15 Regularitätsbedingungen (CQ), Verfahren hinter `fmincon`

Optimierung für Nichtmathematiker

WS 2020/21

Optimalitätsbedingungen

Wir nehmen an, x^* sei ein lokales Minimum der Aufgabe

$$\boxed{\begin{aligned} \text{Minimiere} \quad & f(x), && x \in \mathbb{R}^n, \\ \text{sodass} \quad & g_i(x) \leqslant 0, && i = 1, \ldots, m, \\ \text{und} \quad & h_j(x) = 0, && j = 1, \ldots, p. \end{aligned}}$$

Falls bei x^* eine Constraint Qualification (CQ) gilt, dann gibt es Multiplikatoren $\mu \in \mathbb{R}^m$, $\lambda \in \mathbb{R}^p$, sodass die KKT-Bedingungen gelten:

$$\nabla_x \mathcal{L}(x^*, \mu, \lambda) = \nabla f(x^*) + g'(x^*)^\mathsf{T} \mu + h'(x^*)^\mathsf{T} \lambda = 0,$$

$$\left.\begin{aligned} \mu_i \geq 0, & \text{ falls } g_i(x^*) = 0 \\ \mu_i = 0, & \text{ falls } g_i(x^*) < 0 \end{aligned}\right\} \quad \mu \geqslant 0, \quad g(x^*) \leqslant 0, \quad \mu^\mathsf{T} g(x^*) = 0,$$

$$h(x^*) = 0.$$

Quizfrage

Was versteht man unter der Constraint Qualification der linearen Unabhängigkeit (LICQ) an einem zulässigen Punkt $\bar{x}$?

o A Die Funktionen $\{h_j(x)\}_{j=1}^p \cup \{g_i(x)\}_{i \in \mathcal{A}_g(\bar{x})}$ sind linear unabhängig.

o B Die Vektoren $\{\nabla h_j(\bar{x})\}_{j=1}^p \cup \{\nabla g_i(\bar{x})\}_{i=1}^m$ sind linear unabhängig.

o C Die Vektoren $\{\nabla h_j(\bar{x})\}_{j=1}^p$ sind linear unabhängig.

9 D Die Vektoren $\{\nabla h_j(\bar{x})\}_{j=1}^p \cup \{\nabla g_i(\bar{x})\}_{i \in \mathcal{A}_g(\bar{x})}$ sind linear unabhängig.

bei $\bar{x}$ aktiv

Verschiedene CQs

Das gilt auch für LP!

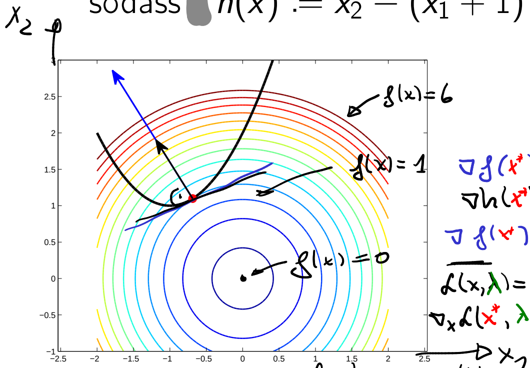
- Unter der LICQ sind die Lagrange-Multiplikatoren an einem lokalen Minimum sogar eindeutig.
- Wenn die LICQ in einem lokalen Minimum nicht gelten, so heißt das noch nicht, dass man dort keine Lagrange-Multiplikatoren finden kann. Es könnte auch eine schwächere CQ erfüllt sein, die wir hier nicht besprechen.
- Bei linearen Optimierungsaufgaben (LP) gelten immer bestimmte CQ, sodass Multiplikatoren existieren.

Abadie CQ

$$\nabla h(x) = \begin{pmatrix} -2(x_1+1) \\ 1 \end{pmatrix} \neq 0$$ Die LICQ ist an jedem Punkt erfüllt.

