradient Hamiltonian Monte Carlo

- Although Chen, Fox, and Guestrin 2014 formulated the theory of the SGHMC, there is no discussion as to how well is the mixing of the samples;
- There is also no empirical analysis on the convergence of the algorithm to the posterior distribution.
- Further, the fact that the SGHMC was published only a couple of years ago, there is no theoretical results yet for simple model estimated using the said algorithm;
- For example for the case of linear regression models.

Stochastic Gradient Hamiltonian Monte Carlo

- Although Chen, Fox, and Guestrin 2014 formulated the theory of the SGHMC, there is no discussion as to how well is the mixing of the samples;
- There is also no empirical analysis on the convergence of the algorithm to the posterior distribution.
- Further, the fact that the SGHMC was published only a couple of years ago, there is no theoretical results yet for simple model estimated using the said algorithm;
- For example for the case of linear regression models.

Stochastic Gradient Hamiltonian Monte Carlo

- To formally begin, let $\tilde{\mathscr{D}}$ be the minibatch or sample of the full dataset $\mathscr{D}$. Then $\tilde{\mathscr{D}} \subseteq \mathscr{D}$, implies that

$$\nabla \tilde{\mathbb{U}}(\mathbf{w}) = -\frac{|\mathscr{D}|}{|\tilde{\mathscr{D}}|} \sum_{\mathbf{x} \in \tilde{\mathscr{D}}} \nabla \log \mathbb{P}(\mathbf{x}|\mathbf{w}) - \nabla \log \mathbb{P}(\mathbf{w}). \tag{16}$$

The minibatch above is uniformly sampled from $\mathscr{D}$, and by doing so, the weight $\frac{|\mathscr{D}|}{|\tilde{\mathscr{D}}|}$ makes $\nabla \tilde{\mathbb{U}}(\mathbf{w})$ an estimate to $\nabla \mathbb{U}(\mathbf{w})$.

Stochastic Gradient Hamiltonian Monte Carlo

- To formally begin, let $\tilde{\mathscr{D}}$ be the minibatch or sample of the full dataset $\mathscr{D}$. Then $\tilde{\mathscr{D}} \subseteq \mathscr{D}$, implies that

$$\nabla\tilde{\mathbb{U}}(\mathbf{w}) = -\frac{|\mathscr{D}|}{|\tilde{\mathscr{D}}|}\sum_{\mathbf{x}\in\tilde{\mathscr{D}}}\nabla\log\mathbb{P}(\mathbf{x}|\mathbf{w}) - \nabla\log\mathbb{P}(\mathbf{w}). \tag{16}$$

The minibatch above is uniformly sampled from $\mathscr{D}$, and by doing so, the weight $\frac{|\mathscr{D}|}{|\tilde{\mathscr{D}}|}$ makes $\nabla\tilde{\mathbb{U}}(\mathbf{w})$ an estimate to $\nabla\mathbb{U}(\mathbf{w})$.

Stochastic Gradient Hamiltonian Monte Carlo

- The error of this estimate which in this case is known as the *stochastic gradient noise*, is given by

$$\nabla\tilde{\mathbb{U}}(\mathbf{w}) - \nabla\mathbb{U}(\mathbf{w}) = \xi. \tag{17}$$

Obviously, $\mathbb{E}[\nabla\tilde{\mathbb{U}}(\mathbf{w})] = \nabla\mathbb{U}(\mathbf{w})$ hence

$$\mathbb{E}[\nabla\tilde{\mathbb{U}}(\mathbf{w}) - \nabla\mathbb{U}(\mathbf{w})] = \mathbb{E}[\xi] = \mathbf{0}. \tag{18}$$

Stochastic Gradient Hamiltonian Monte Carlo

- The error of this estimate which in this case is known as the *stochastic gradient noise*, is given by

$$\nabla\tilde{\mathbb{U}}(\mathbf{w}) - \nabla\mathbb{U}(\mathbf{w}) = \xi. \tag{17}$$

Obviously, $\mathbb{E}[\nabla\tilde{\mathbb{U}}(\mathbf{w})] = \nabla\mathbb{U}(\mathbf{w})$ hence

$$\mathbb{E}[\nabla\tilde{\mathbb{U}}(\mathbf{w}) - \nabla\mathbb{U}(\mathbf{w})] = \mathbb{E}[\xi] = \mathbf{0}. \tag{18}$$

Stochastic Gradient Hamiltonian Monte Carlo

- Let $\mathbb{V}\text{ar}[\boldsymbol{\xi}] = \mathfrak{A}(\mathbf{w})$ be the variance-covariance matrix of the stochastic gradient noise, then by central limit theorem (CLT), $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathfrak{A}(\mathbf{w}))$. And therefore $\tilde{\mathbb{U}}(\mathbf{w})$ is approximated as follows:

$$\nabla\tilde{\mathbb{U}}(\mathbf{w}) \approx \nabla\mathbb{U}(\mathbf{w}) + \boldsymbol{\xi}, \quad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathfrak{A}(\mathbf{w})). \tag{19}$$

- In effect, the momentum update of the HMC algorithm now has a noise term added. That is, $\Delta\mathbf{p} = -\gamma\nabla\tilde{\mathbb{U}}(\mathbf{w})$, so that $\mathbb{V}\text{ar}[-\gamma\boldsymbol{\xi}] = \gamma^2\mathfrak{A}(\mathbf{w})$ or $2\mathfrak{B}(\mathbf{w})$ where $\mathfrak{B}(\mathbf{w}) = \frac{1}{2}\gamma\mathfrak{A}(\mathbf{w})$ is the diffusion matrix.

