Übungsblatt № 12

Gegeben ist der folgende Algorithmus:

i) Sei $\boldsymbol{x}_0 \in \mathbb{R}^n$ und $\mathrm{H}_0 \in \mathrm{GL}(n, \mathbb{R})$ symmetrisch

ii) Falls $\nabla f(\boldsymbol{x}_k) = \boldsymbol{0}$ brechen wir ab

iii) Berechne $\boldsymbol{d}_k$ als Lösung von $\mathrm{H}_k \boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)$

iv) Setze $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{d}_k$, $\boldsymbol{s}_k = \boldsymbol{d}_k$, $\boldsymbol{y}_k = \nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k)$ und

$$\mathrm{H}_{k+1} = \mathrm{H}_k + \frac{(\boldsymbol{y}_k - \mathrm{H}_k \boldsymbol{s}_k)(\boldsymbol{y}_k - \mathrm{H}_k \boldsymbol{s}_k)^{\mathrm{t}}}{(\boldsymbol{y}_k - \mathrm{H}_k \boldsymbol{s}_k)^{\mathrm{t}} \boldsymbol{s}_k}$$

v) Gehe zu i)

Aufgabe 130: Lokales Quasi-Newton Verfahren mit SR1-Update

Gegeben sei das quadratische Optimierungsproblem mit $\frac{1}{2}\boldsymbol{x}^{\mathrm{t}}\mathrm{Q}\boldsymbol{x} + \boldsymbol{c}^{\mathrm{t}}\boldsymbol{x}$

$$\mathrm{Q} = \mathrm{diag}(2, 3, 4) \qquad \boldsymbol{c} = \begin{bmatrix} 8 \\ 9 \\ 8 \end{bmatrix}$$

Starten Sie mit $\boldsymbol{x}_0 = \boldsymbol{0}$ und $\mathrm{H}_0 = \mathrm{I}$.

a) Wenden Sie den oben beschriebenen Algorithmus an

b) Wenden Sie eine Variante des oben beschriebenen Algorithmus an, bei dem Sie die Schrittweite nach der Minimierungsregel bestimmen

Für a) ergibt sich nun der folgende Algorithmus

```
from numpy import diag, array, outer
from numpy.linalg import norm, inv

__MAX_STEPS = 1e4
__EPS = 1e-9
Q = diag([2,3,4])
c = array([8,9,8])
f = lambda x : 0.5 * x.T @ Q @ x + c.T @ x
fp = lambda x: Q @ x + c

def Hk(y, H, s)
    x = y - H @ s
    denom = x.T @ s
    return H + 1.0 / denom  * outer(x,x)

def quasi_newton(H0, x0):
    x = x0
    H = H0
    counter = 0
    while counter <= __MAX_STEPS and norm(fp(x)) >= __EPS:
        d = -inv(H) @ fp(x)
        xprev = x
        x = x + d
        s = x - xprev
        y = fp(x) - fp(xprev)
        H = Hk(y,H,s)
        counter += 1
    return x
```

Als Minimierungsregel für b) verwenden wir

$$t = -\frac{-\nabla f(\boldsymbol{x})^{\mathrm{t}}\boldsymbol{d}}{\boldsymbol{d}^{\mathrm{t}}\mathrm{Q}\boldsymbol{d}}$$

Beide Verfahren liefern

$$\boldsymbol{x}_{\min} = \begin{bmatrix} -4 \\ -3 \\ -2 \end{bmatrix} \qquad f(\boldsymbol{x}_{\min}) = -37.5$$

Allerdings braucht die Variante mit konstanter Schrittweite 3 Iterationen, bis die Abbruchbedingung von $\|\nabla f(\boldsymbol{x}_k)\| < 10^{-9}$ erreicht ist, während mit der exakten Schrittweite sogar nur 2 Iterationen durchgeführt werden.