Illustration notwend. Bedingungen

$$\text{Minimiere} \quad f(x) := x_1^2 + x_2^2, \qquad x = (x_1, x_2) \in \mathbb{R}^2$$
$$\text{sodass} \quad h(x) := x_2 - (x_1 + 1)^2 - 1 = 0$$



3D-Sicht auf f

$\nabla f(x^*)$, $\nabla h(x^*)$ sind Vielfache voneinander

$$\nabla f(x^*) = \underbrace{-\lambda}_{\text{Unbekannt}} \nabla h(x^*)$$

$$\mathcal{L}(x,\lambda) = f(x) + \lambda h(x)$$

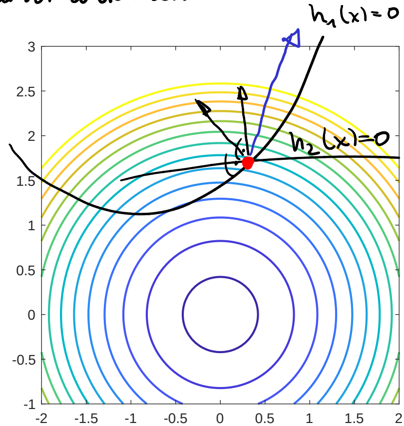
$$\nabla_x \mathcal{L}(x^*, \lambda) = \nabla f(x^*) + \lambda \nabla h(x^*) \boxed{= 0}$$

$\mathcal{L}$ = Lagrangefkt.

$\{\nabla h_1(x^*), \nabla h_2(x^*)\}$ ist eine linear unabhängige Menge von Vektoren im $\mathbb{R}^2$, also gilt LICQ bei x^*.

Mehrere Gleichungsbedingungen

x^* ist isolierter Punkt der zulässigen Menge und damit automatisch lokales Minimum.



$\in \mathbb{R}^2$

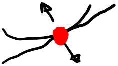
$\mathcal{L}(x,\lambda) = f(x) + \lambda_1 h_1(x) + \lambda_2 h_2(x)$

$\nabla_x \mathcal{L}(x^*,\lambda) = \nabla f(x^*) + \lambda_1 \nabla h_1(x^*) + \lambda_2 \nabla h_2(x^*) \boxed{=0}$

d.h.

$\nabla f(x^*) = -\lambda_1 \nabla h_1(x^*) - \lambda_2 \nabla h_2(x^*)$

Man kann solche λ_1, λ_2 finden!

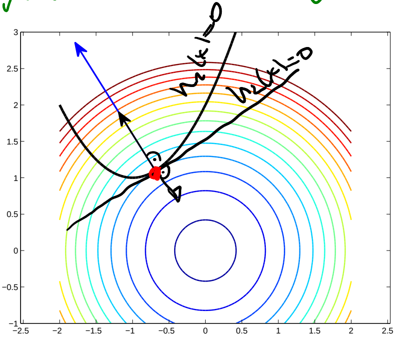


$\{\nabla h_1(x^*), \nabla h_2(x^*)\}$ ist eine linear abhängige Menge von Vektoren. Die LICQ gilt bei x^* nicht!

Mehrere Gleichungsbedingungen

$$\nabla f(x^*) + \lambda_1 \nabla h_1(x^*) + \underbrace{\lambda_2 \nabla h_2(x^*)}_{\text{linear abhängig}} = 0$$ möglich?

Ja, eine Wahl von λ_1, λ_2 ist möglich, und es gibt sogar unendlich viele Möglichkeiten!



$\begin{pmatrix}\lambda_1 \\ \lambda_2\end{pmatrix}$ erfüllt ein LGS:

$$\begin{bmatrix} | & | \\ \nabla h_1(x^*) & \nabla h_2(x^*) \\ | & | \end{bmatrix} \begin{pmatrix}\lambda_1 \\ \lambda_2\end{pmatrix} = -\nabla f(x^*)$$

Die Lösungsmenge ist ein 1D Raum.

