1DR2243:A - Teoria i metody optymalizacji, lab.
Nieliniowe zadanie LS z więzami

(opracował: M.T., ostatnia modyfikacja: 13 grudnia 2022)

1 Wstęp

1.1 Nieliniowe zadanie najmniejszych kwadratów z więzami

Nieliniowe zadanie najmniejszych kwadratów z więzami ma postać

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|f(x)\|^2 \\ \text{subject to} \quad & g(x) = 0 \end{aligned} \tag{1}$$

gdzie $f : \mathbb{R}^n \to \mathbb{R}^m$, $g : \mathbb{R}^n \to \mathbb{R}^p$, są różniczkowalnymi funkcjami wektorowymi

$$f(x) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{bmatrix}, \qquad g(x) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_p(x) \end{bmatrix}. \tag{2}$$

Zadanie (1) możemy sformułować inaczej jako

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f_1(x)^2 + \ldots + f_m(x)^2, \\ \text{subject to} \quad & g_1(x) = 0 \\ & \vdots \\ & g_p(x) = 0 \end{aligned} \tag{3}$$

Punkt $x \in \mathbb{R}^n$ nazywamy *dopuszczalnym* (ang. *feasable*) dla zadania (1) jeżeli spełnia równanie

$$g(x) = 0. \tag{4}$$

Punkt $x^\star \in \mathbb{R}^n$ nazywamy rozwiązaniem zadania (1) jeżeli jest dopuszczalny i daje najmniejszą wartość funkcji celu spośród wszsystkich punktów dopuszczalnych, tzn. $g(x^\star) = 0$ oraz

$$g(x) = 0 \Longrightarrow \left(\|f(x^\star)\|^2 \leqslant \|f(x)\|^2\right). \tag{5}$$

Przypomnijmy, że nieliniowe zadanie najmniejszych kwadratów **bez więzów**, czyli zadanie optymalizacji postaci

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \|f(x)\|^2, \tag{6}$$

gdzie $f : \mathbb{R}^n \to \mathbb{R}^m$

$$f(x) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_m(x) \end{bmatrix} \tag{7}$$

jest zadaną (różniczkowalną) funkcją, można rozwiązywać za pomocą algorytmu Levenberga–Marquardta [1]. Metody tej nie można bezpośrednio zatsosować do zadań typu (1), czyli z więzami, jednak stanowi ona ważny składnik omówionych w dalszym części algorytmów, odpowiednio, kary i rozszerzonego lagranżjanu, za pomocą których można rozwiązywać zadania z więzami.

1.2 Algorytm kary (ang. *penalty algorithm*)

Zadanie (1) możemy traktować jako przypadek graniczny zadania

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \|f(x)\|^2 + \mu\|g(x)\|^2, \tag{8}$$

dla dużych wartości parametru $\mu > 0$. Wyraz $\mu\|g(x)\|^2$ nazywamy składnikiem kary (ang. *penalty term*). Zauważmy, że

$$\|f(x)\|^2 + \mu\|g(x)\|^2 = \left\| \begin{bmatrix} f(x) \\ \sqrt{\mu} g(x) \end{bmatrix} \right\|^2 \tag{9}$$

zatem dla ustalonej wartości parametru μ zadanie (8) można rozwiązać metodą Levenberga-Marquardta. Dla dużych wartości μ powinniśmy otrzymać minimalizator $x^\star_\mu$ dla którego $\|f(x^\star_\mu)\|^2$ jest możliwie małe, zaś $\|g(x^\star_\mu)\|^2$ bardzo małe, $\|g(x^\star_\mu)\|^2 \approx 0$, czyli $g(x^\star_\mu) \approx 0$. Oznacza to, że dla odpowiednio dużej warości parametru μ, rozwiązanie zadania (8), które oznaczamy $x^\star_\mu$, jest przybliżonym rozwiązaniem zadania (1). Posumowując, algorytm kary dla nieliniowego zadania LS z więzami (ang. *penalty algorithm for constraint nonlinear least squares*) jest następujący.
Dane: $f : \mathbb{R}^n \to \mathbb{R}^m$, $g : \mathbb{R}^n \to \mathbb{R}^p$, punkt startowy $x^{(0)}$, maksymalna liczba iteracji $k_{\max}$, $\mu^{(0)} = 1$. Dla $k = 0, 1, \ldots, k_{\max}$.
Dla $k = 0, 1, \ldots, k_{\max}$

