CONJUGATE METHOD: A BENCHMARK IN THE NEIGHBOURHOOD OF THE MINIMUM IN A CONVEX QUADRATIC FUNCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

By assuming the Conjugate method as optimal method for convex quadratic functions, the aim of this project is to compare only one step of: Descent Gradient, Newton's Method and Conjugate Method for a quadratic optimization problem. Compute the first iteration of these algorithms, by taking same starting point randomly selected in the neighbourhood of the minimum of the convex function. Then by the euclidean distance and a non parametric test decide which algorithm is the optimal one after first iteration.

1 CONTEXT

The conjugate gradient method is an iterative method for solving a linear system of equations.

$$Ax + b = 0$$

The problem can be stated equivalently as the following minimization problem:

$$min \;\; f(x) = \frac{1}{2}x^T Ax + B^T x$$

This equivalence will allow me to interpret the conjugate gradient method either as an algorithm for solving linear systems or as a technique for minimaizing convex quadratic functions (the derivation of the steps are in APPENDIX subsection 2.)

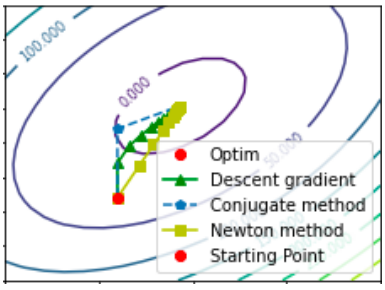

Figure 1: Behaviour of the Algorithms in the neighbourhood of a convex quadratic function

On Figure 1. I can see that the Conjugate method reaches the minimum in 2 steps, the algorithm reaches the minimum with a number of operations that does not exceed a polynomial of the problem dimensions. The direction of each step is orthogonal to the previous one due to the property of the set of conjugate vectors $\{p_0, p_1, \}$ [1]. Newton's Method, by using Hessian information, has the ability to achieve a quadratic rate of convergence in the neighbourhood of the minimum [2] , Newton's method reaches the minimum with a number of operation lower than the Descent Gradient algorithm for which, by taking a constant learning rate, the convergence rate is linear. However the most

[1] Nocedal "et al.", Numerical Optimization (" See chapter 5.")
[2] Bottou "et al." , Optimization Methods for Large-Scale Machine Learning ("See chapter 6.1.")

efficient remain the conjugate method. In some sense the conjugate gradient is the optimal first order method for convex quadratic functions [3] , (does not require access to the Hessian itself as Newton' s Method).

In the following section I investigate if after only one step the Conjugate Method is once again the optimal first order method or if the algorithm direction gets confused by the orthogonal property of Conjugate vector. Again on Figure 1. after one step the Newton's Method point directly to the minimum instead Descent Gradient and Conjugate Method have different trajectories. To investigate this I compute only the first iteration of the three algorithms starting from the same initial points randomly selected in the open spherical neighbourhood of the minimum. To evaluate the algorithms after one step I compute the euclidean distances between the points reached after one iteration and the global minimum. Then a non-parametric test is proposed to decide the optimal algorithm after one step. The Wilcoxon Mann-Whitney test [4] is used to see if the performance of Gradient Descent algorithm and Newton's Method are significantly different from the performance of Conjugate Method after one step.

2 EXPERIMENT

After having defined a strictly convex quadratic function (the study of the function is proposed in APPENDIX subsection 1.), and identify with x_1^*, x_2^*, y^* the global minimum of the function, I randomly choose 30 starting points in the open spherical neighbourhood of the minimum (the method used is explained in APPENDIX subsection 3.). I recall the definition of open spherical neighbourhood: Given x_1^*,x_2^* coordinates of the minimum of the convex quadratic function Identified by $x_{1,2}^* \in R^2$ I define the open spherical neighbourhood of center $x_{1,2}^*$ and radius r the topological space including $\forall x \in R^2$ such that $d(x_{1,2}^*, x) < r : I(x_{1,2}^*, r) = \{\forall x \in R^2; d(x_{1,2}^*, x) < r\}$ [5]. In the experiment the radius selected to study the neighbourhood of the minimum are 10 and 20.