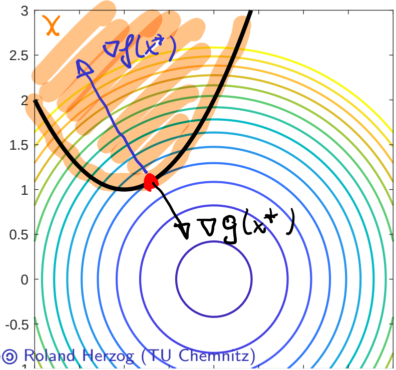
$\{\nabla g(x^*)\}$ ist linear unabhängig, d.h. die LICQ ist bei x^* erfüllt.

aktiv bei x^* !

Ungleichungsbedingung

$$\text{Minimiere} \quad f(x) := x_1^2 + x_2^2, \quad x = (x_1, x_2) \in \mathbb{R}^2$$

$$\text{sodass} \quad g(x) := -\left[x_2 - (x_1 + 1)^2 - 1\right] \le 0$$



$\mathcal{L}(x,\mu) = f(x) + \mu\, g(x)$

$\nabla_x \mathcal{L}(x,\mu) = \nabla f(x) + \mu \nabla g(x) \boxed{= 0}$

Aber: μ muss ≥ 0 sein!

① Das unterscheidet Ungleichungs- von Gleichungs-NB!

$\{\nabla h(x^*)\}$ ist linear unabhängig, also gilt LICQ bei x^*

$\nabla g(x^*)$ gehört nicht dazu, da $g(x^*) < 0$ ist

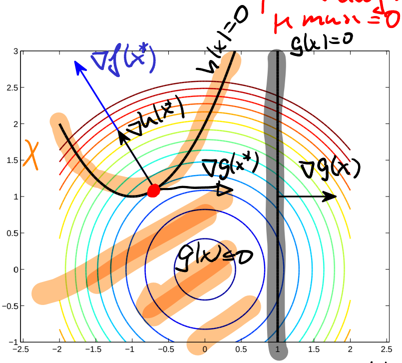
Inaktive Ungleichung

$\mathcal{L}(x,\mu,\lambda) = f(x) + \mu g(x) + \lambda h(x)$

$\nabla_x \mathcal{L}(x,\mu,\lambda) = \nabla f(x) + \mu \nabla g(x) + \lambda \nabla h(x) \boxed{= 0}$

darf nicht verwendet werden!

μ muss $= 0$ sein!



② Zu inaktiven Ungleichungen gehört $\mu_i = 0$!

$\boxed{g(x^*) < 0}$

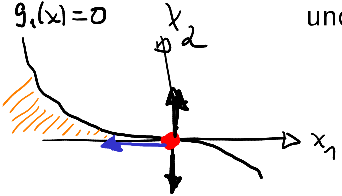
unwichtige (inaktive) Ungleichungsbedingung

Beispiel ohne Multiplikatoren

$$\overbrace{\text{Minimiere}\quad -x_1}^{\text{Maximiere } +x_1}, \quad x = (x_1, x_2) \in \mathbb{R}^2 \qquad \nabla f(x) = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$$

$$\text{sodass}\quad g_1(x) := x_2 + x_1^3 \leqslant 0$$

$$\text{und}\quad g_2(x) := -x_2 \leqslant 0.$$



$x^* = (0,0)$ $\quad \nabla g_1(x) = \begin{pmatrix} 3x_1^2 \\ 1 \end{pmatrix}, \nabla g_1(x^*) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

$$\nabla g_2(x) = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \nabla g_2(x^*) = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

Existieren Lagrangesche Multipl.? LICQ gilt bei x^* nicht.

$$\nabla f(x^*) + \mu_1 \nabla g_1(x^*) + \mu_2 \nabla g_2(x^*) = 0 \quad \text{möglich?}$$

$$\begin{pmatrix} -1 \\ 0 \end{pmatrix} + \mu_1 \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \mu_2 \begin{pmatrix} 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \text{unmöglich!}$$

Es existieren keine Multiplikatoren!

Matlab-Demonstration

`Aufgabe_ohne_Multiplikatoren.m`

Quizfrage

Was könnten wir tun, wenn wir bei einer Aufgabe Schwierigkeiten mit der Existenz von Lagrange-Multiplikatoren feststellen?