1. Metodą LM, przyjmując punkt startowy $x^{(k)}$, $\mu^{(k)}$ znajdź rozwiązanie $x^\star_\mu$ zadania

   $$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \|\tilde{f}(x)\|^2, \tag{10}$$

   gdzie

   $$\tilde{f}(x) = \begin{bmatrix} f(x) \\ \sqrt{\mu} g(x) \end{bmatrix}, \tag{11}$$

   a następnie podstaw

   $$x^{(k+1)} = x^\star_\mu. \tag{12}$$

2. Wykonaj aktualizację

   $$\mu^{(k+1)} = 2\mu^{(k)}. \tag{13}$$

Teoretycznie można by od razu wziąć odpowiednio dużą wartość parametru μ, jednak prowadziłoby to do trudności numerycznych, dlatego parametr ten lepiej zwiększać stopniowo w kolejnych iteracjach. W praktyce nawet przy takim ostrożnym podejściu algorytm kary nie zawsze jest skuteczny, dlatego częściej stosuje się tzw. algorytm rozszerzonego lagranżjanu (ang. *augmented Lagrangian algorithm, ALA*) [1, 2].

1.3 Algorytm rozszerzonego lagranżjanu

Lagranżjan dla zadania (1) jest postaci

$$\begin{aligned} L(x, z) &= \|f(x)\|^2 + \sum_{i=1}^{p} z_i g_i(x) \\ &= \|f(x)\|^2 + z^{\mathrm{T}} g(x), \end{aligned} \tag{14}$$

gdzie $z \in \mathbb{R}^p$ jest wektorem mnożników Lagrange'a. Na mocy twierdzenia Lagrange'a wiadomo, że warunkiem koniecznym optymalności dla zadania (1) jest zerowanie się gradentu lagranżjanu, tzn. $\nabla L(x, z) = 0$. Mamy

$$\nabla L(x, z) = \begin{bmatrix} \nabla_x L(x, z) \\ \nabla_z L(x, z) \end{bmatrix} = 0, \tag{15}$$

Mamy

$$
\begin{aligned}
\nabla\|f(x)\|_2^2 &= \nabla\left(\sum_{i=1}^{N} f_i(x)^2\right)\\
&= \sum_{i=1}^{N}\nabla\left(f_i(x)^2\right)\\
&= \sum_{i=1}^{N} 2f_i(x)\nabla f_i(x)\\
&= 2[\mathrm{D}f(x)]^{\mathrm{T}} f(x), \qquad (16)
\end{aligned}
$$

oraz

$$
\begin{aligned}
\nabla(z^{\mathrm{T}}g(x)) &= \nabla\left(\sum_{i=1}^{p} z_i g_i(x)\right)\\
&= \sum_{i=1}^{p} z_i\nabla g_i(x)\\
&= \begin{bmatrix}\nabla g_1(x) & \dots & \nabla g_p(x)\end{bmatrix}\begin{bmatrix} z_1\\ \vdots \\ z_p\end{bmatrix}\\
&= [\mathrm{D}g(x)]^{\mathrm{T}} z, \qquad (17)
\end{aligned}
$$

zatem, uwzględniając (14), z warunku (15) otrzymujemy

$$
\begin{aligned}
\nabla L(x,z) &= \begin{bmatrix}\nabla_x L(x,z)\\ \nabla_z L(x,z)\end{bmatrix}\\
&= \begin{bmatrix}\nabla_x\left(\|f(x)\|^2 + z^{\mathrm{T}}g(x)\right)\\ \nabla_z\left(\|f(x)\|^2 + z^{\mathrm{T}}g(x)\right)\end{bmatrix}\\
&= \begin{bmatrix}2[\mathrm{D}f(x)]^{\mathrm{T}}f(x) + [\mathrm{D}g(x)]^{\mathrm{T}}z\\ g(x)\end{bmatrix}\\
&= 0, \qquad (18)
\end{aligned}
$$

czyli

$$
\begin{aligned}
2[\mathrm{D}f(x)]^{\mathrm{T}}f(x) + [\mathrm{D}g(x)]^{\mathrm{T}}z = 0, \qquad (19a)\\
g(x) = 0. \qquad (19b)
\end{aligned}
$$