For each starting point in the neighbourhood I compute one step of the three algorithms and store the distances from that point to the minimum in 3 different lists in order to create a sample for every algorithm.

ALGORITHM

1. $Initialization\ of: \alpha_{DG},\ \alpha_{NM},\ Distances_{DG},\ Distances_{NM},\ Distances_{CM},\ r$
2. **for** $i = 1, 2, ..., 30$ **do**
3. $Generate\ starting\ coordinate\ in\ I(x_{1,2}^*, r):\ x_{1,0} = r\sqrt{Uniform_{[0,1]}}\ \cos(Uniform_{[0,2\pi)}) +\ x_1^*$
4. $Generate\ starting\ coordinate\ in\ I(x_{1,2}^*, r):\ x_{2,0} = r\sqrt{Uniform_{[0,1]}}\ \sin(Uniform_{[0,2\pi)}) +\ x_2^*$
5. $Store\ starting\ value:\ x_0 = \begin{bmatrix} x_{1,0} \\ x_{2,0} \end{bmatrix}$
6. $First\ Step\ Gradient\ Descent:\ x_{DG,1} = x_0 - \alpha_{DG}\nabla f(x_0)$
7. $First\ Step\ Gradient\ Descent:\ y_{DG,1} = f(x_{DG,1})$
8. $Compute\ the\ euclidean\ distance:\ d_{DG,1} = d([x_{DG,1}, y_{DG,1}], [x_{1,2}^*, y^*])$
9. $Store\ d_{DG,1}\ in\ Distances_{DG}$
10. $First\ Step\ Newton's\ Method:\ x_{NM,1} = x_0 - \alpha_{NM}H^{-1}f(x_0)\nabla f(x_0)$
11. $First\ Step\ Newton's\ Method:\ y_{NM,1} = f(x_{NM,1})$
12. $Compute\ the\ euclidean\ distance:\ d_{NM,1} = d([x_{NM,1}, y_{NM,1}], [x_{1,2}^*, y^*])$
13. $Store\ d_{NM,1}\ in\ Distances_{NM}$
14. $First\ Step\ Conjugate\ Method:\ x_{CM,1} = x_0 - \frac{\langle \nabla f(x_o), \nabla f(x_0)\rangle}{\langle \nabla f(x_o), \nabla f(x_0)\rangle_A}\ \nabla f(x_0)$
15. $First\ Step\ Conjugate\ Method:\ y_{CM,1} = f(x_{CM,1})$
16. $Compute\ the\ euclidean\ distance:\ d_{CM,1} = d([x_{CM,1}, y_{CM,1}], [x_{1,2}^*, y^*])$
17. $Store\ d_{CM,1}\ in\ Distances_{CM}$
18. **End for**

[3] Bubeck, Convex Optimization: Algorithms and Complexity (" See chapter 2.4.")

[4] Birnbaum , On a use of The Mann - Whitney Statistic ("See chapter 2".)

[5] Dixmier "et al.", General Topology ("See chapter 1.3."))

I assume that the three samples, from now on called CM, DG and NM, that I have drawn with the algorithm are representative of the respectively population:

- Population of distances between first step Conjugate Method algorithm and the minimum of a convex quadratic function, considering randomly starting values in the open spherical neighbourhood of the minimum. (CM)
- Population of distances between first step Newton's Method algorithm, with a fixed learning rate, and the minimum of a convex quadratic function, considering randomly starting values in the open spherical neighbourhood of the minimum. (DG)
- Population of distances between first step Descent Gradient algorithm, with a fixed learning rate, and the minimum of a convex quadratic function, considering randomly starting values in the open spherical neighbourhood of the minimum. (NM)

From now on for simplicity I refer to these three populations by simply calling them Conjugate Method population, Descent Gradient population and Newton's Method population.

The aim is to study if the means of two populations are significantly different from each other. In particular I am interested in the Conjugate Method (benchmark) and see if its population's mean is different from the means of Newton's method population and Descent Gradient population. In order to compare the central location in two independent samples according to a parametric perspective, I can use the Student's t-test, However the assumption of normal distribution in the three population is not respected [6], in addition asymptotic results can not be used .

