Convex Optimization: Part III

CMPT 882

Feb. 4

Textbook

- S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2008.

Outline

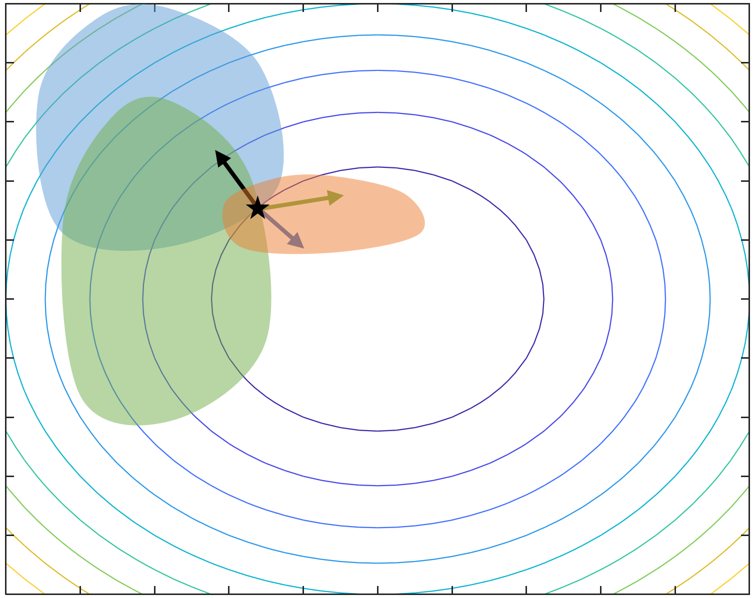
- Solving convex optimization problems
  - Solving the optimality conditions
  - Gradient methods for approximating solutions to convex optimization problems
    - Unconstrained case

Optimality Conditions for Convex Programs

- Full optimization problem:

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && g_i(x) \le 0, i = 1, \dots, n \\ & && a_j^\top x = b_j, j = 1, \dots, m \end{aligned}$$

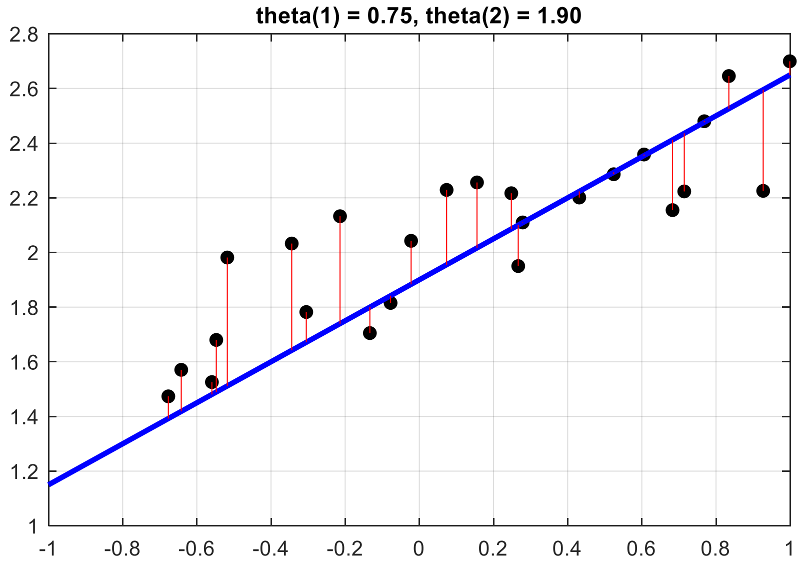
- Penalty view point:
  - Lagrangian: $L(x, \lambda) = f(x) + \sum_{i=1}^{n} \lambda_i g_i(x) + \sum_{j=1}^{m} \mu_j \left(a_j^\top x - b_j\right),\ \lambda_i \ge 0$
- Karush-Kuhn-Tucker (KKT) Conditions:
  - Stationarity $\nabla_x L(x^*, \lambda^*, \mu^*) = 0$
  - Primal feasibility: $g_i(x^*) \le 0,\ a_i^\top x^* - b_i = 0$
  - Dual feasibility: $\lambda^* \ge 0$
  - Complementary slackness: $\lambda_i^* g_i(x^*) = 0,\ i = 1, \dots, n$



- Solve above systems of equations to obtain optimum

Example: Least Squares

$$\underset{\theta}{\text{minimize}} \|X\theta - Y\|_2^2$$



- Scalar example:
  - Data: $\{x_i, y_i\}_{i=1}^n$, $x_i, y_i \in \mathbb{R}$
  - Model: $y = mx + b, m, b \in \mathbb{R}$
  - Sum of error of model: $\sum_{i=1}^n (y_i - mx_i - b)^2$
  - No constraints: allow *any* m, b
- Error in matrix form: $e_i = y_i - \begin{bmatrix} x_i & 1 \end{bmatrix} \begin{bmatrix} m \\ b \end{bmatrix}$
  - Stacking the data points: $E_i = \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}}_{Y} - \underbrace{\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix}}_{X} \underbrace{\begin{bmatrix} m \\ b \end{bmatrix}}_{\theta}$