Każde rozwiązanie zadania (1) musi spełniać równania (19). Tzw. rozszerzony lagranżjan (ang. *augmented Lagrangian*) dla zadania (1) definiujemy następująco

$$
\begin{aligned}
L_\mu(x,z) &= L(x,z) + \mu\|g(x)\|^2\\
&= \|f(x)\|^2 + z^{\mathrm{T}}g(x) + \mu\|g(x)\|^2, \qquad (20)
\end{aligned}
$$

i jest „zwykłym” lagranżjanem dla zadania

$$
\begin{aligned}
&\underset{x}{\text{minimize}} \quad \|f(x)\|^2 + \mu\|g(x)\|^2, \qquad (21)\\
&\text{subject to} \quad g(x) = 0, \qquad (22)
\end{aligned}
$$

które jest równoważne zadaniu (1) Zauważmy, że

$$
\begin{aligned}
\mu\left\|g(x) + \frac{1}{2\mu}z\right\|^2 &= \mu\left[g(x) + \frac{1}{2\mu}z\right]^{\mathrm{T}}\left[g(x) + \frac{1}{2\mu}z\right]\\
&= \mu\|g(x)\|^2 + 2\mu[g(x)]^{\mathrm{T}}\left(\frac{1}{2\mu}\right)z + \mu\left\|\frac{1}{2\mu}z\right\|^2\\
&= \mu\|g(x)\|^2 + [g(x)]^{\mathrm{T}}z + \mu\left\|\frac{1}{2\mu}z\right\|^2 \qquad (23)
\end{aligned}
$$

zatem

$$
[g(x)]^{\mathrm{T}}z + \mu\|g(x)\|^2 = \mu\left\|g(x) + \frac{1}{2\mu}z\right\|^2 - \mu\left\|\frac{1}{2\mu}z\right\|^2, \qquad (24)
$$

zatem

$$
\begin{aligned}
L_\mu(x,z) &= \|f(x)\|^2 + [g(x)]^{\mathrm{T}}z + \mu\|g(x)\|^2,\\
&= \|f(x)\|^2 + \mu\left\|g(x) + \frac{1}{2\mu}z\right\|^2 - \mu\left\|\frac{1}{2\mu}z\right\|^2 \qquad (25)
\end{aligned}
$$

W metodzie ALA minimalizujemy względem x, używając algorytmu LM, rozszerzony lagranżjan (20) dla ciągu wartości $(\mu^{(k)}, z^{(k)})$, $k = 0, 1, \ldots$. Ponieważ wyrażenie

$$
\mu\left\|\frac{1}{2\mu}z\right\|^2 \qquad (26)
$$

nie zależy od x, to de facto minimalizujemy

$$
\|f(x)\|^2 + \mu\left\|g(x) + \frac{1}{2\mu}z\right\|^2 \qquad (27)
$$

czyli

$$
\left\|\begin{bmatrix} f(x)\\ \sqrt{\mu}g(x) + \frac{1}{2\sqrt{\mu}}z\end{bmatrix}\right\|^2. \qquad (28)
$$

Dowolny minimalizator $\tilde{x}$ wyrażenia (27), musi spełniać warunek optymalności

$$
\begin{aligned}
0 &= 2[\mathrm{D}f(\tilde{x})]^{\mathrm{T}}f(\tilde{x}) + 2\mu[\mathrm{D}g(\tilde{x})]^{\mathrm{T}}\left(g(\tilde{x}) + \frac{1}{2\mu}z\right)\\
&= 2[\mathrm{D}f(\tilde{x})]^{\mathrm{T}}f(\tilde{x}) + [\mathrm{D}g(\tilde{x})]^{\mathrm{T}}\left(2\mu g(\tilde{x}) + z\right). \qquad (29)
\end{aligned}
$$

Innymi słowy, po zminimalizowaniu (względem x) metodą LM wyrażenia (27) mamy minimalizator $\tilde{x}$, który spełnia równanie

$$
2[\mathrm{D}f(\tilde{x})]^{\mathrm{T}}f(\tilde{x}) + [\mathrm{D}g(\tilde{x})]^{\mathrm{T}}\left(2\mu g(\tilde{x}) + z\right) = 0. \qquad (30)
$$

Chcielibyśmy, aby były spełnione zależności

$$
\begin{aligned}
2[\mathrm{D}f(\tilde{x})]^{\mathrm{T}}f(\tilde{x}) + [\mathrm{D}g(\tilde{x})]^{\mathrm{T}}z = 0, \qquad (31a)\\
g(\tilde{x}) = 0. \qquad (31b)
\end{aligned}
$$

Żeby zachodziła równość (31a) wystarczy przyjąć

$$
\tilde{z} = 2\mu g(\tilde{x}) + z. \qquad (32)
$$

Bardzo się ciesząc z tego sprytnego posunięcia, musimy pamiętać, że wciąż trzeba „coś zrobić” z równością (31b), która w ogólnym przypadku nie jest spełniona. Dlatego dla nowej wartości $\tilde{z}$ ponownie minimalizujemy rozszerzony lagranżjan wględem x.

