Optimization in Medical Research

Nonlinear Optimization Project 2

Shuo Yang, Chuang Miao, Jie Liu, Timothy Thomas

4/11/2013

1. Introduction

In this part, a brief overview of steepest descent (SD) method is presented. In a search direction method (general iterations $x_{k+1} = x_k + \alpha_k p_k$), the algorithm first chooses a search direction p and then performs a line search to find a step size α. In the steepest descent method, the search direction $p = -\nabla f(x)$ is the negative gradient at each iteration point, and along this direction the objective function f decreases most rapidly. That is why it is called the steepest descent method [1, 2]. Provided a well selected line length, this method is guaranteed to produce a decrease in f. In the following, we want to look at this method in detail for its practical implementations.

The first question is how to determine the line length in each iteration, i.e. how to choose the step size α_k. A straightforward idea is to seek the global minimizer of the function $\min_\alpha f(x + \alpha p)$, but this is usually computationally expensive [3]. Alternatively, α can be chosen empirically and fixed as a small constant to avoid solving the minimization problem, which is applied commonly, as shown in our applications. We can also use an inexact line search to find a step size close to the optimal one when a set of conditions are satisfied [3]. Such line search is often done in two stages: first a bracketing phase finds an interval containing suitable step lengths, then a refinement phase which computes a good step length within this interval. The feasible interval can be determined by the Wolfe Conditions:
Define $\phi(\alpha) = f(x + \alpha p)$, then $\phi(\alpha_k) = f(x_{k+1})$ and $\phi(0) = f(x_k)$.
The Wolfe conditions require each step size α_k to satisfy:
$\phi(\alpha) \leq \phi(0) + c_1 \alpha \phi'(0)$ and
$\phi'(\alpha) \geq c_2 \phi'(0)$ with $0 < c_1 < c_2 < 1$.
The first condition gives a sufficient reduction in f following each iteration but has no control in unacceptably short length, while the second condition bounds the lower limit. We can calculate the range of α to satisfy the Wolfe conditions but practically we just need to use a search method to find some α values that satisfy them. The "backtracking approach" [3] is normally used.

The second question we are concerned with is whether the SD method is guaranteed to converge. This turns out to require us to find some conditions for convergence[3]. A function is said to be "Lipschitz continuous" if there exists a constant L, such that $|f(x) - f(y)| \leq L||x - y||$, for all x, y $\in$ an open set $\mathbb{N}$ (i.e. the derivative of f is bounded between –L and +L). Then based on the "Zoutendijk theorem", in any iteration, if the objective function is bounded and its gradient $g = -\nabla f$ is Lipschitz continuous, and each iteration follows a descent direction p_k and has a line length α_k which satisfies the Wolfe conditions, then the iteration converges. In particular, if f is C^2 and its Hessian is bounded in norm (its maximum eigenvalue is less than a constant), the SD method is guaranteed to produce a sequence of gradients which converge to zero, provided the line search satisfies the Wolfe conditions. Besides, the convergence rate of SD is linear and the

iteration follows a “zig-zag” pattern (as shown in Figure1). This can be seen when we solve for the exact value of step size: $\frac{d}{d\alpha} f(x_k - \alpha g_k) = 0$. Using the chain rule, we have $g_k^T g(x_k - \alpha g_k) = g_k^T g_{k+1} = 0$, so that successive search directions are perpendicular. This causes the “zig-zag” characteristic of the SD method. For an inexact line search, usually the step size is close to the exact optimal one, so often $g_k^T g_{k+1} \cong 0$.