The corresponding nonparametric test to the Student's t-test is Wilcoxon Mann-Whitney test [7]

The three samples,CM, DG and NM, are composed of n_{CM}, n_{DG} and n_{NM} (at least ordered) observations respectively; let $X_{CM,1}, ..., X_{CM,i}, ..., X_{CM,n_{CM}}$, $X_{DG,1}, ..., X_{DG,i}, ..., X_{DG,n_{DG}}$ and $X_{NM,1}, ..., X_{NM,i}, ..., X_{NM,n_{NM}}$ be the random variables associated to the random operation of unit sampling from the corresponding populations above defined.

Assumptions:

- The random variables $X_{CM,i}, i = 1, ..., n_{CM}$ are IID with unknown distribution function $C()$;
- The random variables $X_{DG,j}, j = 1, ..., n_{DG}$ are IID with unknown distribution function $D()$;
- The random variables $X_{NM,k}, k = 1, ..., n_{NM}$ are IID with unknown distribution function $N()$.
- $C()$, $D()$, $N()$ are mutually independent.

By considering the first two distribution functions $C(x)$, $D(x)$ respectively Conjugate method and descent gradient algorithm populations, they are distinguished by a 'location shift' model that is:

$$D(x) = C(x - \Delta) \forall x \ , \ with \ \ \Delta \in R$$

Again by considering the first and the third distribution function $C(x)$, $N(x)$ respectively Conjugate method and Newton's Method populations, they are distinguished by a 'location shift' model that is:

$$N(x) = C(x - \Delta) \forall x \ , \ with \ \ \Delta \in R$$

To test if Descent gradient population and Newton's method populations have the same location of Conjugate Method population two systems of hypothesis are proposed, one tailed directional and two tailed directional. The one tailed directional has as alternative hypothesis the location of Conjugate Method population is lower than the other population. This means that if I reject the null hypothesis and the alternative is true the distances between first step Conjugate Method algorithm and the minimum of a convex quadratic function is lower than the same distances in the other algorithm. The two tailed directional has as alternative hypothesis the location of Conjugate Method population is different from the other population.

[6] Shapiro "et al.", An Analysis of Variance Test for Normality ("See pp. 591–611")

[7] Fay "et al.", Wilcoxon-Mann-Whitney or t-test? On assumptions for hypothesis tests and multiple interpretations of decision rules ("See chapter 5.")

Test the hypothesis that Descent Gradient population have the same location of Conjugate Method population.

$Two\ tailed\ H1$

$$\begin{cases} H0 : D(x) = C(x)\ \ \forall x\ that\ is,\ \ H0 : \Delta = 0 \\ H1 : D(x) > C(x)\ \ \forall x\ that\ is,\ \ H1 : \Delta < 0 \end{cases}$$

$One\ tailed\ H1$

$$\begin{cases} H0 : D(x) = C(x)\ \ \forall x\ that\ is,\ \ H0 : \Delta = 0 \\ H1 : D(x) \neq C(x)\ \ \forall x\ that\ is,\ \ H1 : \Delta \neq 0 \end{cases}$$

Test the hypothesis that Newton's Method population have the same location of Conjugate Method population.

$Two\ tailed\ H1$

$$\begin{cases} H0 : N(x) = C(x)\ \ \forall x\ that\ is,\ \ H0 : \Delta = 0 \\ H1 : N(x) > C(x)\ \ \forall x\ that\ is,\ \ H1 : \Delta < 0 \end{cases}$$

$One\ tailed\ H1$

$$\begin{cases} H0 : N(x) = C(x)\ \ \forall x\ that\ is,\ \ H0 : \Delta = 0 \\ H1 : N(x) \neq C(x)\ \ \forall x\ that\ is,\ \ H1 : \Delta \neq 0 \end{cases}$$

3 Result and Conclusion

The algorithm for the creation of the samples is repeated 10 times for each learning rate selected for Descent Gradient and Newton's Method. At every iteration , with the samples created, the p-values for both null hypothesis are computed $H0 : D(x) = C(x)$ and $H0 : N(x) = C(x)$. Then the mean of p-values related to the same system of hypothesis is stored in the summary tables (Appendix subsection 4.).