Posumowując, algorytm ALA dla nieliniowego zadania LS z więzami (ang. *penalty algorithm for constraint nonlinear least squares*) jest następujący.

Dane: $f : \mathbb{R}^n \to \mathbb{R}^m$, $g : \mathbb{R}^n \to \mathbb{R}^p$, punkt startowy $x^{(0)}$, maksymalna liczba iteracji $k_{\max}$, $\mu^{(0)} = 1$, $z^{(1)} = \mathbf{1}$.

Dla $k = 0, 1, \ldots, k_{\max}$:

1. Metodą LM, przyjmując punkt startowy $x^{(k)}$ znajdź rozwiązanie $x_\mu^\star$ zadania

$$
\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \|f(x)\|^2 + \mu^{(k)}\left\|g(x) + \frac{1}{2\mu^{(k)}}z^{(k)}\right\|^2, \qquad (33)
$$

   Podstaw

$$
x^{(k+1)} = x_\mu^\star. \qquad (34)
$$

2. Wykonaj aktualizację

$$
z^{(k+1)} = z^{(k)} + 2\mu^{(k)}g(x^{(k+1)}) \qquad (35)
$$

   oraz

$$
\mu^{(k+1)} = \begin{cases}\mu^{(k)} & \text{jeśli } \left\|g(x^{(k+1)})\right\| < 0.25\left\|g(x^{(k+1)})\right\|\\ 2\mu^{(k)} & \text{jeśli } \left\|g(x^{(k+1)})\right\| \geqslant 0.25\left\|g(x^{(k+1)})\right\|\end{cases}
$$

Zauważmy, że punkt pierwszy sprowadza się do minimalizacji (względem x) metodą LM wyrażenia

$$\left\| \begin{bmatrix} f(x) \\ \sqrt{\mu^{(k)}}g(x) + \frac{1}{2\sqrt{\mu^{(k)}}}z^{(k)} \end{bmatrix} \right\|^2, \tag{36}$$

Algorytm ALA nie jest badziej skomplikowany niż algorytm kary, jednak w praktyce działa znacznie lepiej, głównie z powodu wolniejszego wzrostu wartości parametru $\mu^{(k)}$ w miarę wzrostu indeksu k.

Przykład 1. [1] Stosując metodę ALA rozwiązać zadanie

$$\underset{x}{\text{minimize}} \quad \|f(x)\|^2, \tag{37a}$$

$$\text{subject to} \quad g(x) = 0, \tag{37b}$$

gdzie

$$f(x) = \begin{bmatrix} x_1 + \exp(-x_2) \\ x_1^2 + 2x_2 + 1 \end{bmatrix}, \tag{38a}$$

$$g(x) = x_1 + x_1^3 + x_2 + x_2^2. \tag{38b}$$

Przyjmując punkt startowy

$$x^{(0)} = \begin{bmatrix} 0.5000 \\ -0.5000 \end{bmatrix}, \tag{39}$$

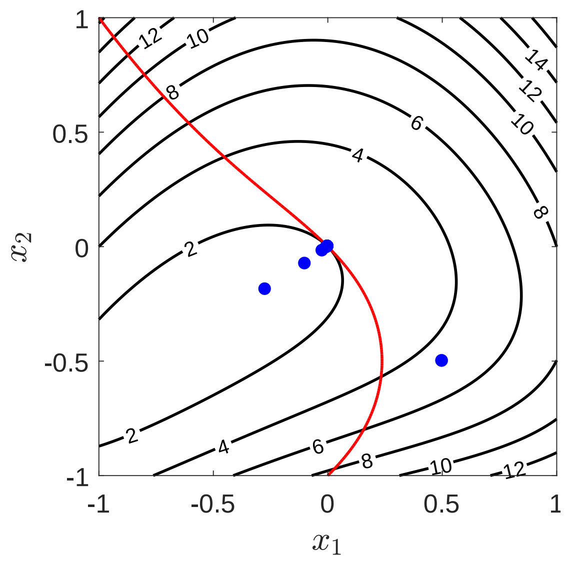
otrzymujemy

$$x^{(1)} = \begin{bmatrix} -0.2730 \\ -0.1866 \end{bmatrix}, \quad x^{(2)} = \begin{bmatrix} -0.0993 \\ -0.0747 \end{bmatrix},$$