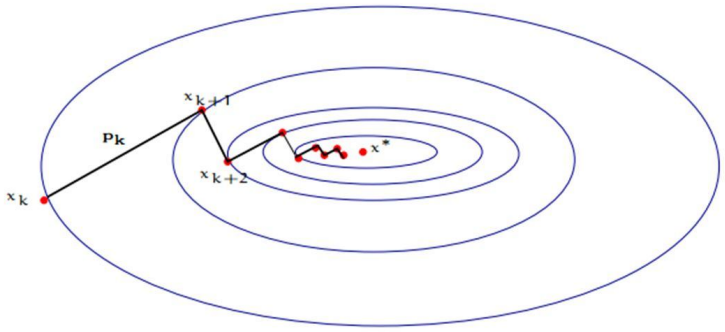

Figure1. Typical steepest descent steps

2. Optimization in Computational Health Informatics

2.1. Background

Computational health informatics is a branch of Health Informatics. The health domain provides an extremely wide variety of problems that can be tackled using computational techniques, and computer scientists are attempting to discover these medical knowledge underlying the large amounts of clinical data by studying the principles of computer science that will allow for meaningful (to medicine) algorithms and systems to be developed[4]. Researchers study and manage information, study behavior related to decisions, and develop computational methods and use them to generate knowledge by machine learning and statistical techniques.

Using computers to analyze health data has been around since the 1950s, but it wasn't until the 1990s that the first sturdy models appeared[4]. Computer models are used to examine various topics such as how exercise affects obesity, healthcare costs, and many more.

There are considerable number of uncertainties in medical knowledge, such as ‘How influential are different preconditions in making the event happen?’ or ‘How does the chance of true occurrence change over time?’. Bayesian Net is one of the most popular techniques because it can answer such queries and capture the probability of these uncertainties based on the observations.

2.2. Mathematical Model

Bayesian networks can represent essentially any full joint probability distribution and in many cases can do so very concisely [5]. A Bayesian network is a directed graph in which each node is annotated with quantitative probability information. The full specification is as follows:

1. Each node corresponds to a random variable, which may be discrete or continuous.

2. A set of directed links or arrows connects pairs of nodes. If there is an arrow from node X to node Y, X is said to be a parent of Y. The graph has no directed cycles (and hence is a directed acyclic graph, or DAG.
3. Each node x_i has a conditional probability distribution $P(X_i \mid Parents(X_i))$ that quantifies the effect of the parents on the node.

It is usually easy for a domain expert to decide what direct influences exist in the domain. Once the topology of the Bayesian network is laid out, we need only specify a conditional probability distribution for each variable, given its parents. We will see that the combination of the topology and the conditional distributions suffices to specify (implicitly) the full joint distribution for all the variables.

We now formally introduce some basic notation that will be useful for the paper. In a Bayesian network with n discrete valued nodes, we denote the parameters by $\boldsymbol{\theta_{ijk}}$ ($\boldsymbol{i} \in \{\mathbf{1}, \mathbf{2}, \dots, \boldsymbol{n}\}, \boldsymbol{j} \in \{\mathbf{1}, \mathbf{2}, \dots, \boldsymbol{v_i}\}, \boldsymbol{k} \in \{\mathbf{1}, \mathbf{2}, \dots, \boldsymbol{r_i}\}$) which denotes the conditional probability of X_i to be its k-th value given the j-th configuration of its parents (i.e. $\boldsymbol{P}(\boldsymbol{X_i^k} | \boldsymbol{Pa_i^j})$). r_i denotes the number of states of the discrete variable X_i; Pa_i represents the set of node X_i's parents; the number of configurations of Pa_i is $\boldsymbol{v_i} = \prod_{\boldsymbol{X_t} \in \boldsymbol{PA_i}} \boldsymbol{r_t}$; j is the index of a particular configurationPa_i^j.

Given the structure of a Bayesian network, the optimized conditional probability distributions that best represent the training data are learned by maximize the likelihood function of the training data given the proposed model, which equals to solve the following optimization problem:

$$\widehat{\boldsymbol{\theta}} = \boldsymbol{argmax}_{\boldsymbol{\theta} \in \boldsymbol{\Theta}^{\boldsymbol{Q}}} \prod_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{n}} \prod_{\boldsymbol{j}=\mathbf{1}}^{\boldsymbol{v_i}} \prod_{\boldsymbol{k}=\mathbf{1}}^{\boldsymbol{r_i}} \boldsymbol{\theta}_{\boldsymbol{ijk}}^{\boldsymbol{N_{ijk}}} \quad (1)$$

Where $\boldsymbol{N_{ijk}}$ represent the number of samples that satisfy the corresponding configuration of (X_i^k, Pa_i^j). We thus have a large number of maximization tasks which are independent, aside from the constraint$\theta \in \Theta^Q$.