Newton's Method population and Descent gradient population reached the same location of Conjugate population with a learning rate respectively $\alpha_{NM} = 0.7$ and $\alpha_{DG} = 0.35$, in both cases the null hypothesis of the two systems (two-tailed and one-tailed) are not rejected for a significance level of 1 %. In the case of Descent gradient the optimal learning rate should be $\alpha_{DG} = 0.35$. By using learning rates smaller than those mentioned above, not only the convergence is slowed down but also create a Newton's Method population and a Descent gradient population with a location farther from zero than the Conjugate Method population. With $\alpha_{NM} < 0.6$ and $\alpha_{DG} = 0.25$ the null hypothesis of the two systems (two-tailed and one-tailed) are rejected for a significance level of 1 %. In the Descent Gradient algorithm by selecting a learning rate greater than the optimal one ($\alpha_{DG} = 0.35$) not only would be lead to divergence but also create a Descent gradient population with a location farther from zero than the Conjugate Method population. For $\alpha_{DG} > 0.4$ the null hypothesis of the two system (two-tailed and one-tailed) are rejected for a significance level of 1 %.

In the Newton's Method the optimal learning rate should be $\alpha_{NM} > 0.7$, for those learning rates the Newton's Method population created can reach a location closer to zero than the Conjugate Method population, with a p-value for one tailed system of hypothesis close to 1. Since $f(x)$ is a convex quadratic function from any initial point in the neighbourhood of the minimum with $\alpha_{NM} = 1$ the Newton's Method reach the minimum after 1 step. However a Newton iteration, with the access to the Hessian itself, demands much in terms of computation and storage. Instead the conjugate gradient method in generating its set of conjugate vectors, it can compute a new vector p_t by using only the previous vector p_{t-1} (Appendix subsection 2.), It does not need to know all the previous elements. This property implies that the method requires little storage and computation [8] .

From the results obtained in the experiment and the statement in the last paragraph I can conclude that Conjugate Method is the optimal method after the first iteration in the neighbourhood of the minimum of a convex quadratic function. The procedure adopted can be used to select the optimal learning rate for Descent gradient algorithm, in the neighbourhood of the minimum, using the Conjugate method as benchmark.

[8] Shewchuk, An Introduction to the Conjugate Gradient Method Without the Agonizing Pain ("See chapter 6.")

4 APPENDIX

4.1 THE CONVEX QUADRATIC FUNCTION USED

I proof that it is a strictly convex function and invertible, therefore I deduce the global minimum of the function.

$$f(x) = \frac{1}{2}x^T Ax + B^T x \tag{1}$$

$$f(x) = \frac{1}{2}x^T \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} x + \begin{bmatrix} 4 \\ -6 \end{bmatrix} x \tag{2}$$

$$\nabla f(x) = Ax + B^T \tag{3}$$

$$\nabla^2 f(x) = Hf(x) = A \tag{4}$$

$$Hf(x) = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \tag{5}$$

I recall the second order condition for convexity: As I already showed $f(x)$ is twice differentiable, that is, its Hessian $\nabla^2 f(x) = Hf(x)$ exists at each point in the domain of $f(x)$, which is open from now one identified by dom. Then $f(x)$ is convex if and only if the $dom \;\; f(x)$ is convex and its Hessian is positive semidefinite: $\forall x \in domf, \nabla^2 f(x) \succeq 0$ Strict convexity can be partially characterized by second-order conditions. If $\nabla^2 f(x) \succ 0, \;\; \forall x \in dom \;\; f(x)$, then $f(x)$ is strictly convex. The converse, however, is not true.

Determine the eigen values of the function to see if the matrix A is positive definite:

$$(A - \lambda I) = \begin{bmatrix} 2-\lambda & -1 \\ -1 & 2-\lambda \end{bmatrix} \tag{6}$$

$$det \begin{bmatrix} 2-\lambda & -1 \\ -1 & 2-\lambda \end{bmatrix} = (2-\lambda)(2-\lambda) - 1 = \lambda^2 - 5\lambda + 4 \tag{7}$$

$$solutions : \lambda = 4, \lambda = 1 \; positive \;\; eigen \;\; values \tag{8}$$

The function is strictly convex both eigen values are positive defined. since A is positive definite, so 0 is not an eigenvalue of A Therefore, the system of equations $Ax = 0$ has no non-trivial solution, and so A is invertible.