Stochastic Gradient Hamiltonian Monte Carlo

- Let $\mathbb{V}\text{ar}[\boldsymbol{\xi}] = \mathfrak{A}(\mathbf{w})$ be the variance-covariance matrix of the stochastic gradient noise, then by central limit theorem (CLT), $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathfrak{A}(\mathbf{w}))$. And therefore $\tilde{\mathbb{U}}(\mathbf{w})$ is approximated as follows:

$$\nabla\tilde{\mathbb{U}}(\mathbf{w}) \approx \nabla\mathbb{U}(\mathbf{w}) + \boldsymbol{\xi}, \quad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathfrak{A}(\mathbf{w})). \tag{19}$$

- In effect, the momentum update of the HMC algorithm now has a noise term added. That is, $\Delta\mathbf{p} = -\gamma\nabla\tilde{\mathbb{U}}(\mathbf{w})$, so that $\mathbb{V}\text{ar}[-\gamma\boldsymbol{\xi}] = \gamma^2\mathfrak{A}(\mathbf{w})$ or $2\mathfrak{B}(\mathbf{w})$ where $\mathfrak{B}(\mathbf{w}) = \frac{1}{2}\gamma\mathfrak{A}(\mathbf{w})$ is the diffusion matrix.

Stochastic Gradient Hamiltonian Monte Carlo

- So if the batch size, $|\tilde{\mathscr{D}}|$, becomes small, then the variability of ξ becomes large. The resulting discrete time system can be viewed as a γ-discretization of the following continuous stochastic differential equation:

$$\frac{\mathrm{d}\,\mathbf{w}}{\mathrm{d}\,t} = \mathbf{\Sigma}^{-1}\mathbf{p} \quad \text{and} \quad \frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} \approx -\nabla\mathbb{U}(\mathbf{w}) + \xi^{\bigstar}, \tag{20}$$

where $\xi^{\bigstar} \sim \mathcal{N}(\mathbf{0}, 2\mathfrak{B})$.

Stochastic Gradient Hamiltonian Monte Carlo

- And because of this term, the preservation of the entropy under the Hamiltonian dynamics is not anymore satisfied.
- This is shown in one of the results from Chen, Fox and Guestrin 2014, please refer to the said article for the theoretical results of the entropy of the target density.

Stochastic Gradient Hamiltonian Monte Carlo

- To address the problem presented above, a friction term is added to the equation. And this introduces a correction step even before considering errors introduced by the discretization of the dynamical system. So that Equation (20) becomes

$$\frac{\mathrm{d}\,\mathbf{w}}{\mathrm{d}\,t} = \boldsymbol{\Sigma}^{-1}\mathbf{p} \quad \text{and} \quad \frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} = -\nabla\mathbb{U}(\mathbf{w}) - \mathfrak{B}\boldsymbol{\Sigma}^{-1}\mathbf{p} + \xi^{\bigstar}. \quad (21)$$

where $\xi^{\bigstar} \sim \mathcal{N}(\mathbf{0}, 2\mathfrak{B})$.

Stochastic Gradient Hamiltonian Monte Carlo in Practice

- The parameter $\mathfrak{B}$ up to this point is assumed to be known. However, this is not the case in real scenario. So a remedy is to consider an estimate of $\mathfrak{B}$ instead, denoted as $\hat{\mathfrak{B}}$, and define a user-specified friction term $\mathfrak{C} \succeq \hat{\mathfrak{B}}$. That is $\mathfrak{C} - \hat{\mathfrak{B}} \succeq 0$ suggests that the matrix $\mathfrak{C} - \hat{\mathfrak{B}}$ is positive-semidefinite. So that the dynamics becomes

$$\frac{\mathrm{d}\,\mathbf{w}}{\mathrm{d}\,t} = \mathbf{\Sigma}^{-1}\mathbf{p} \quad \text{and} \quad \frac{\mathrm{d}\,\mathbf{p}}{\mathrm{d}\,t} = -\nabla\tilde{\mathbb{U}}(\mathbf{w}) - \mathfrak{C}\mathbf{\Sigma}^{-1}\mathbf{p} + \xi^{\star}. \tag{22}$$