To eliminate the need for the simplex constraints (i.e., that $\forall i, j \sum_k \theta_{ijk} = 1$) with a reparameterization, define μ_{ijk} such that $\boldsymbol{\theta_{ijk}} = \frac{\boldsymbol{exp}(\boldsymbol{\mu_{ijk}})}{\sum_{\boldsymbol{k'}=\mathbf{1}}^{\boldsymbol{r_i}} \boldsymbol{exp}(\boldsymbol{\mu_{ijk'}})}$. So the gradient of log-likelihood function wrt. μ_{ijk} is given by

$$\frac{\boldsymbol{\partial J_L(\mu_{ijk})}}{\boldsymbol{\partial \mu_{ijk}}} = \frac{\boldsymbol{\partial}}{\boldsymbol{\partial \mu_{ijk}}} \sum_{\boldsymbol{j'k'}} \boldsymbol{N_{ij'k'}} (\boldsymbol{\mu_{ij'k'}} - \boldsymbol{ln}\, \boldsymbol{Z_{j'}^i}) = \boldsymbol{N_{ijk}} - \frac{\boldsymbol{exp(\mu_{ijk})}}{\boldsymbol{Z_j^i}} \sum_{\boldsymbol{k'}} \boldsymbol{N_{ijk'}} \quad (2)$$

Where $\boldsymbol{Z_j^i} = \sum_{\boldsymbol{k}=\mathbf{1}}^{\boldsymbol{r_i}} \boldsymbol{exp}(\boldsymbol{\mu_{ijk}})$

2.3. Qualitative Constraints

Monotonicity: Monotonic influence means that stochastically, higher values of a random variable, say X result in higher (or lower) values of another variable Y, and is denoted as $\boldsymbol{X} \overset{\boldsymbol{M+}}{\succ} \boldsymbol{Y}$ (or $\boldsymbol{X} \overset{\boldsymbol{M-}}{\succ} \boldsymbol{Y}$). The interpretation is that increasing values of X shifts the cumulative distribution function of Y to the right (i.e., higher values of Y are more likely). This means that $\boldsymbol{P}(\boldsymbol{Y} \leq \boldsymbol{y} | \boldsymbol{X} = \boldsymbol{x_1}) \leq \boldsymbol{P}(\boldsymbol{Y} \leq \boldsymbol{y} | \boldsymbol{X} = \boldsymbol{x_2})$ (where $x_1 \geq x_2$). Note that the same denotation can be extended in the

presence of multiple parents by fixing the values of the other parents. Altendorf et al. [6] used these qualitative constraints to learn the parameters of a Bayes net by introducing a penalty to the objective function when the constraints are violated. Assume there is a monotonic constraint: $\boldsymbol{P(X_i \le k_c | Pa_i^{j_2}) \le P(X_i \le k_c | Pa_i^{j_1})}$ (where $\boldsymbol{Pa_i^{j_2} \ge Pa_i^{j_1}}$). Then a constraint function $\boldsymbol{\delta}$ with margin $\boldsymbol{\epsilon}$ can defined as:

$$\boldsymbol{\delta = P\left(X_i \le k_c \middle| Pa_i^{j_2}\right) - P\left(X_i \le k_c \middle| Pa_i^{j_1}\right) + \epsilon} \quad (3)$$

The corresponding penalty function is $\boldsymbol{P_{j_1,j_2}^{i,k_c} = I_{(\delta>0)}\delta^2}$ (where I=1 when $\delta > 0$ and I=0 when $\delta \le 0$). Then the gradient of the penalty function wrt. μ_{ijk} is given by

$$\boldsymbol{\frac{\partial}{\partial \mu_{ijk}} P_{j_1,j_2}^{i,k_c} = \frac{\partial}{\partial \mu_{ijk}} I_{(\delta>0)}\delta^2 = 2I_{(\delta>0)}\delta exp(\mu_{ijk})(I_{(j=j_2)} - I_{(j=j_1)})\frac{I_{(k\le k_c)}Z_j^i - Z_{jk_c}^i}{(Z_j^i)^2}} \quad (4)$$

Where $\boldsymbol{Z_j^i = \sum_{k=1}^{r_i} exp(\mu_{ijk})}$ and $\boldsymbol{Z_{jk_c}^i = \sum_{k=1}^{k_c} exp(\mu_{ijk})}$.