1 **A** die Zielfunktion verändern

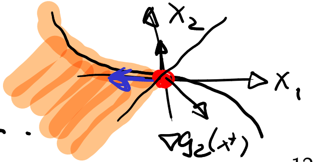
6 **B** Nebenbedingungen verändern

2 **C** Nebenbedingungen weglassen

0 **D** die Zielfunktion mit (-1) multiplizieren

Dadurch reduzieren sich die für die Linearkombi. zur Verfügung stehenden Vektoren, und es existieren „erst recht" keine Multiplikatoren.

$g_2(x) = x_1 - x_2 \le 0$



SQP-Verfahren in fmincon

(vgl. Quasi-Newton-Verfahren, Kap. 1)

Minimiere $\quad f(x), \quad x \in \mathbb{R}^n$

sodass $\begin{cases} c_{\text{eq}}(x) = 0 \\ c_{\text{ineq}}(x) \leqslant 0 \\ A_{\text{eq}}\, x = b_{\text{eq}} \\ A_{\text{ineq}}\, x \leqslant b_{\text{ineq}} \end{cases}$

sowie $\quad \ell \leqslant x \leqslant u$

(QP = quadr. Programm) $(d \in \mathbb{R}^n)$

$$\boxed{\begin{aligned} & f(x^{(k)}) + \nabla f(x^{(k)})^T d + \tfrac{1}{2} d^T B^{(k)} d \\ & c_{eq}(x^{(k)}) + c_{eq}'(x^{(k)})\, d = 0 \\ & c_{ineq}(x^{(k)}) + c_{ineq}'(x^{(k)})\, d \le 0 \\ & A_{eq} x^{(k)} + A_{eq}\, d = b_{eq} \\ & A_{ineq} x^{(k)} + A_{ineq}\, d \le b_{ineq} \\ & \ell \le x^{(k)} + d \le u \end{aligned}}$$

$B^{(k)}$ approximiert die Hessematrix der Lagrangefunktion

SQP-Verfahren in `fmincon`

- $B^{(k)}$ ist ein Ersatz für die Hessematrix der Lagrange-Funktion mit Hilfe von Quasi-Newton-Updates
- Mit Hilfe einer Liniensuche wird eine geeignete Schrittweite $\alpha^{(k)}$ bestimmt und $x^{(k+1)} := x^{(k)} + \alpha^{(k)} d^{(k)}$ gesetzt.
- Die Liniensuchfunktion berücksichtigt neben der Zielfunktion auch die Nebenbedingungen.

Zeit für Ihre Fragen

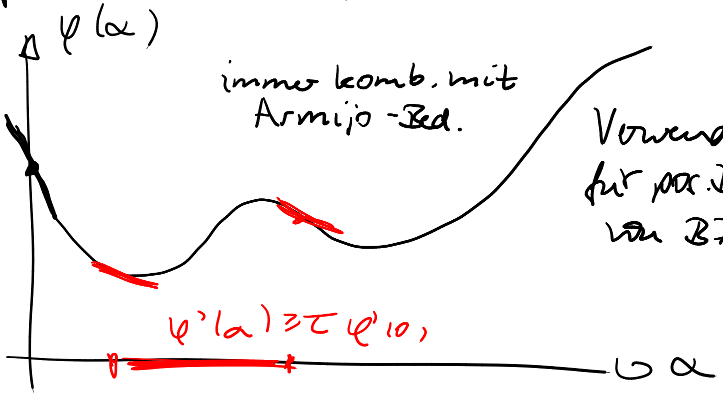
Was sind Ihre Fragen zu den Themen der Woche?

$\longrightarrow$ Benutzen Sie den Chat.

Fragen und Antworten 1

$\varphi'(\alpha) \geq \tau \varphi'(0)$

Wolfe-Liniensuche $\quad \varphi(\alpha) = f(x^{(k)} + \alpha d^{(k)})$



Verwendung:
für pos. Def.
von BFGS

Wo ist $\varphi'(\alpha) = \tau \underbrace{\varphi'(0)}_{<0}$?

Fragen und Antworten 2