Stochastic Gradient Hamiltonian Monte Carlo

Algorithm 6 *Stochastic Hamiltonian MCMC*

Initialize Leapfrog parameters: γ and τ;
Initialize estimate for $\hat{\mathfrak{B}}(\mathbf{w}) = \frac{\gamma}{2}\hat{\mathfrak{A}}(\mathbf{w})$, and specify the matrix $\mathfrak{C}$;
Set initial location $\mathbf{w}^{(r=0)}(t=0)$;
for $r \in \{0, \cdots, r_{\max}\}$ **do**
    Draw initial momentum, $\mathbf{p}^{(r)}(t=0) \sim \frac{\exp[-\mathbb{K}(\mathbf{p})]}{Z}$;
    Simulate Hamiltonian dynamics using Leapfrog:
    for $t \in \{0, \cdots, \tau\}$ **do**

$$\Delta\mathbf{w}^{(r)}(t+\gamma) \triangleq \gamma\nabla_{\mathbf{p}^{(r)}(t+1)}\mathbb{K}(\mathbf{p}^{(r)}(t+1)), \tag{4.31}$$

$$\Delta\mathbf{p}^{(r)}(t+\gamma) \triangleq -\gamma\nabla_{\mathbf{w}^{(r)}(t)}\tilde{\mathbb{U}}(\mathbf{w}^{(r)}(t)) - \mathfrak{C}(\mathbf{w}^{(r)}(t))\boldsymbol{\Sigma}^{-1}\mathbf{p} + \xi^{\star} \tag{4.32}$$

$$\text{where } \xi^{\star} \sim \mathcal{N}(\mathbf{0}, 2\gamma(\mathfrak{C} - \mathfrak{B})) \tag{4.33}$$

    end for
    $\mathbf{w}^{(r+1)} \triangleq \mathbf{w}^{(r)}$
end for

Autoregressive Distributed Lag (ADL) Model

- In this thesis, the objective model is the Autoregressive Distributed Lag (ADL) which is a specialized type of dynamic linear models (L. J. Welty et al 2009).
- In particular, the response variable of this model is dependent on predictors which includes the *autoregressive* term — the lag values of the response; and the *distributed lag* term — other explanatory variables known (or tested) to have effect on the response variable.

Autoregressive Distributed Lag (ADL) Model

- In this thesis, the objective model is the Autoregressive Distributed Lag (ADL) which is a specialized type of dynamic linear models (L. J. Welty et al 2009).
- In particular, the response variable of this model is dependent on predictors which includes the *autoregressive* term — the lag values of the response; and the *distributed lag* term — other explanatory variables known (or tested) to have effect on the response variable.

Autoregressive Distributed Lag (ADL) Model

- This popular model have been used on different field of discipline, from econometrics, epidemiology to agriculture.
- And to the best knowledge of the author, none has yet integrated the SGHMC to Bayesian ADL (BADL). Thus in this paper, the proposed model is abbreviated as BADL-SGHMC.

Autoregressive Distributed Lag (ADL) Model

- This popular model have been used on different field of discipline, from econometrics, epidemiology to agriculture.
- And to the best knowledge of the author, none has yet integrated the SGHMC to Bayesian ADL (BADL). Thus in this paper, the proposed model is abbreviated as BADL-SGHMC.

Autoregressive Distributed Lag (ADL) Model

The simplest ADL model is of order $p = 1$ and $q = 0$, denoted by ADL$(1, 0)$:

$$y(t) = w_0 + w_1 y(t-1) + w_2 x(t) + \varepsilon(t), \quad \varepsilon \sim \mathcal{N}(0, \sigma). \tag{23}$$

With other m explanatory variables:

$$y(t) = w_0 + w_1 y(t-1) + \sum_{i=1}^{m} w_{2+i} x_i(t) + \varepsilon(t), \quad \varepsilon \sim \mathcal{N}(0, \sigma). \tag{24}$$

Autoregressive Distributed Lag (ADL) Model

The simplest ADL model is of order $p = 1$ and $q = 0$, denoted by ADL$(1, 0)$:

$$y(t) = w_0 + w_1 y(t-1) + w_2 x(t) + \varepsilon(t), \quad \varepsilon \sim \mathcal{N}(0, \sigma). \tag{23}$$

With other m explanatory variables:

$$y(t) = w_0 + w_1 y(t-1) + \sum_{i=1}^{m} w_{2+i} x_i(t) + \varepsilon(t), \quad \varepsilon \sim \mathcal{N}(0, \sigma). \tag{24}$$

Autoregressive Distributed Lag (ADL) Model

ADL$(1,1)$:

$$y(t) = w_0 + w_1 y(t-1) + \sum_{i=1}^{m}\sum_{l=0}^{1} w_{(2+l\cdot m)+i} x_i(t-l) + \varepsilon(t). \quad (25)$$

For general ADL(p,q):

$$y(t) = w_0 + \sum_{k=1}^{p} w_k y(t-k) + \sum_{i=1}^{m}\sum_{l=0}^{q} w_{\kappa(p,l,m,i)} x_i(t-l) + \varepsilon(t). \quad (26)$$

where $\varepsilon \sim \mathcal{N}(0,\sigma)$.