Synergy: If two variables X_1 and X_2 have monotonic influences on a third variable Y ($X_1 \overset{M+}{>} Y$, $X_2 \overset{M+}{>} Y$), and they have a synergistic relationship denoted as $X_1, X_2 \overset{S+}{>} Y$ (anti-synergy is denoted as S-). This means that increasing X_1 has a greater (lesser for anti-synergy) effect on Y for high values of X_2 than low values of X_2 (and likewise for increasing X_2 with fixed X_1). This synergistic constraint can be mathematically represented as:

$$\boldsymbol{P\left(Y \le k_c \middle| X_1^i, X_2^j\right) + P\left(Y \le k_c \middle| X_1^{i+1}, X_2^{j+1}\right) \le P\left(Y \le k_c \middle| X_1^{i+1}, X_2^j\right) + P\left(Y \le k_c \middle| X_1^i, X_2^{j+1}\right).}$$

We can now derive the gradients using a penalty function similar to the work on monotonicity. Similar to the monotonicity case, we can define an objective function δ for the synergistic constraints as:

$$\boldsymbol{\delta = P\left(Y \le k_c \middle| X_1^i, X_2^j\right) + P\left(Y \le k_c \middle| X_1^{i+1}, X_2^{j+1}\right) - P\left(Y \le k_c \middle| X_1^{i+1}, X_2^j\right) - P\left(Y \le k_c \middle| X_1^i, X_2^{j+1}\right) + \epsilon} \quad (5)$$

Then the gradient of the penalty function wrt. μ_{ijk} can be computed as:

$$\boldsymbol{\frac{\partial}{\partial \mu_{ijk}} P_{j_1,j_2,j_3,j_4}^{i,k_c} = 2I_{(\delta>0)}\delta exp(\mu_{ijk})(I_{(j=j_1)} + I_{(j=j_4)} - I_{(j=j_2)} - I_{(j=j_3)})\frac{I_{(k\le k_c)}Z_j^i - Z_{jk_c}^i}{(Z_j^i)^2}} \quad (6)$$

2.4. Gradient Descent Algorithm

The final function to optimize will be the log-likelihood minus the sum of all the involved penalty functions times a penalty weight w:

$$\boldsymbol{J(\mu_{ijk}) = J_L(\mu_{ijk}) - \omega \sum_{j'} P_{j'}^{i,k_c}} \quad (7)$$

The process of the algorithm is shown in Figure2.

1. Initialize the $\boldsymbol{\mu_{ijk}}$ parameters at the unconstrained point
(found simply by counting the observations, with a Laplace correction)
2. If this point satisfies the constraints, return it
3. Otherwise, initialize a weight w for the penalty functions
4. Take steps in the steepest direction of the penalized likelihood until convergence
5. If we converged outside the feasible region, increase the penalty weight and repeat the previous step.

Figure2. Constrained optimization algorithm

2.5. Experiments

An example scenario is presented in Figure 3 where the task is to predict deaths from cardiovascular disease based on the patient's physiological indexes. Based on input from our domain experts, we have the following qualitative constraints: i) high blood pressure involves three factors: high diastolic blood pressure and high systolic blood pressure both of which indicate high probability of CVD death (M+ for both), and anti-hbp-med which results in decrease of the probability (M-); ii) diabetes status influenced by three factors: high Hba1c and long duration both of which represent higher risk of CVD death (M+), and Rx-t2dm which leads to the decrease of the risk (M-); iii) cholesterol level affected by three factors: LDLC which reveals the level of bad cholesterol (M+), HDLC which represents the level of good cholesterol (M-) and lipmed which leads to lower risk of CVD death (M-) iv) male has higher probability of dying from CVD than female (M+); v) smoking leads to higher probability of CVD death (M+); vi) increasing in age results in increased risk of CVD death (M+).

