Random thoughts on finding maximums

Prof. Jacob M. Montgomery

Statistical Computing

April 18, 2017

The problem

- We often want to find maxima (local or global).

The problem

- We often want to find maxima (local or global).
- Why?

The problem

- We often want to find maxima (local or global).
- Why?
- What is the analytical solution?

The problem

- We often want to find maxima (local or global).
- Why?
- What is the analytical solution?
- The problem arises when:

The problem

- We often want to find maxima (local or global).
- Why?
- What is the analytical solution?
- The problem arises when:
    - There is no analytical solution

The problem

- We often want to find maxima (local or global).
- Why?
- What is the analytical solution?
- The problem arises when:
  - There is no analytical solution
  - High dimensional problems

Optimization in general

- NOTE: Finding a global maxima can be particularly difficult.
- Assume we have some function $f : \mathcal{R} \rightarrow \mathcal{R}$
- Suppose that a global maximum value of f exists and is located at x^*
- Then:

Optimization in general

- NOTE: Finding a global maxima can be particularly difficult.
- Assume we have some function $f : \mathcal{R} \rightarrow \mathcal{R}$
- Suppose that a global maximum value of f exists and is located at x^*
- Then:
  - $f'(x^*) = 0$
  - $f''(x^*) < 0$
- Maximization methods work by generating a sequence of points such that $x(0), x(1), x(2), \ldots,$ converges to x^*

Optimization in general

- Start with some proposed solution $x(n)$
- Choose new solution (in the neighborhood) $x(n+1)$
- Stop using some combination of the following rules:
  - $|x(n) - x(n-1)| \leq \epsilon$
  - $|f(x(n)) - f(x(n-1))| \leq \epsilon$
  - $|f'(x(n))| \leq \epsilon$

Optimization in general

1. Choose a start value $x(0)$
2. Evaluate the function and/or the first derivative
3. *Choose some new value* $x(1)$
4. Evaluate the function and/or the first derivative at $x(1)$
5. Check the stop rule.
   - If met, terminate
   - If not met, return to (3)

Some notes

- The trick is in choosing the new proposals.

Some notes

- The trick is in choosing the new proposals.
- The method chosen can have HUGE effects on speed of convergence or whether it is ever reached

Some notes

- The trick is in choosing the new proposals.
- The method chosen can have HUGE effects on speed of convergence or whether it is ever reached
- This is actually a very exciting area of research.

Some notes

- The trick is in choosing the new proposals.
- The method chosen can have HUGE effects on speed of convergence or whether it is ever reached
- This is actually a very exciting area of research.
  - Machine learning

Some notes

- The trick is in choosing the new proposals.
- The method chosen can have HUGE effects on speed of convergence or whether it is ever reached
- This is actually a very exciting area of research.
    - Machine learning
    - Big data

Some notes

- The trick is in choosing the new proposals.
- The method chosen can have HUGE effects on speed of convergence or whether it is ever reached
- This is actually a very exciting area of research.
  - Machine learning
  - Big data

Some notes

- The trick is in choosing the new proposals.
- The method chosen can have HUGE effects on speed of convergence or whether it is ever reached
- This is actually a very exciting area of research.
  - Machine learning
  - Big data

Abstract. During the last decade, the data sizes have grown faster than the speed of processors. In this context, the capabilities of statistical machine learning methods is limited by the computing time rather than the sample size. A more precise analysis uncovers qualitatively different tradeoffs for the case of small-scale and large-scale learning problems. The large-scale case involves the computational complexity of the underlying optimization algorithm in non-trivial ways. Unlikely optimization algorithms such as stochastic gradient descent show amazing performance for large-scale problems. In particular, second order stochastic gradient and averaged stochastic gradient are asymptotically efficient after a single pass on the training set.

Why is this so hard?