$$x^{(3)} = \begin{bmatrix} -0.0232 \\ -0.0183 \end{bmatrix}, \quad x^{(4)} = \begin{bmatrix} -0.0054 \\ -0.0043 \end{bmatrix},$$

$$x^{(5)} = \begin{bmatrix} -0.0013 \\ -0.0010 \end{bmatrix}, \quad x^{(6)} = \begin{bmatrix} -0.0003 \\ -0.0002 \end{bmatrix},$$

$$x^{(7)} = \begin{bmatrix} -0.0001 \\ -0.0001 \end{bmatrix}, \quad x^{(8)} = \begin{bmatrix} -0.0000 \\ -0.0000 \end{bmatrix}$$



Rysunek 1: Krzywa czerwona to zerowa poziomica funkcji g, natomiast krzywe czarne są poziomicami funkcji $\|f\|^2$. Niebieskie kropy oznaczają kolejne punkty iteracji. (Przykład 1)

Przypomnijmy warunki optymalności (19)

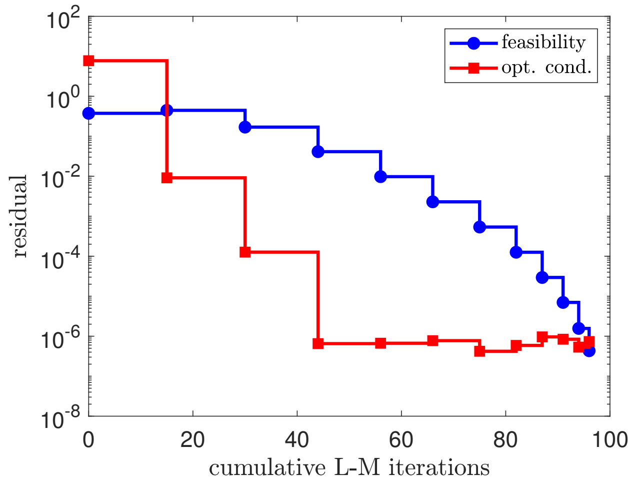
$$2[\mathrm{D}f(x)]^\mathrm{T} f(x) + [\mathrm{D}g(x)]^\mathrm{T} z = 0,$$
$$g(x) = 0.$$

Zdefiniujmy, opdowiednio, residuum warunku optymalnosci (ang. *optimality condition residual, OR* i residuum warunku dopuszczalnosci (ang. *feasibility residual, FR*),

$$\mathrm{OR} = \|2[\mathrm{D}f(x)]^\mathrm{T} f(x) + [\mathrm{D}g(x)]^\mathrm{T} z\|, \tag{40a}$$

$$\mathrm{FR} = \|g(x)\|. \tag{40b}$$

Wraz ze wzrosetm liczby iteracji wskaźniki te powinny być coraz mniejsze, w teorii wyrażenia, odpowiednio, $\|2[\mathrm{D}f(x)]^\mathrm{T} f(x) + [\mathrm{D}g(x)]^\mathrm{T} z\|$ i $\|g(x)\|$ powinny dążyć do zera. W Tabeli 1 przedstawiono wartości logarytmów tych wskaźników dla kolejnych iteracji, tzn. dla kolejnych par $(x^{(0)}, z^{(0)})$, $(x^{(1)}, z^{(1)})$, itd., natomiast na Rys. 2 wykresy tych wskaźników w funkcji skumulowanej liczby iteracji LM.



Rysunek 2: Residuum warunku dopuszczalnosci (ang. *feasibility residual*) i residuum warunku optymalnosci (ang. *optimality condition residual*) w funkcji skumulowanych iteracji Levenberga-Marquardta, w metodzie ALA. (Przykład 1)

Tabela 1: Wartości residuów dla warunków wykonalności (ang. *feasibility residual, FR*) i optymalności (ang. *optimality condition residual, OR*).

k	$\log_{10}$(FR)	$\log_{10}$(OR)
0	-0.4260	0.8892
1	-0.3515	-2.0393
2	-0.7710	-3.8973
3	-1.3848	-6.1855
4	-2.0100	-6.1758
5	-2.6393	-6.1117
6	-3.2697	-6.3768
7	-3.9004	-6.2303
8	-4.5313	-6.0172
9	-5.1553	-6.0757
10	-5.8058	-6.2736

2 Zadania

Zadanie 1. Powtórzyć obliczenia z przykładu 1 i wykonać odpowiednie wykresy (Rys. 1). Wyznaczyć wartości wskaźników FR i OR dla kolejnych iteracji.