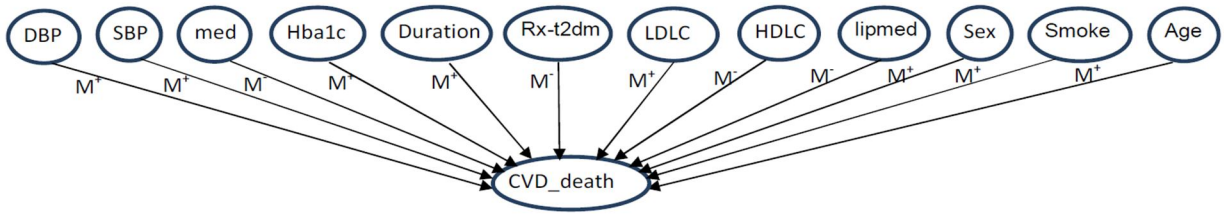

Figure 3. Qualitative Constrained Bayesian Network without hidden nodes.

Figure 4 presents the AUC-ROC of non-constrained and constrained optimization algorithms on 11 different standard machine learning domains. For each dataset, we learned the parameters by implementing 3 algorithms: learning merely from data, monotonic constraints modification and synergy constraints modification. As can be seen, in all the domains, the use of qualitative constraints outperforms learning the conditional distributions merely from data.

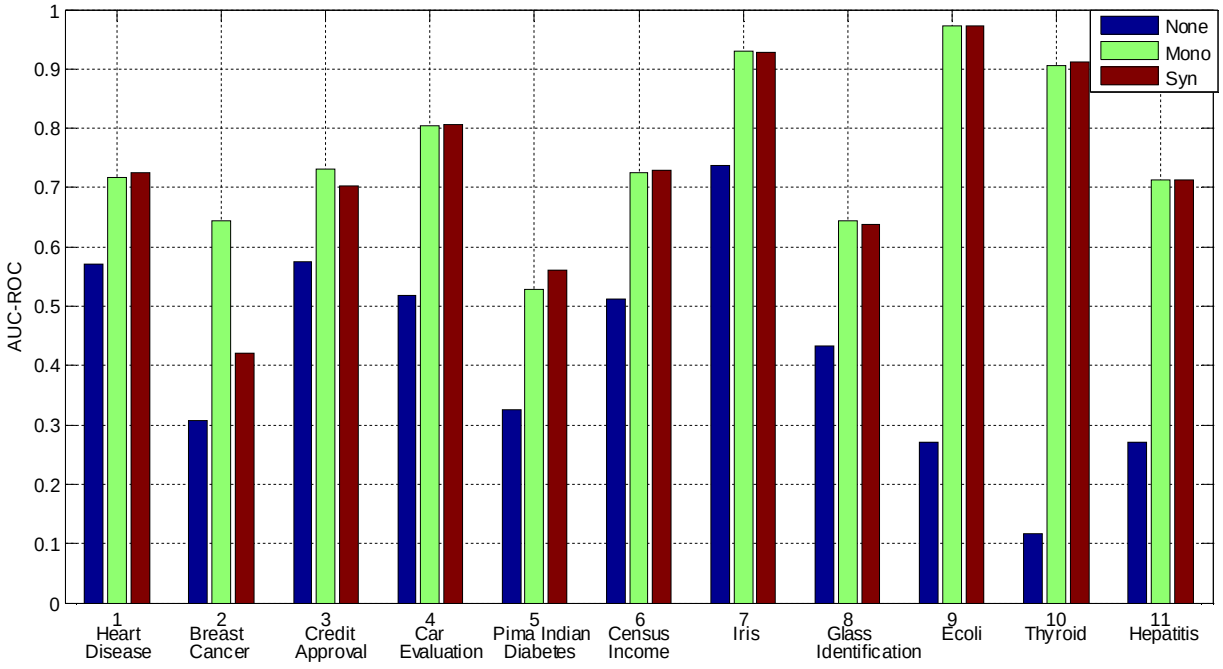

Figure 4. AUC-ROC of 3 algorithms in 11 domains.