- Start with some proposed solution $x(n)$
- Choose new solution (in the neighborhood) $x(n+1)$
- Stop using some combination of the following rules:
  - $|x(n) - x(n-1)| \leq \epsilon$
  - $|f(x(n)) - f(x(n-1))| \leq \epsilon$
  - $|f'(x(n))| \leq \epsilon$

Golden section

- If we have $f(l) \leq f(m)$ and $f(m) \leq f(r)$, then there must be a local maximum on the interval $[a, b]$.
- If the difference between $f(l)$ and $f(r)$ is very small, then m must be the (near enough) the local maximum.
- If not, then we move either $x(l)$ or $x(r)$ towards $x(m)$ and then move $x(m)$ accordingly.
- Repeat until they are "close enough"

Golden section

Start with $x_l < x_m < x_r$ s.t. $f(x_l) \leq f(x_m)$ and $f(x) \leq f(r)$

1. If $x_r - x_l \leq \epsilon$ stop
2. If $x_r - x_m > x_m - x_l$ then do (3) otherwise do (4)
3. Let $y = x_m + (x_r - x_m)/(1+\rho)$ if $f(y) \geq f(x_m)$ then put $x_l = x_m$ and $x_m = y$ otheriwse put $x_r = y$
4. Let $y = x_m - (x_m - x_l)/(1+\rho)$ if $f(y) \geq f(x_m)$ then put $x_r = x_m$ and $x_m = y$ otherwise put $x_l = y$
5. Return to (1)

Here $\rho = \frac{1+\sqrt{5}}{2}$ often called the golden ratio.

Golden section

Start with $x_l < x_m < x_r$ s.t. $f(x_l) \leq f(x_m)$ and $f(x) \leq f(r)$

1. If $x_r - x_l \leq \epsilon$ stop
2. If $x_r - x_m > x_m - x_l$ then do (3) otherwise do (4)
3. Let $y = x_m + (x_r - x_m)/(1+\rho)$ if $f(y) \geq f(x_m)$ then put $x_l = x_m$ and $x_m = y$ otheriwse put $x_r = y$
4. Let $y = x_m - (x_m - x_l)/(1+\rho)$ if $f(y) \geq f(x_m)$ then put $x_r = x_m$ and $x_m = y$ otherwise put $x_l = y$
5. Return to (1)

Here $\rho = \frac{1+\sqrt{5}}{2}$ often called the golden ratio.
Class exercise: Do this $dgamma()$, where you choose the parameter values. Plotting the thing may help.

Maximization in R

- *optimize*: A multi-dimensional extension for the golden-section
- *optim*: *Nelder-Mead*, *quasi-Newton*, and *conjugate gradient*

Use both methods to find local maxima for

$$sin(x^2/2 - y^2/4) \times cos(2x - \exp(y))$$

over the interval $x \in [-1, 3]$ and $y \in [-1, 3]$. Play around with it. Try different starting values etc.

EM: Ensemble Bayesian Model Averaging

- We are interesting in prediction $\mathbf{y}^{t^*}$
- We out-of-sample forecasts for events $\mathbf{y}^t$ in generated from K forecasting models or teams, $M_1, M_2, \ldots, M_K$.
- The predictive PDF for the quantity of interest is $p(\mathbf{y}^{t^*}|M_k)$
- The conditional probability for each model is
  $p(M_k|\mathbf{y}^t) = p(\mathbf{y}^t|M_k)\pi(M_k) / \sum_{k=1}^{K} p(\mathbf{y}^t|M_k)\pi(M_k)$
- The marginal predictive PDF is $p(\mathbf{y}^{t^*}) = \sum_{k=1}^{K} p(\mathbf{y}^{t^*}|M_k)p(M_k|\mathbf{y}^t)$.

Example: Ensemble Bayesian Model Averaging

- Denote $w_k = p(M_k|\mathbf{y}^t)$
- Let $p(\mathbf{y}^{t^*}|M_k) = N(f_k^{t^*}, \sigma^2)$

$$p(y|f_1^{s|t^*}, \ldots, f_K^{s|t^*}) = \sum_{k=1}^{K} w_k N(f_k^{t^*}, \sigma^2). \tag{1}$$

$$\mathcal{L}(\mathbf{w}, \sigma^2) = \sum_t \log\left(\sum_{k=1}^{K} w_k N(f_k^t, \sigma^2)\right), \tag{2}$$

E-M Algorithm

$$\hat{z}_k^{(j+1)t} = \frac{\hat{w}_k^{(j)} p^{(j)}(y|f_k^t)}{\sum_{k=1}^{K} \hat{w}_k^{(j)} p^{(j)}(y|f_k^t)}, \tag{3}$$

$$\hat{w}_k^{(j+1)} = \frac{1}{n} \sum_t \hat{z}_k^{(j+1)t}, \tag{4}$$