Autoregressive Distributed Lag (ADL) Model

ADL$(1,1)$:

$$y(t) = w_0 + w_1 y(t-1) + \sum_{i=1}^{m}\sum_{l=0}^{1} w_{(2+l\cdot m)+i} x_i(t-l) + \varepsilon(t). \quad (25)$$

For general ADL(p,q):

$$y(t) = w_0 + \sum_{k=1}^{p} w_k y(t-k) + \sum_{i=1}^{m}\sum_{l=0}^{q} w_{\kappa(p,l,m,i)} x_i(t-l) + \varepsilon(t). \quad (26)$$

where $\varepsilon \sim \mathcal{N}(0,\sigma)$.

Autoregressive Distributed Lag (ADL) Model

And since the error term is centered on 0, then

$$\mathbb{E}[y(t)] = w_0 + \sum_{k=1}^{p} w_k y(t-k) + \sum_{i=1}^{m}\sum_{l=0}^{q} w_{\kappa(p,l,m,i)} x_i(t-l). \quad (27)$$

Objectives of the Study

General Objectives

1. derive the necessary theoretical results;
2. compare the performance of the proposed model, BADL-SGHMC, for three cases:
   1. leapfrog step size $\gamma = .09$ for 1,000 iterations;
   2. leapfrog step size $\gamma = .009$ for 10,000 iterations;
   3. leapfrog step size $\gamma = .0009$ for 100,000 iterations.
3. apply the proposed model to forecasting Philippine's economic growth; and
4. create software packages for SGHMC for R and Julia programming languages.

Objectives of the Study

General Objectives

1. derive the necessary theoretical results;
2. compare the performance of the proposed model, BADL-SGHMC, for three cases:
    1. leapfrog step size $\gamma = .09$ for 1,000 iterations;
    2. leapfrog step size $\gamma = .009$ for 10,000 iterations;
    3. leapfrog step size $\gamma = .0009$ for 100,000 iterations.
3. apply the proposed model to forecasting Philippine's economic growth; and
4. create software packages for SGHMC for R and Julia programming languages.

Objectives of the Study

General Objectives

1. derive the necessary theoretical results;
2. compare the performance of the proposed model, BADL-SGHMC, for three cases:
    1. leapfrog step size $\gamma = .09$ for 1,000 iterations;
    2. leapfrog step size $\gamma = .009$ for 10,000 iterations;
    3. leapfrog step size $\gamma = .0009$ for 100,000 iterations.
3. apply the proposed model to forecasting Philippine's economic growth; and
4. create software packages for SGHMC for R and Julia programming languages.

Objectives of the Study

General Objectives

1. derive the necessary theoretical results;
2. compare the performance of the proposed model, BADL-SGHMC, for three cases:
    1. leapfrog step size $\gamma = .09$ for 1,000 iterations;
    2. leapfrog step size $\gamma = .009$ for 10,000 iterations;
    3. leapfrog step size $\gamma = .0009$ for 100,000 iterations.
3. apply the proposed model to forecasting Philippine's economic growth; and
4. create software packages for SGHMC for R and Julia programming languages.

Objectives of the Study

General Objectives

1. derive the necessary theoretical results;
2. compare the performance of the proposed model, BADL-SGHMC, for three cases:
    1. leapfrog step size $\gamma = .09$ for 1,000 iterations;
    2. leapfrog step size $\gamma = .009$ for 10,000 iterations;
    3. leapfrog step size $\gamma = .0009$ for 100,000 iterations.
3. apply the proposed model to forecasting Philippine's economic growth; and
4. create software packages for SGHMC for R and Julia programming languages.

Objectives of the Study

General Objectives

1. derive the necessary theoretical results;
2. compare the performance of the proposed model, BADL-SGHMC, for three cases:
    1. leapfrog step size $\gamma = .09$ for 1,000 iterations;
    2. leapfrog step size $\gamma = .009$ for 10,000 iterations;
    3. leapfrog step size $\gamma = .0009$ for 100,000 iterations.
3. apply the proposed model to forecasting Philippine's economic growth; and
4. create software packages for SGHMC for R and Julia programming languages.

Objectives of the Study

General Objectives

1. derive the necessary theoretical results;
2. compare the performance of the proposed model, BADL-SGHMC, for three cases:
   1. leapfrog step size $\gamma = .09$ for 1,000 iterations;
   2. leapfrog step size $\gamma = .009$ for 10,000 iterations;
   3. leapfrog step size $\gamma = .0009$ for 100,000 iterations.
3. apply the proposed model to forecasting Philippine's economic growth; and
4. create software packages for SGHMC for R and Julia programming languages.

Objectives of the Study

Specific Objectives

1. The derivation of the theoretical results:
    1. given the *a priori* on the parameters of the BADL model, derive the posterior distribution; and
    2. given the *a posteriori* from the preceding objective, derive the stochastic gradient of the potential energy;

Objectives of the Study

Specific Objectives

1. The derivation of the theoretical results:
    1. given the *a priori* on the parameters of the BADL model, derive the posterior distribution; and
    2. given the *a posteriori* from the preceding objective, derive the stochastic gradient of the potential energy;

Objectives of the Study

Specific Objectives

1. The comparison on the performance of the proposed model, BADL-SGHMC, against the performance of the BADL-MH and BADL-HMC. This is done by considering four markov chains for each parameter of the BADL with dispersed random initial values from uniform distribution.