Zadanie 2. Wykonać ćwiczenie 19.1 z [1]. (Rozwiązanie $x^\star = [0.5677\ 0.8328\ 0.5753]$).

Zadanie 3. Wykonać ćwiczenie 19.3 z [1]

Literatura

[1] S. Boyd and L. Vandenberghe. *Introduction to Applied Linear Algebra: Vectors, Matrices, and Least Squares*. Cambridge University Press, 2018.

[2] A. R. Conn, N. I. M. Gould, and P. L. Toint. A globally convergent augmented lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM journal on numerical analysis*, 28(2):545–572, 1991.

19.1 *Projection on a curve.* We consider a constrained nonlinear least squares problem with three variables $x = (x_1, x_2, x_3)$ and two equations:

$$\begin{array}{ll} \text{minimize} & (x_1-1)^2 + (x_2-1)^2 + (x_3-1)^2 \\ \text{subject to} & x_1^2 + 0.5x_2^2 + x_3^2 - 1 = 0 \\ & 0.8x_1^2 + 2.5x_2^2 + x_3^2 + 2x_1x_3 - x_1 - x_2 - x_3 - 1 = 0. \end{array}$$

The solution is the point closest to $(1,1,1)$ on the nonlinear curve defined by the two equations.

(a) Solve the problem using the augmented Lagrangian method. You can start the algorithm at $x^{(1)} = 0$, $z^{(1)} = 0$, $\mu^{(1)} = 1$, and start each run of the Levenberg–Marquardt method with $\lambda^{(1)} = 1$. Stop the augmented Lagrangian method when the feasibility residual $\|g(x^{(k)})\|$ and the optimality condition residual

$$\|2Df(x^{(k)})^T f(x^{(k)}) + Dg(x^{(k)})^T z^{(k)}\|$$

are less than 10^{-5}. Make a plot of the two residuals and of the penalty parameter μ versus the cumulative number of Levenberg–Marquardt iterations.

(b) Solve the problem using the penalty method, started at $x^{(1)} = 0$ and $\mu^{(1)} = 1$, and with the same stopping condition. Compare the convergence and the value of the penalty parameter with the results for the augmented Lagrangian method in part (a).

Rysunek 3: Ćwiczenie 19.1 z [1]

19.3 *Boolean least squares.* The *Boolean least squares problem* is a special case of the constrained nonlinear least squares problem (19.1), with the form

$$\begin{array}{ll} \text{minimize} & \|Ax-b\|^2 \\ \text{subject to} & x_i^2 = 1, \quad i = 1, \ldots, n, \end{array}$$

where the n-vector x is the variable to be chosen, and the $m \times n$ matrix A and the m-vector b are the (given) problem data. The constraints require that each entry of x is either -1 or $+1$, *i.e.*, x is a Boolean vector. Since each entry can take one of two values, there are 2^n feasible values for the vector x. The Boolean least squares problem arises in many applications.

One simple method for solving the Boolean least squares problem, sometimes called the *brute force method*, is to evaluate the objective function $\|Ax - b\|^2$ for each of the 2^n possible values, and choose one that has the least value. This method is not practical for n larger than 30 or so. There are many heuristic methods that are much faster to carry out than the brute force method, and approximately solve it, *i.e.*, find an x for which the objective is small, if not the smallest possible value over all 2^n feasible values of x. One such heuristic is the augmented Lagrangian algorithm 19.2.

(a) Work out the details of the update step in the Levenberg–Marquardt algorithm used in each iteration of the augmented Lagrangian algorithm, for the Boolean least squares problem.

(b) Implement the augmented Lagrangian algorithm for the Boolean least squares problem. You can choose the starting point $x^{(1)}$ as the minimizer of $\|Ax - b\|^2$. At each iteration, you can obtain a feasible point $\tilde{x}^{(k)}$ by rounding the entries of $x^{(k)}$ to the values ± 1, *i.e.*, $\tilde{x}^{(k)} = \mathbf{sign}(x^{(k)})$. You should evaluate and plot the objective value of these feasible points, *i.e.*, $\|A\tilde{x}^{(k)} - b\|^2$. Your implementation can return the best rounded value found during the iterations. Try your method on some small problems, with $n = m = 10$ (say), for which you can find the actual solution by the brute force method. Try it on much larger problems, with $n = m = 500$ (say), for which the brute force method is not practical.

Rysunek 4: Ćwiczenie 19.3 z [1]