

G V V Sharma*

CONTENTS

1	Constrained Optimization	1
2	Convex Function	2
3	Gradient Descent	2
4	Lagrange Multipliers	3
5	Quadratic Programming	3

Abstract—This book provides an introduction to optimization based on the NCERT textbooks from Class 6-12. Links to sample Python codes are available in the text.

Download python codes using

svn co <https://github.com/gadepall/school/trunk/ncert/optimization/codes>

1 CONSTRAINED OPTIMIZATION

- Express the problem of finding the distance of the point $\mathbf{P} = \begin{pmatrix} 3 \\ -5 \end{pmatrix}$ from the line

$$L : (3 \quad -4)\mathbf{x} = 26 \quad (1.1.1)$$

as an optimization problem.

Solution: The given problem can be expressed as

$$\min_{\mathbf{x}} g(\mathbf{x}) = \|\mathbf{x} - \mathbf{P}\|^2 \quad (1.1.2)$$

$$\text{s.t. } \mathbf{n}^T \mathbf{x} = c \quad (1.1.3)$$

where

$$\mathbf{n} = \begin{pmatrix} 3 \\ -4 \end{pmatrix} \quad (1.1.4)$$

$$c = 26 \quad (1.1.5)$$

- Explain Problem 1.1 through a plot and find a graphical solution.

- Solve (1.1.2) using cvxpy.

Solution:

codes/line_dist_cvx.py

- Convert (1.1.2) to an *unconstrained* optimization problem.

Solution: L in (1.1.1) can be expressed in terms of the direction vector \mathbf{m} as

$$\mathbf{x} = \mathbf{A} + \lambda \mathbf{m}, \quad (1.4.1)$$

where \mathbf{A} is any point on the line and

$$\mathbf{m}^T \mathbf{n} = 0 \quad (1.4.2)$$

Substituting (1.4.1) in (1.1.2), an unconstrained optimization problem

$$\min_{\lambda} f(\lambda) = \|\mathbf{A} + \lambda \mathbf{m} - \mathbf{P}\|^2 \quad (1.4.3)$$

is obtained.

- Solve (1.4.3).

Solution:

$$f(\lambda) = (\lambda \mathbf{m} + \mathbf{A} - \mathbf{P})^T (\lambda \mathbf{m} + \mathbf{A} - \mathbf{P}) \quad (1.5.1)$$

$$= \lambda^2 \|\mathbf{m}\|^2 + 2\lambda \mathbf{m}^T (\mathbf{A} - \mathbf{P}) + \|\mathbf{A} - \mathbf{P}\|^2 \quad (1.5.2)$$

$$\because f^{(2)}(\lambda) = 2\|\mathbf{m}\|^2 > 0 \quad (1.5.3)$$

the minimum value of $f(\lambda)$ is obtained when

$$f^{(1)}(\lambda) = 2\lambda \|\mathbf{m}\|^2 + 2\mathbf{m}^T (\mathbf{A} - \mathbf{P}) = 0 \quad (1.5.4)$$

$$\Rightarrow \lambda_{\min} = -\frac{\mathbf{m}^T (\mathbf{A} - \mathbf{P})}{\|\mathbf{m}\|^2} \quad (1.5.5)$$

Choosing \mathbf{A} such that

$$\mathbf{m}^T (\mathbf{A} - \mathbf{P}) = 0, \quad (1.5.6)$$

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in. All content in this manual is released under GNU GPL. Free and open source.

substituting in (1.5.5),

$$\lambda_{\min} = 0 \quad \text{and} \quad (1.5.7)$$

$$\mathbf{A} - \mathbf{P} = \mu \mathbf{n} \quad (1.5.8)$$

for some constant μ . (1.5.8) is a consequence of (1.4.2) and (1.5.6). Also, from (1.5.8),

$$\mathbf{n}^T (\mathbf{A} - \mathbf{P}) = \mu \|\mathbf{n}\|^2 \quad (1.5.9)$$

$$\Rightarrow \mu = \frac{\mathbf{n}^T \mathbf{A} - \mathbf{n}^T \mathbf{P}}{\|\mathbf{n}\|^2} = \frac{c - \mathbf{n}^T \mathbf{P}}{\|\mathbf{n}\|^2} \quad (1.5.10)$$

from (1.1.3). Substituting $\lambda_{\min} = 0$ in (1.4.3),

$$\min_{\lambda} f(\lambda) = \|\mathbf{A} - \mathbf{P}\|^2 = \mu^2 \|\mathbf{n}\|^2 \quad (1.5.11)$$

upon substituting from (1.5.8). The distance between \mathbf{P} and L is then obtained from (1.5.11) as

$$\|\mathbf{A} - \mathbf{P}\| = |\mu| \|\mathbf{n}\| \quad (1.5.12)$$

$$= \frac{|\mathbf{n}^T \mathbf{P} - c|}{\|\mathbf{n}\|} \quad (1.5.13)$$

after substituting for μ from (1.5.10).