Objectives of the Study

Specific Objective

The following are the statistical methodologies used for assessing the performance of the models:

1. Heidelberger-Welch, for stationarity test on the Markov chains;
2. Gelman-Rubin, for convergence test of averages of the Markov chains;
3. Autocorrelations, for assessing the assumption of the independent and identically distributed (IID) samples from the *a posteriori*; and
4. Root Mean Squared Error (RMSE), for assessing the in-sample and out-of-sample forecast performance of the three models.

The above procedures are performed across three cases of the leapfrog step size mentioned in the General Objectives.

Objectives of the Study

Specific Objective

The following are the statistical methodologies used for assessing the performance of the models:

1. Heidelberger-Welch, for stationarity test on the Markov chains;
2. Gelman-Rubin, for convergence test of averages of the Markov chains;
3. Autocorrelations, for assessing the assumption of the independent and identically distributed (IID) samples from the *a posteriori*; and
4. Root Mean Squared Error (RMSE), for assessing the in-sample and out-of-sample forecast performance of the three models.

The above procedures are performed across three cases of the leapfrog step size mentioned in the General Objectives.

Objectives of the Study

Specific Objective

The following are the statistical methodologies used for assessing the performance of the models:

1. Heidelberger-Welch, for stationarity test on the Markov chains;
2. Gelman-Rubin, for convergence test of averages of the Markov chains;
3. Autocorrelations, for assessing the assumption of the independent and identically distributed (IID) samples from the *a posteriori*; and
4. Root Mean Squared Error (RMSE), for assessing the in-sample and out-of-sample forecast performance of the three models.

The above procedures are performed across three cases of the leapfrog step size mentioned in the General Objectives.

Objectives of the Study

Specific Objective

The following are the statistical methodologies used for assessing the performance of the models:

1. Heidelberger-Welch, for stationarity test on the Markov chains;
2. Gelman-Rubin, for convergence test of averages of the Markov chains;
3. Autocorrelations, for assessing the assumption of the independent and identically distributed (IID) samples from the *a posteriori*; and
4. Root Mean Squared Error (RMSE), for assessing the in-sample and out-of-sample forecast performance of the three models.

The above procedures are performed across three cases of the leapfrog step size mentioned in the General Objectives.

Objectives of the Study

Specific Objective

The following are the statistical methodologies used for assessing the performance of the models:

1. Heidelberger-Welch, for stationarity test on the Markov chains;
2. Gelman-Rubin, for convergence test of averages of the Markov chains;
3. Autocorrelations, for assessing the assumption of the independent and identically distributed (IID) samples from the *a posteriori*; and
4. Root Mean Squared Error (RMSE), for assessing the in-sample and out-of-sample forecast performance of the three models.

The above procedures are performed across three cases of the leapfrog step size mentioned in the General Objectives.

Objectives of the Study

Specific Objective

The following are the statistical methodologies used for assessing the performance of the models:

1. Heidelberger-Welch, for stationarity test on the Markov chains;
2. Gelman-Rubin, for convergence test of averages of the Markov chains;
3. Autocorrelations, for assessing the assumption of the independent and identically distributed (IID) samples from the *a posteriori*; and
4. Root Mean Squared Error (RMSE), for assessing the in-sample and out-of-sample forecast performance of the three models.

The above procedures are performed across three cases of the leapfrog step size mentioned in the General Objectives.

Objectives of the Study

Specific Objective

1. Apply the derived theoretical results to forecasting Philippine's year-over-year economic growth rate. In particular, the comparison of the models detailed in the preceding objective is performed using this data. The following are the time series involved in modeling:
    1. The Response Variable
        - Growth Rate of Gross Domestic Product
    2. The Predictors
        - Growth Rate of Peso/US Dollar Exchange Rate;
        - Growth Rate of Stock Price Index; and
        - Growth Rate of Gross International Reserves.

Objectives of the Study

Specific Objective

1. Apply the derived theoretical results to forecasting Philippine's year-over-year economic growth rate. In particular, the comparison of the models detailed in the preceding objective is performed using this data. The following are the time series involved in modeling:
   1. The Response Variable
      - Growth Rate of Gross Domestic Product
   2. The Predictors
      - Growth Rate of Peso/US Dollar Exchange Rate;
      - Growth Rate of Stock Price Index; and
      - Growth Rate of Gross International Reserves.

Objectives of the Study

Specific Objective

1. Apply the derived theoretical results to forecasting Philippine's year-over-year economic growth rate. In particular, the comparison of the models detailed in the preceding objective is performed using this data. The following are the time series involved in modeling:
    1. The Response Variable
        - Growth Rate of Gross Domestic Product
    2. The Predictors
        - Growth Rate of Peso/US Dollar Exchange Rate;
        - Growth Rate of Stock Price Index; and
        - Growth Rate of Gross International Reserves.