Determine the global minimum of the function is indeed solving the linear system of equations $Ax + b = 0$:

$$\nabla f(x) = 0 \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 4 \\ -6 \end{bmatrix} = 0 \tag{9}$$

$$\begin{cases} 2x_1 - x_2 + 4 = 0 \\ 2x_2 - x_1 - 6 = 0 \end{cases}$$

$$\begin{cases} x_1 = -\frac{2}{3} \\ x_2 = \frac{8}{3} \end{cases}$$

$$f(x^*) = \frac{1}{2}\left[-\tfrac{2}{3} \;\; \tfrac{8}{3}\right] \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} -\frac{2}{3} \\ \frac{8}{3} \end{bmatrix} + \begin{bmatrix} 4 \\ -6 \end{bmatrix} \begin{bmatrix} -\frac{2}{3} \\ \frac{8}{3} \end{bmatrix} = -\frac{28}{3} \tag{10}$$

global minimum of the function is in

$$x_1^* = -\frac{2}{3}, x_2^* = \frac{8}{3}, y^* = -\frac{28}{3}$$

4.2 CONJUGATE METHOD ALGORITHM FOR CONVEX QUADRATIC FUNCTION

$$x_{t+1} = x_t - \langle \nabla f(x_t), p_t \rangle \frac{p_t}{||p_t||_A^2} \tag{11}$$

$$where \{p_0, p_1\} \; the \;\; set \;\; of \;\; conjugate \;\; vectors \; it \;\; is \; an \; orthogonal \; set \; for \; \langle \cdot, \cdot \rangle_A \tag{12}$$

$$at \; the \; starting \; point \; t = 0, \; p_0 = \nabla f(x_0) \tag{13}$$

$$so \; the \; next \; step \; will \; be \; x_1 = x_0 - \langle \nabla f(x_o), \nabla f(x_0) \rangle \frac{\nabla f(x_0)}{\langle \nabla f(x_o), \nabla f(x_0) \rangle_A} \tag{14}$$

$$that \; is \; equivalent \; to \; x1 = x_0 - \frac{\langle \nabla f(x_o), \nabla f(x_0) \rangle}{\langle \nabla f(x_o), \nabla f(x_0) \rangle_A} \nabla f(x_0) \tag{15}$$

$$for \; t >= 1, \; p_t = \nabla f(x_t) + \frac{\langle \nabla f(x_t), \nabla f(x_t) \rangle}{\langle \nabla f(x_{t-1}), \nabla f(x_{t-1}) \rangle} p_{t-1} \tag{16}$$

$$then \; p_1 = \nabla f(x_1) + \frac{\langle \nabla f(x_1), \nabla f(x_1) \rangle}{\langle \nabla f(x_0), \nabla f(x_0) \rangle} p_0 \tag{17}$$

$$by \; applying \; \langle \nabla f(x_1), p_1 \rangle = \langle \nabla f(x_1), \nabla f(x_1) \rangle \tag{18}$$

$$x_2 = x_1 - \frac{\langle \nabla f(x_1), \nabla f(x_1) \rangle}{\langle p_1, p_1 \rangle_A} p_1 \tag{19}$$

4.3 RANDOM GENERATION OF POINT IN A CIRCLE

To generate random points within the r circle, it is incorrect to use two uniformly distributed variables $u \in [0, 1]$ and $\theta \in [0, 2\pi)$ and then take

$x_{1,0} = ru\cos(\theta) + x_1^*$
$x_{2,0} = ru\sin(\theta) + x_2^*$

As starting coordinates for the algorithms, this gives a concentration of points in the center of the circle.

The correct transformation is instead given by

$x_{1,0} = r\sqrt{u}\cos(\theta) + x_1^*$

$x_{2,0} = r\sqrt{u}\sin(\theta) + x_2^*$

Assume for simplicity that I am working with the unit circle,r = 1.

