Convexization in Markov Chain Monte Carlo

Dimitri Kanevsky[1], Avishy Carmi[2]

[1]IBM T. J. Watson
Yorktown Heights, NY

[2]Department of Aerospace Engineering
Technion, Israel

August 23, 2011

Problem Statement

- MCMC processes in general are governed by non convex objective functions that are difficult to optimize.
- Standard regularization of MCMC processes (e.g with quadratic penalties) in general improve optimization performance accuracy but slow the optimization process significantly.
- There are various efficient methods in general to optimize convex functions. It is natural in optimization of non-convex functions to use convex lower bound functions in intermediate steps.
- How can we incorporate into MCMC methods from convex optimization theory?

Summary

- The goal of the paper is to introduce a general convexization process for arbitrary functions to assist Markov Chain Monte Carlo (MCMC) optimization.
- We describe how concave low bound (auxiliary) functions are used as intermediate steps in optimization of general functions
- In the paper a recently introduced technique how to build auxiliary functions is described
- We give examples of concave auxiliary functions for convex functions
- We apply a theory of auxiliary functions to stochastic optimization
- We integrate in a Metropolis-Hastings method auxiliary functions
- We illustrate our variant of Metropolis-Hastings method with numerical experiments by solving sparse optimization problems.

Definition of auxiliary functions

Let $f(x) : \mathcal{U} \subset \mathbb{R}^n \to \mathbb{R}$ be a real valued differentiable function in an open subset $\mathcal{U}$. Let $\mathbf{Q}_f = \mathbf{Q}_f(x,y) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ be twice differentiable in $x \in \mathcal{U}$ for each $y \in \mathcal{U}$. We define $\mathbf{Q}_f$ as an auxiliary function for f in $\mathcal{U}$ if the following properties hold.

1. $\mathbf{Q}_f(x,y)$ is a strictly concave function of x for any $y \in \mathcal{U}$ with its (unique) maximum point belonging to $\mathcal{U}$ (recall that twice differentiable function is strictly concave or convex over some domain if its Hessian function is positive or negative definite in the domain, respectively).
2. Hyperplanes tangent to manifolds defined by $z = g_y(x) = \mathbf{Q}_f(x,y)$ and $z = f(x)$ at any $x = y \in \mathcal{U}$ are parallel to each other, i.e.

$$\nabla_x \mathbf{Q}_f(x,y)|_{x=y} = \nabla_x f(x) \tag{1}$$

3. For any $x \in \mathcal{U}$ $f(x) = \mathbf{Q}_f(x,x)$
4. For any $x, y \in \mathcal{U}$ $f(x) \geqslant \mathbf{Q}_f(x,y)$

Optimization process for auxiliary functions

In an optimization process via an $\mathbf{Q}$-function it is usually assumed that finding an optimum of an $\mathbf{Q}$-function is "easier" than finding a (local) optimum of the original function f. Naturally, a desired outcome is for the equation $\nabla_x \mathbf{Q}_f(x, y) = 0$ to have a closed form solution.

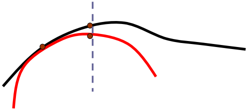
The optimization recursion via an auxiliary function can be described as follows (where we use EM style).

E-step Given x^t construct $\mathbf{Q}_f(x, x^t)$

M-step Find

$$x^{t+1} = \arg\max_{x \in \mathcal{U}} \mathbf{Q}_f(x, x^t) \tag{2}$$

For updates (2) we have
$f(x^{t+1}) = \mathbf{Q}_f(x^{t+1}, x^{t+1}) \geqslant \mathbf{Q}_f(x^t, x^{t+1}) \geqslant \mathbf{Q}_f(x^t, x^t) = f(x^t)$.
This means that iterative update rules have a "growth" property (i.e. the value of the original function increases for the new parameters values).



Auxiliary function:
In this figure the upper curve denotes the plot of the objective function $f : x \longrightarrow \mathbb{R}$ and the curve in red, i.e. the concave lower curve, represents the $\mathcal{A}$-function $\mathbf{Q}_f(., x_0) : x \longrightarrow \mathbb{R}$. As it be can seen from this figure, for some x_1 that maximizes $\mathbf{Q}_f(x, x_0)$ we have $f(x_1) > f(x_0)$.

Convergent statement

- Let call a point $x \in \mathcal{U}$ critical if $\nabla_x f(x) = 0$.
- In the paper we prove the following convergence statement
  Proposition
  Let $\mathbf{Q}_f$ be an auxiliary function for f in $\mathcal{U}$ and let $\mathcal{S} = \{x^t, t = 1, 2, ...\}$. Then all limit points of $\mathcal{S}$ that lie in $\mathcal{U}$ are critical points. Assume in addition that f has a local maximum at some limit point of the sequence $\mathcal{S}$ in $\mathcal{U}$ and that f is strictly concave in some open neighborhood of this point. Then there exists only one critical point of $\mathcal{S}$ in $\mathcal{U}$
- This means that iterative application of update rules via an auxiliary function converges to a critical point.