Objectives of the Study

Specific Objective

1. Apply the derived theoretical results to forecasting Philippine's year-over-year economic growth rate. In particular, the comparison of the models detailed in the preceding objective is performed using this data. The following are the time series involved in modeling:
    1. The Response Variable
        - Growth Rate of Gross Domestic Product
    2. The Predictors
        - Growth Rate of Peso/US Dollar Exchange Rate;
        - Growth Rate of Stock Price Index; and
        - Growth Rate of Gross International Reserves.

Objectives of the Study

Specific Objective

1. Create software packages for SGHMC for R and Julia programming languages using Github.com as the repository. That is, the package can be installed from this website.

Autoregressive Distributed Lag (ADL) Model

BADL(1, 1)-SGHMC:

$$y(t) = w_0 + w_1 y(t-1) + \sum_{i=1}^{m}\sum_{l=0}^{1} w_{(2+l\cdot m)+i}x_i(t-l) + \varepsilon(t). \quad (28)$$

- **w** is treated as random vector.
- **w** is estimated using Bayesian MCMC, specifically the SGHMC.

Initial Results

Proposition

Let $\mathscr{D} = \{[\mathbf{x}(t), y(t)], \forall t \in \mathbb{Z}_{+}^{\tau}\}$ *be the data such that* $y(t)$ *is modeled by a Gaussian function with mean given in Equation (27) and constant variance* $\alpha^{-1} \in \mathbb{R}_{+}$. *If* $\mathbf{w}$ *is the vector of coefficients of ADL(p, q) such that* $\mathbf{w} \sim \mathcal{N}_{d}(\mathbf{0}, \beta^{-1}\mathbf{I})$, *where* $\beta^{-1} \in \mathbb{R}_{+}$, *then the posterior is a multivariate Gaussian distribution with covariance matrix* $\boldsymbol{\Sigma} = (\alpha\mathfrak{G}^{T}\mathfrak{G} + \beta\mathbf{I})^{-1}$ *and mean vector* $\boldsymbol{\mu} = \alpha\boldsymbol{\Sigma}\mathfrak{G}^{T}\mathbf{y}$, *where* $\mathfrak{G}$ *is the design matrix.*

Initial Results

proof: Let $\mathbf{w} \triangleq [w_0\ w_1\ \cdots\ w_{\kappa(p,q,m,m)}]^\mathsf{T}$, $\kappa(p,l,m,m) \triangleq [(p+1)+l\cdot m]+m$ and let $\mathbf{z}(t) \triangleq [1\ y(t-1)\ \cdots\ x_m(t-q)]^\mathsf{T}$, then the ADL$(p,q)$ can be written as

$$y(t) = \mathbf{w}^\mathsf{T}\mathbf{z}(t) + \varepsilon(t), \quad \varepsilon(t) \sim \mathcal{N}(0, \alpha^{-1}). \tag{29}$$

The likelihood is therefore given by

$$\mathcal{L}(\mathbf{w}|\mathbf{y}) \triangleq \left(\frac{\alpha}{2\pi}\right)^{\tau/2} \exp\left\{-\sum_{t=1}^{\tau} \frac{\alpha[y(t)-\mathbf{w}^\mathsf{T}\mathbf{z}(t)]^2}{2}\right\}. \tag{30}$$

Let $\mathbf{y} \triangleq [y(1)\ y(2)\ \cdots\ y(\tau)]^\mathsf{T}$ and let $\mathfrak{G} \triangleq [(\mathbf{z}(t)^\mathsf{T})]$, i.e. $\mathfrak{G} \in \mathbb{R}^\tau \times \mathbb{R}^d$. Thus in matrix form

$$\mathcal{L}(\mathbf{w}|\mathbf{y}) \propto \exp\left[-\frac{\alpha}{2}(\mathbf{y}-\mathfrak{G}\mathbf{w})^\mathsf{T}(\mathbf{y}-\mathfrak{G}\mathbf{w})\right]. \tag{31}$$

Initial Results

The prior is given by

$$\mathbb{P}(\mathbf{w}) = \frac{1}{\sqrt{(2\pi)^d|\beta^{-1}\mathbf{I}|}} \exp\left[-\frac{1}{2}\mathbf{w}^{\mathsf{T}}\beta\mathbf{I}\mathbf{w}\right]. \tag{32}$$

So that the posterior would be

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) \propto \exp\left[-\frac{\alpha}{2}(\mathbf{y}-\mathfrak{G}\mathbf{w})^{\mathsf{T}}(\mathbf{y}-\mathfrak{G}\mathbf{w})\right]\exp\left[-\frac{1}{2}\mathbf{w}^{\mathsf{T}}\beta\mathbf{I}\mathbf{w}\right] \tag{33}$$

$$= \exp\left\{-\frac{1}{2}\left[\alpha(\mathbf{y}-\mathfrak{G}\mathbf{w})^{\mathsf{T}}(\mathbf{y}-\mathfrak{G}\mathbf{w}) + \mathbf{w}^{\mathsf{T}}\beta\mathbf{I}\mathbf{w}\right]\right\}. \tag{34}$$