Taking a look on a perimeter of a circle with circumference 2 I should find twice as many points as the number of points on the perimeter of a circle with circumference 1. Since the circumference of a circle $2\pi r$ grows linearly with r, it follows that the number of random points should grow linearly with r. In other words, the desired probability density function $p(a)$ grows linearly. Since a $p(a)$ should have an area equal to 1 and the maximum radius is 1 $p(a) = 2a$ So I know how the desired density of my random values should look like. I have to generate such random values taking in account my original one $u \in [0, 1]$

I used the inverse transform sampling [9] :

1. Create the cumulative distribution function (CDF) defined as $P(a) = \int p(a) = \int 2a = a^2$
2. Mirror the (CDF) along $b = P(a)$
   - CDF:$b = a^2$

[9]Devroye, Non-Uniform Random Variate Generation ("See chapter 2")

- Swap:$a = b^2$
- Solve:$b = \sqrt{a}$
- CDF^{-1}:$b = \sqrt{a}$

3. Apply the resulting function to $u \in [0, 1]$: $\sqrt{u}$

4.4 MANN-WHITNEY TESTS TABLES RESULTS

Table 1: $H0: \ D(X) = C(X) \ \ in \ \ I(x^*_{1,2}, 10)$

α_{DG}	$p-value \ \ H1: \Delta \neq 0$	$p-value \ \ H1: \Delta < 0$
0.25	0.0013^{***}	0.0010^{***}
0.35	0.3092	0.1301
0.4	0.0507^{*}	0.024^{**}
0.45	0.0010^{***}	0.0004^{***}

$Significance \ \ level: \ \ 1\%^{***}, 5\%^{**}, 10\%^{*}$

Table 2: $H0: \ D(X) = C(X) \ \ in \ \ I(x^*_{1,2}, 20)$

α_{DG}	$p-value \ \ H1: \Delta \neq 0$	$p-value \ \ H1: \Delta < 0$
0.25	0.0005^{***}	0.0003^{***}
0.35	0.3436	0.1305
0.4	0.0703	0.0239^{**}
0.45	0.0005^{***}	0.0002^{***}

$Significance \ \ level: \ \ 1\%^{***}, 5\%^{**}, 10\%^{*}$

Table 3: $H0: \ N(X) = C(X) \ in \ I(x^*_{1,2}, 10)$

α_{NM}	$p-value \ H1: \Delta \neq 0$	$p-value \ H1: \Delta < 0$
0.5	0.0001^{***}	0.0001^{***}
0.55	0.0003^{***}	0.0009^{***}
0.6	0.0129^{**}	0.0061^{***}
0.7	0.3991	0.2925

$Significance \ level: \ 1\%^{***}, 5\%^{**}, 10\%^{*}$

Table 4: $H0: \ N(X) = C(X) \ in \ I(x^*_{1,2}, 20)$

α_{NM}	$p-value \ H1: \Delta \neq 0$	$p-value \ H1: \Delta < 0$
0.5	0.00017^{***}	0.0001^{***}
0.55	0.0040^{***}	0.001^{***}
0.6	0.0235^{**}	0.0199^{**}
0.7	0.2698	0.1299

$Significance \ level: \ 1\%^{***}, 5\%^{**}, 10\%^{*}$

5 REFERENCES

[1] Nocedal ”et al.”, Numerical Optimization (” See chapter 5.”)

[2] Bottou ”et al.” , Optimization Methods for Large-Scale Machine Learning (”See chapter 6.1.”)

[3] Bubeck, Convex Optimization: Algorithms and Complexity (” See chapter 2.4.”)

[4] Birnbaum , On a use of The Mann - Whitney Statistic (”See chapter 2”.)

[5] Dixmier ”et al.”, General Topology (”See chapter 1.3.”)

[6] Shapiro ”et al.”, An Analysis of Variance Test for Normality (”See pp. 591–611”)

[7] Fay ”et al.”, Wilcoxon-Mann-Whitney or t-test? On assumptions for hypothesis tests and multiple interpretations of decision rules (”See chapter 5.”)

[8] Shewchuk, An Introduction to the Conjugate Gradient Method Without the Agonizing Pain (”See chapter 6.”)

[9] Devroye, Non-Uniform Random Variate Generation (”See chapter 2”).