Assignment 1

魏家博 (Chia-Po Wei)

Department of Electrical Engineering

National Sun Yat-sen University

Outline

Assignment 1a: Linear Equations

Assignment 1b: Least Squares Problem

Assignment 1a: Linear Equations

- Open assignment_1a.ipynb
- Implement the function `solve_linear_equation()` that achieves the following requirements
  - Input: Array $\mathbf{A}$ and array $\mathbf{b}$
  - Output: 1 if $\mathbf{Ax} = \mathbf{b}$ has only one solution
  - 0 if $\mathbf{Ax} = \mathbf{b}$ has infinitely many solutions
  - -1 if $\mathbf{Ax} = \mathbf{b}$ has no solutions
- You will use the two propositions in the next page.

Useful Propositions

Proposition 1. The following three statements are equivalent.

1. The linear system $\mathbf{Ax} = \mathbf{b}$ is consistent.
2. The vector $\mathbf{b}$ can be expressed as a linear combination of $\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_n$.
3. $\operatorname{rank}(\mathbf{A}) = \operatorname{rank}([\mathbf{A} \quad \mathbf{b}])$

Proposition 2. Suppose $\mathbf{A} \in \mathbb{R}^{m\times n}$ and $\mathbf{b} \in \mathbb{R}^m$. The following three statements are equivalent.

1. The linear system $\mathbf{Ax} = \mathbf{b}$ has at most one solution for every $\mathbf{b} \in \mathbb{R}^m$.
2. The column vectors of $\mathbf{A}$ are linearly independent.
3. $\operatorname{rank}(\mathbf{A}) = n$

Hints for Assignment 1a

1. Use np.linalg.matrix_rank() to calculate the rank of $\mathbf{A}$.
2. Use np.hstack() to create array $[\mathbf{A}, \mathbf{b}]$.
3. Use np.linalg.matrix_rank() to calculate the rank of $[\mathbf{A}, \mathbf{b}]$.
4. Use **Proposition 1** to determine whether $\mathbf{Ax} = \mathbf{b}$ is consistent.
5. If it is consistent, then use **Proposition 2** to determine whether $\mathbf{Ax} = \mathbf{b}$ has only one solution.
    - You can use A.shape[1] to get the column number of $\mathbf{A}$

Questions for Assignment 1a

1. Can we use np.linalg.solve() to determine whether $\mathbf{Ax} = \mathbf{b}$ is consistent? Explain the reason for your answer.
2. Given $\mathbf{A} \in \mathbb{R}^{m \times n}$. If $\mathrm{rank}(\mathbf{A}) = n$, can we infer that $\mathbf{Ax} = \mathbf{b}$ is consistent? Explain the reason for your answer.

Assignment 1b: Least Squares Problem

Consider

$$\min_{\mathbf{x}} f(\mathbf{x}) = \min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2^2 = \min_{\mathbf{x}} (\mathbf{Ax} - \mathbf{b})^T (\mathbf{Ax} - \mathbf{b})$$

where $\mathbf{A} = \begin{bmatrix} 1 & -2 \\ 1 & -2 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$.

1. Use the normal equation to find the optimal solution, i.e., solve

$$\mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b} \implies \begin{bmatrix} 1 & -2 \\ 1 & -2 \end{bmatrix}^T \begin{bmatrix} 1 & -2 \\ 1 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 & -2 \\ 1 & -2 \end{bmatrix}^T \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

   Find all x_1, x_2 that satisfy the above equation. In fact, there are infinitely many optimal solutions for this exercise.

Steps for Assignment 1b

2. Open assignment_1b.ipynb
3. Implement the following pseudo code on the left-hand side for gradient descent in function `show_gradient_descent()`

```
x = [−2,2]^T # initial condition
alpha = 0.02 #learning rate
max_iter = 1000
f = ||Ax − b||_2^2
for k in range(max_iter):
  print k, (x1,x2), f
        as shown in the right table
  x_prev = x
  x = x - alpha * ∇f(x)
  # ∇f(x) = 2(A^T Ax − A^T b)
  if ||x − x_prev|| ≤ 10^−8 then break
  f = ||Ax − b||_2^2
```

k	(x_1, x_2)	$f(\mathbf{x})$
0	(-2, 2)	113.00
1	(-1.4, 0.8)	41.00
2	(-1.,04, 0.08)	15.08
3	(-0.69, -0.61)	2.39
4	(-0.62, -0.77)	1.18
5	(-0.57, -0.86)	0.74
⋮	⋮	⋮
37	(-0.5, -1.0)	0.5