Expanding the terms in the exponential factor becomes

$$\alpha\mathbf{y}^{\mathsf{T}}\mathbf{y} - 2\alpha\mathbf{w}^{\mathsf{T}}\mathfrak{G}^{\mathsf{T}}\mathbf{y} + \mathbf{w}^{\mathsf{T}}(\alpha\mathfrak{G}^{\mathsf{T}}\mathfrak{G} + \beta\mathbf{I})\mathbf{w}. \tag{35}$$

Initial Results

Hence

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) \propto \mathcal{C} \exp\left\{-\frac{1}{2}\left[\mathbf{w}^{\mathsf{T}}(\alpha\mathfrak{G}^{\mathsf{T}}\mathfrak{G} + \beta\mathbf{I})\mathbf{w} - 2\alpha\mathbf{w}^{\mathsf{T}}\mathfrak{G}^{\mathsf{T}}\mathbf{y}\right]\right\}. \quad (36)$$

Notice the terms in the exponential factor is of the form $ax^2 - 2bx$. This suggest a quadratic equation and therefore can be factored by completing the square. To do so, let $\mathbf{D} \triangleq \alpha\mathfrak{G}^{\mathsf{T}}\mathfrak{G} + \beta\mathbf{I}$ and $\mathbf{b} \triangleq \alpha\mathfrak{G}^{\mathsf{T}}\mathbf{y}$, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) \propto \mathcal{C} \exp\left\{-\frac{1}{2}\left[\mathbf{w}^{\mathsf{T}}\mathbf{D}\mathbf{w} - 2\mathbf{w}^{\mathsf{T}}\mathbf{b}\right]\right\} \quad (37)$$

$$= \mathcal{C} \exp\left\{-\frac{1}{2}\left[\mathbf{w}^{\mathsf{T}}\mathbf{D}\mathbf{w} - \mathbf{w}^{\mathsf{T}}\mathbf{b} - \mathbf{b}^{\mathsf{T}}\mathbf{w}\right]\right\}. \quad (38)$$

Initial Results

Next is to add a term that is not a function of $\mathbf{w}$ which can be assumed to be part of the constant $\mathcal{C}$. Let this term be $\mathbf{b}^{\mathsf{T}}\mathbf{D}^{-1}\mathbf{b}$, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) \propto \mathcal{C} \exp\left\{-\frac{1}{2}\left[\mathbf{w}^{\mathsf{T}}\mathbf{D}\mathbf{w} - \mathbf{w}^{\mathsf{T}}\mathbf{b} - \mathbf{b}^{\mathsf{T}}\mathbf{w} + \mathbf{b}^{\mathsf{T}}\mathbf{D}^{-1}\mathbf{b}\right]\right\}. \quad (39)$$

In order to proceed, the matrix $\mathbf{D}$ must be symmetric and invertible since later this will be the covariance matrix of the posterior which requires such property. If satisfied, then $\mathbf{I} = \mathbf{D}\mathbf{D}^{-1} = \mathbf{D}^{-1}\mathbf{D}$. So that

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) \propto \mathcal{C} \exp\left\{-\frac{1}{2}\left[\mathbf{w}^{\mathsf{T}}\mathbf{D}\mathbf{w} - \mathbf{w}^{\mathsf{T}}\mathbf{D}\mathbf{D}^{-1}\mathbf{b} - \mathbf{b}^{\mathsf{T}}\mathbf{D}^{-1}\mathbf{D}\mathbf{w} + \mathbf{b}^{\mathsf{T}}\mathbf{D}^{-1}\mathbf{D}\mathbf{D}^{-1}\mathbf{b}\right]\right\}.$$

Initial Results

Finally, let $\mathbf{\Sigma} \triangleq \mathbf{D}^{-1}$ and $\boldsymbol{\mu} \triangleq \mathbf{D}^{-1}\mathbf{b}$, then

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) \propto \mathcal{C} \exp\left\{-\frac{1}{2}\left[\mathbf{w}^{\mathsf{T}}\mathbf{\Sigma}^{-1}\mathbf{w} - \mathbf{w}^{\mathsf{T}}\mathbf{\Sigma}^{-1}\boldsymbol{\mu} - \boldsymbol{\mu}^{\mathsf{T}}\mathbf{\Sigma}^{-1}\mathbf{w} + \boldsymbol{\mu}^{\mathsf{T}}\mathbf{\Sigma}^{-1}\boldsymbol{\mu}\right]\right\} \tag{40}$$

$$= \mathcal{C} \exp\left\{-\frac{1}{2}\left[(\mathbf{w} - \boldsymbol{\mu})^{\mathsf{T}}\mathbf{\Sigma}^{-1}(\mathbf{w} - \boldsymbol{\mu})\right]\right\}. \tag{41}$$

Thus $\mathcal{C} = \frac{\mathcal{C}_0}{\mathbb{P}(\mathbf{y})}$, where $\mathcal{C}_0$ is the constant of the Guassian kernel in Equation (41). Therefore,

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \mathcal{N}_d(\mathbf{w}|\boldsymbol{\mu}, \mathbf{\Sigma}), \tag{42}$$

where $\mathbf{\Sigma} = (\alpha\mathfrak{G}^{\mathsf{T}}\mathfrak{G} + \beta\mathbf{I})^{-1}$ and $\boldsymbol{\mu} = \alpha\mathbf{\Sigma}\mathfrak{G}^{\mathsf{T}}\mathbf{y}$.

