Chapter 19

Minimization problems

19.1 Unconstrained minimization

The minimization problem

Find $\bar{x} \in D \subset \mathbb{R}^n$, such that

$$f(\bar{x}) \leq f(x), \quad \forall x \in D, \tag{19.1}$$

with $D \subset \mathbb{R}^n$ the *search space*, $\bar{x} \in D$ the *optimal solution*, and $f : D \to \mathbb{R}$ the *objective function* (or *cost function*).

A *stationary point*, or *critical point*, $\hat{x} \in D$ is a point for which the gradient of the objective function is zero, that is,

$$\nabla f(\hat{x}) = 0, \tag{19.2}$$

and we refer to $x^* \in D$ as a *local minimum* if there exists $\delta > 0$, such that,

$$f(x^*) \leq f(x), \quad \forall x : \|x - x^*\| \leq \delta. \tag{19.3}$$

If the minimization problem is *convex*, an interior local minimum is a global minimum, where in a convex minimization problem tthe search space is convex, i.e.

$$(1-t)x + ty \in D, \tag{19.4}$$

and the objective function is convex, i.e.

$$(1-t)f(x) + tf(y) \leq f((1-t)x + ty), \tag{19.5}$$

for all $x, y \in D$ and $t \in [0, 1]$.

Gradient descent method

The *level set* of the function $f : D \to \mathbb{R}^n$ is defined as

$$L_c(f) = \{x \in D : f(x) = c\}, \tag{19.6}$$

where we note that $L_c(f)$ represents a level curve in $\mathbb{R}^2$, a level surface in $\mathbb{R}^3$, and more generally a hypersurface of dimension $n-1$ in $\mathbb{R}^n$.

Theorem 18. *If $f \in \mathcal{C}^1(D)$, then the gradient $\nabla f(x)$ is orthogonal to the level set $L_c(f)$ at $x \in D$.*

The *gradient descent method* is an iterative method that compute approximations to a local minimum of (19.1), by searching for the next iterate in a direction orthogonal to the level set $L_c(f)$ in which the objective function decreases, the direction of *steepest descent*, with a *step length* α.

Algorithm 18: Method of steepest descent

```
Start from x^(0)                              ▷ initial approximation
for k = 1, 2, ... do
  | x^(k+1) = x^(k) − α^(k) ∇f(x^(k))         ▷ Step with length α^(k)
end
```

Newton's method

For $f \in \mathcal{C}^1(D)$ we know by Taylor's formula that,

$$f(x) \approx f(y) + \nabla f(y) \cdot (x-y) + \frac{1}{2}(x-y)^T Hf(y)(x-y), \tag{19.7}$$

for $x, y \in D$, with the symmetric *Hessian* defined by

$$Hf = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_n} \end{bmatrix}. \tag{19.8}$$

Newton's method to find a local minimum in the form of a stationary point is based on (19.7) with $x = x^{(k+1)}$, $y = x^{(k)}$ and $\Delta x = x^{(k+1)} - x^{(k)}$, for which we seek the stationary point, by

$$\begin{aligned} 0 &= \frac{d}{d(\Delta x)}\left(f(x^{(k)}) + \nabla f(x^{(k)}) \cdot \Delta x + \frac{1}{2}\Delta x^T Hf(x^{(k)})\Delta x\right) \\ &= \nabla f(x^{(k)}) + Hf(x^{(k)})\Delta x, \end{aligned}$$

which gives Newton's method as an iterative method with increment

$$\Delta x = -(Hf(x^{(k)}))^{-1}\nabla f(x^{(k)}). \tag{19.9}$$

Algorithm 19: Newton's method for finding a stationary point

Start from $x^{(0)}$ ▷ `initial approximation`
for $k = 1, 2, ...$ **do**
  $Hf(x^{(k)})\Delta x = -\nabla f(x^{(k)})$ ▷ `Solve linear system for` Δx
  $x^{(k+1)} = x^{(k)} + \Delta x$ ▷ `Update approximation`
end

19.2 Linear system of equations

We now revisit the problem to find a solution $x \in \mathbb{R}^n$ to the system of linear equations

$$Ax = b, \tag{19.10}$$

with $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$, with $m \geq n$.