3. Optimization in CT Imaging

3.1. Background

Since the first commercial x-ray Computed Tomography (CT) prototype was built in 1972 by Godfrey Hounsfield, CT scans have revolutionized medical imaging for providing detailed views of internal organs and structures, additionally CT scans have become a corner stone of the health care industry accounting for 2.3 billion in CT scanner sales in 2009 alone [7]. CT generates a 2D or 3D cross-sectional image from projection data obtained from different directions. This process is the so-called image reconstruction. Several other imaging modalities exist such as Positron Emission Tomography (PET) [8-11]. Single Photon Emission Computed Tomography (SPECT) [12-14] and Magnetic Resonance Imaging (MRI) [15-17]. Unlike CT, in PET and SPECT, the radioactive material is the radiation source which is located inside the body. MRI used a strong magnetic field to generate the internal structures of the body. CT is currently the most widely used imaging modality because CT is fast, easy to use, high resolution and relatively cheap than MRI or PET.

The CT scanners can be distinctively marked by the beam geometry and the scanning trajectory. There are three major types of CT, parallel beam CT, fan beam CT and cone beam CT (CBCT). Cone beam CT was developed on the basis of fan beam CT. Helical cone beam CT (CBCT) is the most common because of its high temporal resolution, high spatial resolution and can be suitable for the long object scanning. For helical CT scans the aquisition of raw data is uninterrupted, recording 800 to 1500 views for every 360 degrees of rotation.2 Parallel beam CT is used for high resolution scans of the lungs, coronary artery calcification scoring, and ECG-gated coronary CT angiography.

Generally speaking, CT scanners are composed of an x-ray source and a detector. A CT imaging system measures the initial intensity of the x-ray beam and its intensity after traversing an object. By comparing the measured x-ray intensity with the initial intensity, we can determine how much x-ray attenuation occurred during. The reconstructed CT images are the gray-level representation of the distribution of object's the x-ray attenuation coefficients in the corresponding cross-sectional plane. The attenuation coefficients vary widely for different maters. For a specific material, it depends on the material's atomic number, the x-ray intensity that transmits through it and probability of a photon being scattered or absorbed.

3.2. Mathematical Model

A lot of imaging modalities (such as CT) can be modeled as a discrete linear system,

$$\boldsymbol{AX} = \boldsymbol{b} + \boldsymbol{e} \tag{8}$$

where $A \in R_{M\times N}$ and $b \in R_M$ are known, e is an unknown noise (or perturbation) vector, and $X \in R_N$ is the "true" and unknown signal/image to be estimated.

In CT image reconstruction problems, $A = (a_{mn})_{M\times N}$ is the linear measurement matrix represents the system matrix which is depend on the CT scanning geometry. $b \in R_M$ represents the projection data. A projection is, basically, one x-ray sent through the patient and this x-ray

is attenuated differently by different tissue types. When this projection is received by a detector, 1-D information about the tissue types in that projection is also received. Then multiple other projections are taken at different angles. From these various projections a 2-D or 3-D image can be reconstructed. $X \in R_N$ is the unknown true image (ground truth, above). Both b and X are formed by stacking the columns of their corresponding two-dimensional images. The problem of estimating X from the observed projection data b is called an image reconstruction problem.

3.3. Gradient Descent Algorithm

A typical approach to problem (8) is the least squares (LS) approach in which the estimator is chosen to minimize the data error:

$$\widehat{\boldsymbol{X}}_{\boldsymbol{LS}} = \boldsymbol{argmin}_{\boldsymbol{X}} \|\boldsymbol{AX} - \boldsymbol{b}\|^2 \tag{9}$$

When M = N and A is nonsingular, the LS estimate is just the naive solution $A^{-1}b$. When the system matrix is over-determined, that is, M > N and the rank of A is N, the least square solution is $(A^TA)^{-1}(A^Tb)$. In many applications, such as CT image reconstruction, it is often the case that A is non-square matrix and extremely huge. For example, for a 256 × 256 image, 256 projections with 256 detector cells, the matrix of A should be 65536 × 65536 and a float format of A will require 16G memory. Therefore, it is unrealistic to solve X using the closed form derived by the least squared approach. Instead, iterative algorithms can be developed to find the least squared solution. Because the system is highly sparse, the non-zero components in the matrix A can be computed on the fly to save memory.