Initial Results

Thus $\mathcal{C} = \frac{\mathcal{C}_0}{\mathbb{P}(\mathbf{y})}$, where $\mathcal{C}_0$ is the constant of the Guassian kernel in Equation (41). Therefore,

$$\mathbb{P}(\mathbf{w}|\mathbf{y}) = \mathcal{N}_d(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}), \tag{43}$$

where $\boldsymbol{\Sigma} = (\alpha\boldsymbol{\mathfrak{G}}^{\mathsf{T}}\boldsymbol{\mathfrak{G}} + \beta\mathbf{I})^{-1}$ and $\boldsymbol{\mu} = \alpha\boldsymbol{\Sigma}\boldsymbol{\mathfrak{G}}^{\mathsf{T}}\mathbf{y}$.

Initial Results

Proposition

Let the posterior of the parameters be $\mathbb{P}(\mathbf{w}|\mathbf{y})$ *given in Proposition 3.1, with* $\mathbf{y} = [y(1)\ y(2)\ \cdots\ y(\tau)]^{\mathrm{T}}$*. Further, let* $\mathbf{w} \sim \mathcal{N}_d(\mathbf{0}, \beta^{-1}\mathbf{I})$*, then the gradient noise of* $-\log \mathbb{P}(\mathbf{w}|\mathbf{y})$*, needed for SGHMC's computation is given below:*

$$-\alpha \sum_{t=1}^{\tau}(y(t) - \mathbf{w}^T\mathbf{z}(t))\mathbf{z}(t) + \beta\mathbf{w} + \boldsymbol{\xi}, \quad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathfrak{A}(\mathbf{w})). \quad (44)$$

Initial Results

proof: Again, let $\mathbf{w} \triangleq [w_0 \;\; w_1 \;\; \cdots \;\; w_{\kappa(p,q,m,m)}]^\mathsf{T}, \kappa(p,l,m,m) \triangleq [(p+1)+l \cdot m]+m$ and let $\mathbf{z}(t) \triangleq [1 \;\; y(t-1) \;\; \cdots \;\; x_m(t-q)]^\mathsf{T}$, then

$$\frac{\mathrm{d}}{\mathrm{d}\,\mathbf{w}}[-\log \mathbb{P}(\mathbf{w}|\mathbf{y})] = -\frac{\mathrm{d}}{\mathrm{d}\,\mathbf{w}}[\ell(\mathbf{w}|\mathbf{y}) + \log \mathbb{P}(\mathbf{w}) - \log \mathbb{P}(\mathbf{y})] \tag{45}$$

$$= -\left[\frac{\mathrm{d}}{\mathrm{d}\,\mathbf{w}}\ell(\mathbf{w}|\mathbf{y}) + \frac{\mathrm{d}}{\mathrm{d}\,\mathbf{w}}\log \mathbb{P}(\mathbf{w})\right] \tag{46}$$

so that

$$\frac{\mathrm{d}}{\mathrm{d}\,\mathbf{w}}\ell(\mathbf{w}|\mathbf{y}) = \frac{\mathrm{d}}{\mathrm{d}\,\mathbf{w}}\log\left\{\left(\frac{\alpha}{2\pi}\right)^{\tau/2}\exp\left[-\sum_{t=1}^{\tau}\frac{\alpha(y(t)-\mathbf{w}^\mathsf{T}\mathbf{z}(t))^2}{2}\right]\right\} \tag{47}$$

$$= -\frac{\mathrm{d}}{\mathrm{d}\,\mathbf{w}}\sum_{t=1}^{\tau}\frac{\alpha(y(t)-\mathbf{w}^\mathsf{T}\mathbf{z}(t))^2}{2} = \alpha\sum_{t=1}^{\tau}(y(t)-\mathbf{w}^\mathsf{T}\mathbf{z}(t))\mathbf{z}(t) \tag{48}$$

Initial Results

and the derivative of the prior with log transformation is given by

$$\frac{\mathrm{d}}{\mathrm{d}\,\mathbf{w}} \log \mathbb{P}(\mathbf{w}) = -\frac{\beta}{2}\frac{\mathrm{d}}{\mathrm{d}\,\mathbf{w}}\mathbf{w}^{\mathsf{T}}\mathbf{w} = -\beta\mathbf{w}. \tag{49}$$

Equation (44) then follows from Equation (19).