Least square method

The linear system of equations $Ax = b$ can be solved by minimzation algorithms, for example, in the form a least squares problem,

$$\min_{x\in D} f(x), \quad f(x) = \|Ax - b\|^2, \tag{19.11}$$

with $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$, and $m \geq n$. The gradient is computed as,

$$\begin{aligned}
\nabla f(x) &= \nabla(\|Ax-b\|^2) = \nabla((Ax)^T Ax - (Ax)^T b - b^T Ax + b^T b) \\
&= \nabla(x^T A^T Ax - 2x^T A^T b + b^T b) = A^T Ax + x^T A^T A - 2A^T b \\
&= A^T Ax + A^T Ax - 2A^T b = 2A^T(Ax - b),
\end{aligned}$$

which gives the following gradient descent method

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} 2A^T(Ax^{(k)} - b) \tag{19.12}$$

Quadratic forms

We consider the minimization problem,

$$\min_{x \in D} f(x), \tag{19.13}$$

where $f(x)$ is the quadratic form

$$f(x) = x^T Ax - b^T x + c, \tag{19.14}$$

with $x \in \mathbb{R}^n$, $A \in \mathbb{R}^{n \times n}$ and $n \in \mathbb{R}$, with a stationary point given by

$$0 = \nabla f(x) = \nabla(\frac{1}{2}x^T Ax - b^T x + c) = \frac{1}{2}Ax + \frac{1}{2}A^T x - b, \tag{19.15}$$

which in the case A is a symmetric matrix corresponds to the linear system of equations,

$$Ax = b. \tag{19.16}$$

To prove that the solution $x = A^{-1}b$ is the solution of the minimization problem, study the error $e = u - y$, with $y \in D$, for which we have that

$$\begin{aligned} f(x+e) &= \frac{1}{2}(x+e)^T A(x+e) - b^T(x+e) + c \\ &= \frac{1}{2}x^T Ax + e^T Ax + \frac{1}{2}e^T Ae - b^T x - b^T e + c \\ &= (\frac{1}{2}x^T Ax - b^T x + c) + \frac{1}{2}e^T Ae + (e^T b - b^T e) \\ &= f(x) + \frac{1}{2}e^T Ae. \end{aligned}$$

We find that if A is a positive definite matrix $x = A^{-1}b$ is a global minimum, and thus any system of linear equations with a symmetric positive definite matrix may be reformulated as minimization of the associated quadratic form.

If A is not positive definite, it may be negative definite with minimum being $-\infty$, singular with non unique minima, or else the quadratic form $f(x)$ has a *saddle-point*.

Gradient descent method

To solve the minimization problem for a quadratic form, we may use a gradient descent method for which the gradient gives the residual,

$$-\nabla f(x^{(k)}) = b - Ax^{(k)} = r^{(k)}, \tag{19.17}$$

that is,

$$x^{(k+1)} = x^{(k)} - \alpha \nabla f(x^{(k)}) = x^{(k)} + \alpha r^{(k)}. \tag{19.18}$$

To choose a step length α that minimizes $x^{(k+1)}$, we compute the derivative

$$\frac{d}{d\alpha} f(x^{(k+1)}) = \nabla f(x^{(k+1)})^T \frac{d}{d\alpha} x^{(k+1)} = \nabla f(x^{(k+1)})^T r^{(k)} = -(r^{(k+1)})^T r^{(k)},$$

which gives that α should be chosen such that the successive residuals are orthogonal, that is

$$(r^{(k+1)})^T r^{(k)} = 0, \tag{19.19}$$

which gives that

$$\begin{aligned}
(r^{(k+1)})^T r^{(k)} &= 0 \\
(b - Ax^{(k+1)})^T r^{(k)} &= 0 \\
(b - A(x^{(k)} + \alpha r^{(k)}))^T r^{(k)} &= 0 \\
(b - A(x^{(k)})^T r^{(k)} - \alpha (Ar^{(k)})^T r^{(k)} &= 0 \\
(r^{(k)})^T r^{(k)} &= \alpha (Ar^{(k)})^T r^{(k)}
\end{aligned}$$

so that

$$\alpha = \frac{(r^{(k)})^T r^{(k)}}{(Ar^{(k)})^T r^{(k)}}. \tag{19.20}$$

Algorithm 20: Steepest descent method for $Ax = b$

Start from $x^{(0)}$ ▷ `initial approximation`

for $k = 1, 2, ...$ **do**

  $r^{(k)} = b - Ax^{(k)}$ ▷ `Compute residual` $r^{(k)}$

  $\alpha = (r^{(k)})^T r^{(k)} / (Ar^{(k)})^T r^{(k)}$ ▷ `Compute step length` $\alpha^{(k)}$

  $x^{(k+1)} = x^{(k)} + \alpha^{(k)} r^{(k)}$ ▷ `Step with length` $\alpha^{(k)}$

end

Conjugate gradient method revisited

We now revisit the conjugate gradient (CG) method in the form of a method for solving the minimization of the quadratic form corresponding to a linear system of equations with a symmetric positive definite matrix.