2 CONVEX FUNCTION

1. The following python script plots

$$f(\lambda) = a\lambda^2 + b\lambda + d \quad (2.1.1)$$

for

$$a = \|\mathbf{m}\|^2 > 0 \quad (2.1.2)$$

$$b = \mathbf{m}^T (\mathbf{A} - \mathbf{P}) \quad (2.1.3)$$

$$c = \|\mathbf{A} - \mathbf{P}\|^2 \quad (2.1.4)$$

where \mathbf{A} is the intercept of the line L in (1.1.1) on the x-axis and the points

$$\mathbf{U} = \begin{pmatrix} \lambda_1 \\ f(\lambda_1) \end{pmatrix}, \mathbf{V} = \begin{pmatrix} \lambda_2 \\ f(\lambda_2) \end{pmatrix} \quad (2.1.5)$$

$$\mathbf{X} = \begin{pmatrix} t\lambda_1 + (1-t)\lambda_2 \\ f[t\lambda_1 + (1-t)\lambda_2] \end{pmatrix}, \quad (2.1.6)$$

$$\mathbf{Y} = \begin{pmatrix} t\lambda_1 + (1-t)\lambda_2 \\ tf(\lambda_1) + (1-t)f(\lambda_2) \end{pmatrix} \quad (2.1.7)$$

for

$$\lambda_1 = -3, \lambda_2 = 4, t = 0.3 \quad (2.1.8)$$

in Fig. 2.1. Geometrically, this means that any point \mathbf{Y} between the points \mathbf{U}, \mathbf{V} on the line UV is always above the point \mathbf{X} on the curve

$f(\lambda)$. Such a function f is defined to be *convex* function

codes/optimization/1.2.py

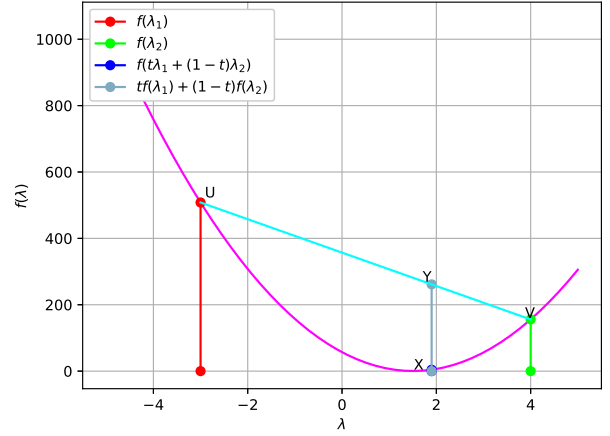


Fig. 2.1: $f(\lambda)$ versus λ

2. Show that

$$f[t\lambda_1 + (1-t)\lambda_2] \leq tf(\lambda_1) + (1-t)f(\lambda_2) \quad (2.2.1)$$

for $0 < t < 1$. This is true for any convex function.

3. Show that

$$(2.2.1) \Rightarrow f^{(2)}(\lambda) > 0 \quad (2.3.1)$$

3 GRADIENT DESCENT

1. Find a numerical solution for (2.1.1)

Solution: A numerical solution for (2.1.1) is obtained as

$$\lambda_{n+1} = \lambda_n - \mu f'(\lambda_n) \quad (3.1.1)$$

$$= \lambda_n - \mu (2a\lambda_n + b) \quad (3.1.2)$$

where λ_0 is an initial guess.

2. Write a program to implement (3.1.2).

Solution: Download and execute

codes/optimization/gd.py

3. Find a closed form solution for (3.1.2) using the one sided Z transform.

4. Find the condition for which (3.1.2) converges, i.e.

$$\lim_{n \rightarrow \infty} |\lambda_{n+1} - \lambda_n| = 0 \quad (3.4.1)$$

4 LAGRANGE MULTIPLIERS

1. Find

$$\min_{\mathbf{x}} g(\mathbf{x}) = \|\mathbf{x} - \mathbf{P}\|^2 = r^2 \quad (4.1.1)$$

$$\text{s.t. } h(\mathbf{x}) = \mathbf{n}^T \mathbf{x} - c = 0 \quad (4.1.2)$$

by plotting the circles $g(\mathbf{x})$ for different values of r along with the line $g(\mathbf{x})$.