A general way to build auxiliary functions

- Assume that $f(x)$ is strictly concave in $\mathcal{U}$. Then for any point $x \in \mathcal{U}$ we can construct a family of auxiliary functions as follows.
- Let us consider the following family of functions.

$$\mathbf{Q}_f(y, x; \lambda) = -\lambda f\left(-\frac{y}{\lambda} + x\left(1 + \frac{1}{\lambda}\right)\right) + f(x) + \lambda f(x) \quad (3)$$

- These family functions (3) obey properties 1-3 for any $\lambda > 0$ in the definition of auxiliary function.
- In general, for an arbitrary function $f(x)$ one can construct auxiliary functions $\mathbf{Q}_f(y, x; \lambda)$ locally (with different λ in neighborhoods for different points x).

Three transformations to build auxiliary functions

The family of functions (3) are obtained via subsequent applications of the following three transformations.

Reflection along x-axis

$$\mathbf{H}_f(y, x) = -f(y) + 2f(x) \tag{4}$$

Reflection along y-axis

$$\mathbf{G}_f(y, x) = \mathbf{H}_f(-y + 2x, x) + 2\mathbf{H}_f(x, x) \tag{5}$$

Scaling

$$\mathbf{Q}_f(y, x; \lambda) = \lambda \mathbf{G}_f\left(\frac{y}{\lambda} + x\left(1 - \frac{1}{\lambda}\right), x\right) + (1-\lambda)\mathbf{G}_f(x, x) \tag{6}$$

Objective function that is sum of convex and concave functions

- Assume that
$$f(x) = g(x) + h(x) \tag{7}$$
where $h(x)$ is strictly convex in $\mathcal{U}$.
- Then we can define an auxiliary function for $f(x)$ as following
$$\mathbf{Q}_f(y, x) = \mathbf{Q}_g(y, x) + \mathbf{Q}_h(y, x, \lambda) \tag{8}$$
where $\mathbf{Q}_g(y, x)$ is some auxiliary function associated with g (for example it coincides with $g(x)$ if $g(x)$ is strictly concave).
- In practical applications some function $\mathbf{Q}_h(y, x, \lambda)$ may be concave but not strictly concave. In this case one can add a small regularized penalty to it to make it strictly concave.

Exponential families

- The important example of convex functions is an exponential family.
- We define an exponential family as any family of densities on $\mathbb{R}^D$, parameterized by θ, that can be written as $\xi(x,\theta) = \frac{\exp\{\theta^T \phi(x)\}}{Z(\theta)}$ where x is a D-dimensional base observation.
- The function $\phi : \mathbb{R}^D \rightarrow \mathbb{R}^d$ characterizes the exponential family. $Z(\theta) = \int_{\Xi} \exp\{\theta^T \phi(x)\} dx$ is the partition function, that provides the normalization necessary for $\xi(x,\theta)$. The function $\log \xi(x,\theta)$ is convex and it is strictly convex if $Var[\phi(x)] \neq 0$
- Some objective functions of exponential densities (e.g. in energy-based models) can be optimized via a recursion procedure that at each recursion require optimization of weighed sum of exponential densities, i.e., a sum of convex and concave functions.

Online gradient descent for stochastic functions

- Find some parameter vector $x \in \mathcal{U}$ such that sum of functions $f^i \to \mathbb{R}$ takes on the smallest value possible:
- $$f^*(x) = \frac{1}{T}\sum_{t=1}^{T} f^t(x) \tag{9}$$

  and

  $$x^* = \arg\min_{x \in \mathcal{U}} f^*(x) \tag{10}$$
- In the elementary online gradient descent algorithm instead of averaging the gradient of the function f^* over the complete training set each operation of the online gradient descent consists of choosing a function f^t at random (as corresponding to a random training example) and and updating the parameter x^t according to the formula

$$x^{t+1} = x^t - \gamma_t \nabla_x f^t(x^t t) \tag{11}$$

Auxiliary stochastic functions

- Assume now that functions $f^i(x)$ are non-concave and we need to solve the maximization problem

$$\max \sum f^i(x) \tag{12}$$

- Assume also that $\mathbf{Q}^i(y,x)$ are auxiliary functions for $f^i(y)$ at x. In this case one can consider the following optimization process.
  Let

$$\mathbf{Q}^*(y,x) = \sum \mathbf{Q}^i(y,x) \tag{13}$$

- Then $\mathbf{Q}^*(y,x)$ is an auxiliary function for $f^*(y) = \sum f^i(y)$. For $t = 1, 2, \dots$ we can optimize $\mathbf{Q}^*(x^t, y)$ using stochastic descent methods and find x^{t+1}. This induces the optimization process for $f^*(x)$ via the auxiliary function $\mathbf{Q}^*(y,x)$.