最佳解
印出小數點後4位

最小值
印出小數點後8位

Hints for Assignment 1b

- You can use np.dot(A, b) or A.dot(b) to perform matrix multiplication $\mathbf{Ab}$
- You can use A.T to obtain $\mathbf{A}^T$
- You can use np.linalg.norm(v) to calculate $||\mathbf{v}||_2$
- You can use x**2 to calculate $\mathbf{x}^2$
- Learn how to set the decimal precision for print(). There are two styles.
  - Old style (Python 2.7): http://interactivepython.org/runestone/static/pip/StringFormatting/interpolation.html
  - New style (Python 3.7): https://pyformat.info/

Steps for Assignment 1b

4. Implement the function `draw_gradient_descent()` to draw Figure 2 as shown below.
   - Let ax = plt.gca(). Use ax.scatter() to plot the point x of each iteration in the 3D space as shown in Figure 1 (draw the red points)
   - Draw all optimal solutions calculated in Step 1 of Assignment 1b as shown in Figure 2 (draw the blue line).

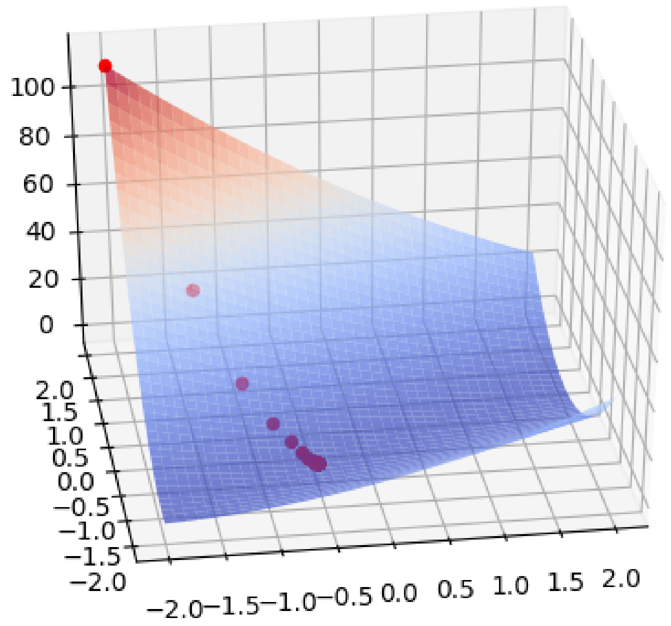

Figure 1

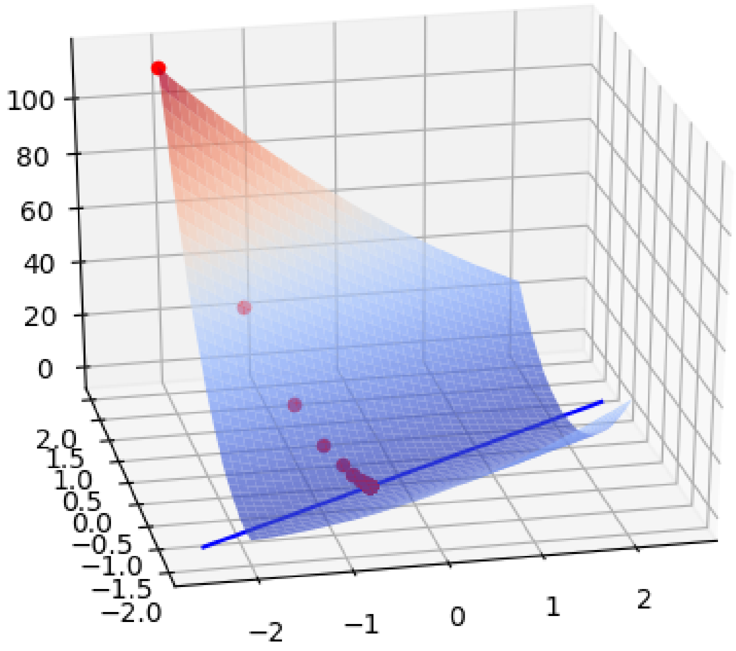

Figure 2

Questions for Assignment 1b

- Answer the following questions.
  a) Try different learning rates of alpha. How to adjust the learning rate for faster speed of convergence? 如何調整 alpha 使得所需迴圈次數愈少愈好(即較快達到收斂)？
  b) Try different initial points of x. Do different initial points give rise to different to optimal solutions? 不同初始點 x 會使演算法找到不同最佳解嗎?
  c) Does the minimum vary with the values of the initial point? 不同初始點 x 會使演算法找到不同最小值嗎?
  d) Is the minimum a local minimum or a global minimum?