Solution: The following code plots Fig. 4.1

codes/optimization/concinc.py

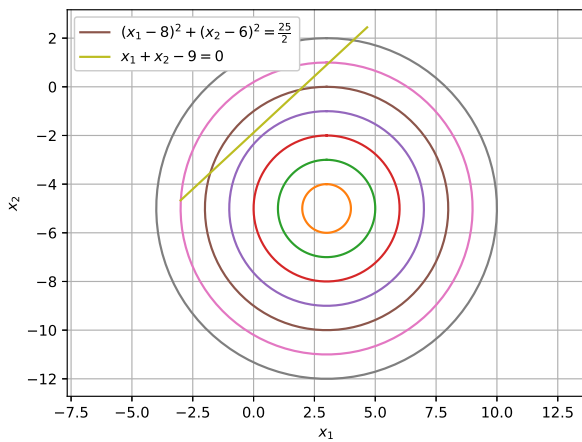


Fig. 4.1: Finding $\min_{\mathbf{x}} g(\mathbf{x})$

2. By solving the quadratic equation obtained from (4.1.1), show that

$$\min_{\mathbf{x}} r = \frac{3}{5} \quad (4.2.1)$$

and find $\mathbf{x} = \mathbf{Q}$ that minimizes r . By labeling \mathbf{Q} in Fig. 4.1, show that \mathbf{Q} is the point of contact of the line L with the circle of minimum radius $r = \frac{3}{5}$.

3. Show that

$$\nabla h(\mathbf{x}) = \begin{pmatrix} 3 \\ -4 \end{pmatrix} = \mathbf{n} \quad (4.3.1)$$

where

$$\nabla = \begin{pmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \end{pmatrix} \quad (4.3.2)$$

4. Show that

$$\nabla g(\mathbf{x}) = 2 \left\{ \mathbf{x} - \begin{pmatrix} 3 \\ -5 \end{pmatrix} \right\} = 2 \{\mathbf{x} - \mathbf{P}\} \quad (4.4.1)$$

5. From Fig. 4.1, show that

$$\nabla g(\mathbf{Q}) = \lambda \nabla h(\mathbf{Q}), \quad (4.5.1)$$

6. Use (4.5.1) and $\mathbf{h}(\mathbf{Q}) = 0$ from (4.1.2) to obtain \mathbf{Q} .

7. Define

$$C(\mathbf{x}, \lambda) = g(\mathbf{x}) - \lambda h(\mathbf{x}) \quad (4.7.1)$$

and show that \mathbf{Q} can also be obtained by solving the equations

$$\nabla C(\mathbf{x}, \lambda) = 0. \quad (4.7.2)$$

What is the sign of λ ? C is known as the Lagrangian and the above technique is known as the Method of Lagrange Multipliers.

Solution:

codes/optimization/lagmul.py

8. Obtain \mathbf{Q} using gradient descent.

5 QUADRATIC PROGRAMMING

1. Find the point on the curve

$$x^2 = 2y \quad (5.1.1)$$

nearest to the point

$$\mathbf{P} = \begin{pmatrix} 0 \\ 5 \end{pmatrix}. \quad (5.1.2)$$

by drawing a figure.

Solution: The following code plots Fig.

2. Frame Problem 5.1.1 as an optimization problem.

Solution: The given problem can be expressed as

$$\min_{\mathbf{x}} \|\mathbf{x} - \mathbf{P}\|^2 \quad (5.2.1)$$

$$\text{s.t. } \mathbf{x}^T \mathbf{V} \mathbf{x} + \mathbf{u}^T \mathbf{x} = 0 \quad (5.2.2)$$

where

$$\mathbf{V} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \mathbf{u} = -\begin{pmatrix} 0 \\ 2 \end{pmatrix} \quad (5.2.3)$$

3. Show that the constraint in 5.2.1 is nonconvex.

4. Show that the following *relaxation* makes (5.2.1) a convex optimization problem.

$$\min_{\mathbf{x}} \|\mathbf{x} - \mathbf{P}\|^2 \quad (5.4.1)$$

$$\text{s.t. } \mathbf{x}^T \mathbf{V} \mathbf{x} + \mathbf{u}^T \mathbf{x} \leq 0 \quad (5.4.2)$$

5. Solve (5.4.1) using cvxpy.
6. Solve (5.4.1) using the method of Lagrange multipliers.
7. Solve (5.4.1) using gradient descent.