Metropolis-Hastings algorithm

- How to combine convexization process with some MCMC technique like Metropolis-Hastings
- We want to draw samples from a probability distribution $P(x)$ that is proportional to some complex (not convex) expression $f(x)$
- Assume that we have an ergodic and balanced Markov chain x^t that at sufficiently long times generates states that obey the $P(x)$ distribution.
- Let $\mathbf{Q}(x'; x^t)$ be proposal densities which depends on the current state x^t to generate a new proposed state x'.
- The new sample x' is "accepted" as the next value $x^{t+1} = x'$ if α is drawn from $U(0,1)$, the uniform distribution satisfies

$$\alpha < \frac{f(x')}{f(x^t)} \frac{\mathbf{Q}(x'; x^t)}{\mathbf{Q}(x^t; x')} \tag{14}$$

Metropolis-Hastings auxiliary algorithm

- We define proposals as auxiliary functions in the Metropolis-Hastings algorithm
- Let $\mathbf{Q}_f(x, y)$ be an auxiliary function for $f(x)$. Then we have:
    - Given the most recent sampled value x^t draw a new proposal state x' with the probability $\mathbf{Q}_f(x'; x^t)$
    - Calculate
    $$a = \frac{f(x')}{f(x^t)} \frac{\mathbf{Q}_f(x'; x^t;)}{\mathbf{Q}_f(x^t; x')} \tag{15}$$
    - The new state x^{t+1} is chosen according to the following rules:
    If $a \geqslant 1$ then $x^{t+1} = x'$
    else $x^{t+1} = x'$ with probability a and $x^{t+1} = x^t$ with probability $1 - a$

Example of optimization problem: compressive sensing

- A Bayesian representation of a compressive sensing problem:

$$\max_x \exp\left(\frac{-0.5||y - Hx||^2}{R}\right) * \exp\left(\frac{-0.5||x||_1^2}{\sigma^2}\right) \tag{16}$$

In this formula y is an m dimensional vector (measurement), H is an $m \times n$ sensing matrix with $m < n$, x is an n dimensional parameter vector, and the function $\exp\left(\frac{-0.5||x||_1^2}{\sigma^2}\right)$ is a "Semi-Gaussian" penalty to enforce the sparsity (here $||x||_1^2 := (\sum_i |x_i|)^2$ for all entries x_i in x).

Auxiliary function for Bayesian representation

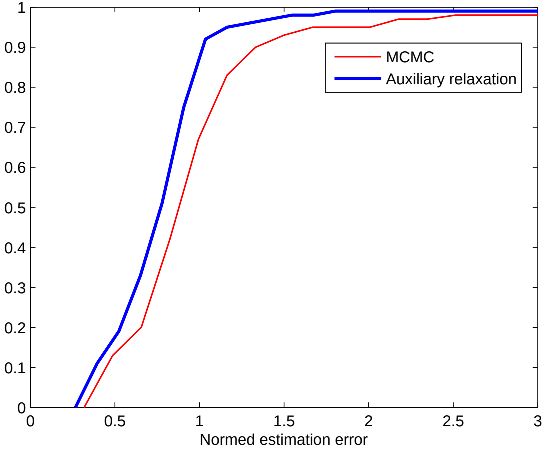
- Auxiliary function for (16) for sufficiently small λ:

$$\mathbf{Q}(x, x_0) = \lambda \exp\left(\frac{-0.5(sign(x_0) * \left(\frac{x}{\lambda} + (1 - \frac{1}{\lambda})x_0\right)^2}{\sigma^2}\right) \quad (17)$$

Numerical experiments - parameters

- We run simulation comparative experiments using the standard Metropolis-Hastings method (14) and the Metropolis-Hastings method (15) with the convex auxiliary function for the problem (16).
- In our simulation experiments parameters where chosen as the following:
    - $n = 10$, $m = 5$. Entries in the sensing matrix H were obtained by sampling according to $\mathcal{N}(0, 1/5)$.
    - the signal support vector $x \in \mathbb{R}^{10}$ is assumed to be a sparse parametric vector with signal support consisting of two elements.
- We had 100 runs to produce the cumulative distribution of errors. In each run we produced 10000 samples and had 5000 burn-in samples.

Cumulative Distribution of errors:



The ordinate axis is the probability and the absica is the normed estimation error.

Conclusions

- In this paper we introduced a novel convexization approach for MCMC that is based on general convexization techniques that allow to build auxilary functions for a wide class of problems.
- We illustrated this convexization method on a compressive sensing problem that was represented in a Bayesian form with a semi-gaussian penalty.
- Simulation experiments showed that Metroplis-Hastings method with axillary functions outperforms a standard Metroplis-Hastings method.
- We plan to test convexization methods on a broad class of MCMC based methods and develop a detailed methodology for a dynamic adjustment of scaling parameters for auxiliary functions in iterative MCMC processes.