Take the derivative of equation (9) with respect to X,

$$\boldsymbol{\nabla} = \boldsymbol{2A^T(AX - b)} \tag{10}$$

The solution of equation (9) can be solved iteratively,

$$\boldsymbol{X^{k+1}} = \boldsymbol{X^k} + \boldsymbol{\alpha_k} \cdot \boldsymbol{A^T(AX - b)} \tag{11}$$

α_k is the step length which is different for each item in X. Here we take the α_k as the reciprocal of the column sums of A. The Eq. (11) can be rewritten as,

$$\boldsymbol{x_n^k} = \boldsymbol{x_n^{k-1}} + \frac{\boldsymbol{1}}{\boldsymbol{\delta \cdot a_{+n}}} \sum_{\boldsymbol{m=1}}^{\boldsymbol{M}} \boldsymbol{a_{m,n}} \left(\boldsymbol{b_m} - \boldsymbol{A_m f^{k-1}}\right) \tag{12}$$

Where $x_n = x_{s,t} > 0$, $n=(s-1)\times T + t$ with $1 \le n \le N$ and $N = S \times T$. S and T are the height and width of image. $a_{+n} = \sum_{m=1}^{M} a_{m,n} > 0$, δ is a small number, we selected δ as the maximum row sum of A. A_m is the m-th row of A, k is the iteration index. The iteration in equation (12) is repeated until the stopping criteria are satisfied, which are a minimum tolerance or a sufficient large iteration number. In our implementation, we empirically selected the iteration number 200 as the stopping criteria.

3.4. Experiments

The numerical phantom experiment was performed assuming a GE Discovery CT750 HD scanner with a circular scanning trajectory. The simulated phantom is the modified Shepp-Logan phantom which simulates the human head. The sinogram of the central slice was used to reconstruct images. The radius of the scanning trajectory was 538.5 mm. Over a 360° range,

984 projections were uniformly acquired. For each projection, 888 detector cells were equiangularly distributed, which defines a field of view of 249.2 mm in radius and an iso-center spatial resolution of 584 μm. The initial image was set to zero. All the reconstructed image sizes are 512 × 512 to cover the whole field of view (FOV), and each pixel covered an area of 973.3 × 973.3 μm. The results of the simulated phantom experiment are shown in Figure 5. We can see that as the iteration number increase, the reconstruction becomes better and better. Figure 6 shows root mean squared error (RMSE) of the projection data. We can see that we need to at least 100 iterations, so the convergence speed is very slow. In this typical CT dataset, one iteration needs about 40 seconds. 100 iterations need more than 1 hour to do the reconstruction.

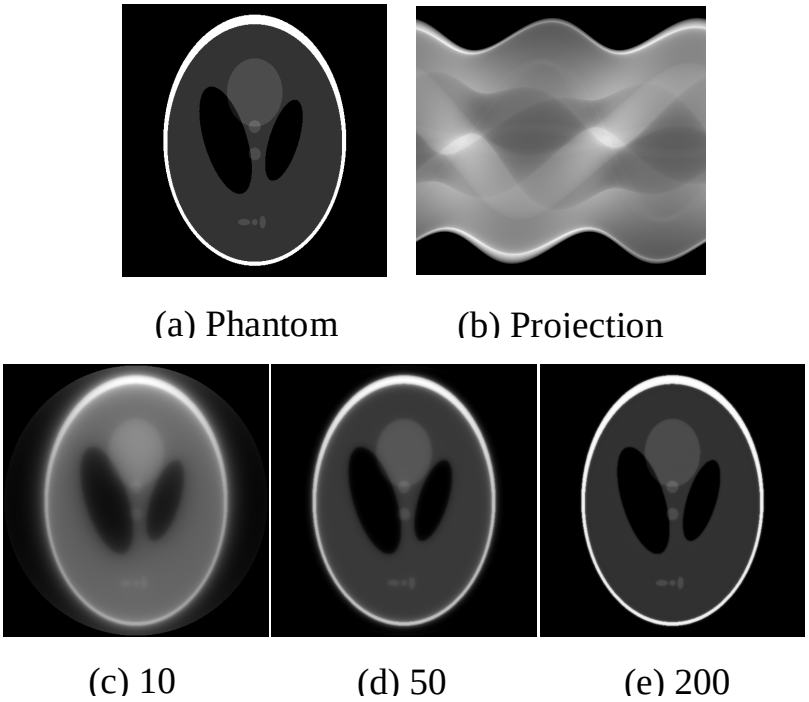

Figure 5. The simulated experimental results. From (a) to (e) are the simulated phantom, projection data, reconstructions from 20, 100 and 200 iterations, respectively.