The idea is to formulate a search method,

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{(k)}, \tag{19.21}$$

with a set of orthogonal search directions $\{d^{(k)}\}_{k=0}^{n-1}$, where the step length $\alpha^{(k)}$ is determined by the condition that $e^{(k+1)} = x - x^{(k+1)}$ should be A-*orthogonal*, or *conjugate*, to $d^{(k)}$, thus

$$\begin{aligned}(d^{(k)})^T A e^{(k+1)} &= 0 \\ (d^{(k)})^T A (e^{(k)} - \alpha^{(k)} d^{(k)}) &= 0,\end{aligned}$$

so that

$$\alpha = \frac{(d^{(k)})^T A e^{(k)}}{(d^{(k)})^T A d^{(k)}} = \frac{(d^{(k)})^T r^{(k)}}{(d^{(k)})^T A d^{(k)}}. \tag{19.22}$$

To construct the orthogonal search directions $d^{(k)}$ we can use the Gram-Schmidt iteration, whereas if we choose the search direction to be the residual we get the steepest descent method.

19.3 Constrained minimization

The constrained minimization problem

We now consider the *constrained minimization problem*,

$$\min_{x \in D} f(x) \tag{19.23}$$

$$g(x) = c, \tag{19.24}$$

with the objective function $f : D \to \mathbb{R}$, and the *constraints* $g : D \to \mathbb{R}^m$, with $x \in D \subset \mathbb{R}^n$ and $c \in \mathbb{R}^m$.

We define the *Lagrangian* $\mathcal{L} : \mathbb{R}^{n+m} \to \mathbb{R}$, as

$$\mathcal{L}(x, \lambda) = f(x) - \lambda \cdot (g(x) - c), \tag{19.25}$$

with the *dual variables*, or *Lagrangian multipliers*, $\lambda \in \mathbb{R}^m$, from which we obtain the optimality conditions,

$$\nabla_x \mathcal{L}(x, \lambda) = \nabla f(x) - \lambda \cdot \nabla g(x) = 0, \tag{19.26}$$

$$\nabla_\lambda \mathcal{L}(x, \lambda) = g(x) - c = 0, \tag{19.27}$$

that is, $n + m$ equations from which we can solve for the unknown variables $(x, \lambda) \in \mathbb{R}^{n+m}$.

Example in $\mathbb{R}^2$

For $f : \mathbb{R}^2 \to \mathbb{R}$, $g : \mathbb{R}^2 \to \mathbb{R}$ and $c = 0$, the lagrangian takes the form

$$\mathcal{L}(x, \lambda) = f(x) - \lambda g(x), \tag{19.28}$$

with the optimality conditions

$$\nabla_x \mathcal{L}(x, \lambda) = \nabla f(x) - \lambda \nabla g(x) = 0, \tag{19.29}$$
$$\nabla_\lambda \mathcal{L}(x, \lambda) = g(x) = 0, \tag{19.30}$$

so that $\nabla f = \lambda \nabla g(x)$, which corresponds to the curve defined by the constraint $g(x) = 0$ being parallel to a level curve of $f(x)$ in $\bar{x} \in \mathbb{R}^2$, the solution to the constrained minimization problem.

Optimal control

We now consider the *constrained minimization problem*,

$$\min_{x \in D} c^T x \tag{19.31}$$
$$Ax = b, \tag{19.32}$$

with $x, b, c \in \mathbb{R}^n$, and $A \in \mathbb{R}^{n \times n}$.

We define the *Lagrangian* $\mathcal{L} : \mathbb{R}^{2n} \to \mathbb{R}$, as

$$\mathcal{L}(x, \lambda) = c^T x - \lambda^T (Ax - b), \tag{19.33}$$

with $\lambda \in \mathbb{R}^n$, from which we obtain the optimality conditions,

$$\nabla_x \mathcal{L}(x, \lambda) = c + A^T \lambda = 0, \tag{19.34}$$
$$\nabla_\lambda \mathcal{L}(x, \lambda) = Ax - b = 0. \tag{19.35}$$