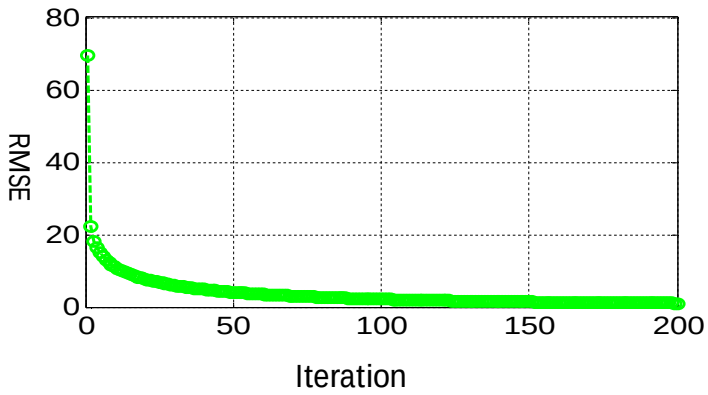

Figure 6. The root mean squared error (RMSE) of the projection data.

References

1. Prof. Plemmons's class note: "An overview of unconstraint optimization"
2. http://en.wikipedia.org/wiki/Gradient_descent
3. J. Kinsella: "Course notes for MS4327 Optimization"
4. http://en.wikipedia.org/wiki/Health_informatics#cite_note-48
5. Stuart Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995, ISBN 0-13-103805-2.
6. E. Altendorf, A. Resticar, and T. Dietterich, *Learning from sparse data by exploiting monotonicity constraints*. In UAI, pages 18-26, 2005.
7. Yorke ED, Keall P, Verhaegen F. Anniversary, *Role of medical physicists and the AAPM in improving geometric aspects of treatment accuracy and precision*. Medical Physics. 2008; 35(3):828-839
8. Burnham, C.A. and G.L. Brownell, *A Multi-Crystal Positron Camera.* Nuclear Science, IEEE Transactions on, 1972. **19**(3): p. 201-205.
9. Ter-Pogossian, M.M., et al., *A positron-emission transaxial tomograph for nuclear imaging (PETT).* Radiology, 1975. **114**(1): p. 89-98.
10. Phelps, M.E., et al., *Application of annihilation coincidence detection to transaxial reconstruction tomography.* J Nucl Med, 1975. **16**(3): p. 210-24.
11. Andersen, A.H., *Algebraic reconstruction in CT from limited views.* Medical Imaging, IEEE Transactions on, 1989. **8**(1): p. 50-55.
12. Kuhl, D.E. and R.Q. Edwards, *Image Separation Radioisotope Scanning.* Radiology, 1963. **80**: p. 653-662.
13. Jasczak, R.J., *Tomographic radiopharmaceutical imaging.* Proceedings of the IEEE, 1988. **76**(9): p. 1079-1094.
14. Neumann, D.R., N.A. Obuchowski, and F.P. Difilippo, *Preoperative 123I/99mTc-sestamibi subtraction SPECT and SPECT/CT in primary hyperparathyroidism.* J Nucl Med, 2008. **49**(12): p. 2012-7.
15. Sheil, W.C., *Magnetic Resonance Imaging (MRI Scan).* MedicineNet.com. Retrieved 08 Feb. 2013.
16. Herman, G.T. and A. Lent, *Iterative reconstruction algorithms.* Computers in Biology and Medicine, 1976. **6**(4): p. 273-294.
17. Hendee, W.R. and C.J. Morgan, *Magnetic Resonance Imaging Part I -- Physical Principles.* Western Journal of Medicine, 1984. **141**(4): p. 491-500.