Optimality Conditions for Convex Programs

- Full optimization problem: $$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && g_i(x) \leq 0, i = 1, \dots, n \\ & && a_j^\top x = b_j, j = 1, \dots, m \end{aligned}$$

- Penalty view point:
  - Lagrangian: $L(x, \lambda) = f(x) + \sum_{i=1}^{n} \lambda_i g_i(x) + \sum_{j=1}^{m} \mu_j \left(a_j^\top x - b_j\right),\ \lambda_i \geq 0$
- Karush-Kuhn-Tucker (KKT) Conditions:
  - Stationarity $\nabla_x L(x^*, \lambda^*, \mu^*) = 0$ $\longleftarrow$ $\boldsymbol{\nabla f(x) = 0}$
  - Primal feasibility: $g_i(x^*) \leq 0,\ a_i^\top x^* - b_i = 0$
  - Dual feasibility: $\lambda^* \geq 0$
  - Complementary slackness: $\lambda_i^* g_i(x^*) = 0,\ i = 1, \dots, n$

- Solve above systems of equations to obtain optimum

Example: Least Squares

$$\underset{\theta}{\text{minimize}} \|X\theta - Y\|_2^2$$

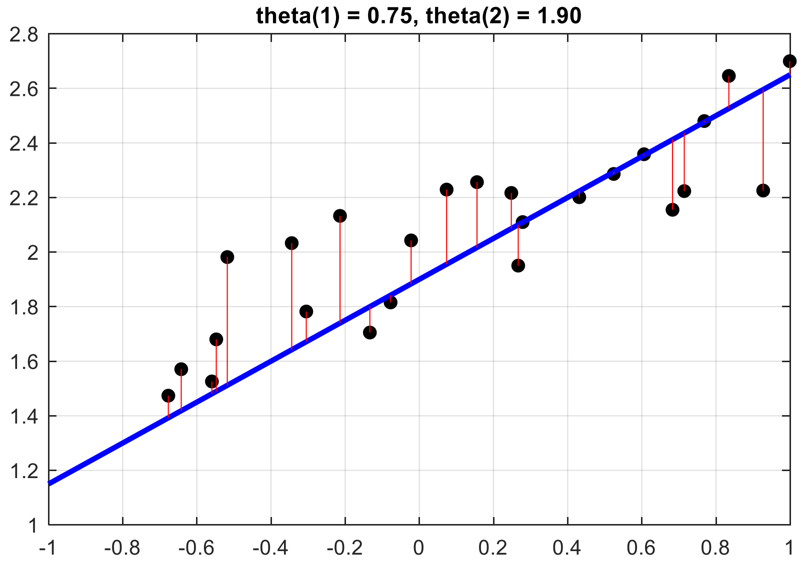
- Analytic solution available!
  - Objective: $f(\theta) = \|X\theta - Y\|_2^2$, set derivative to zero
  - $f(\theta) = (X\theta - Y)^\top (X\theta - Y)$
  - $f(\theta) = \theta^\top X^\top X\theta - 2Y^\top X\theta + Y^\top Y$

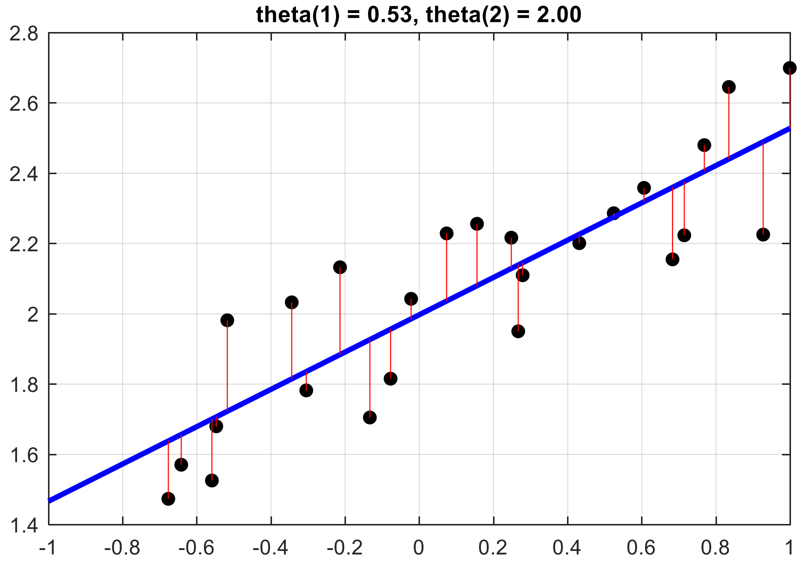
$$\frac{\partial f}{\partial \theta} = 2X^\top X\theta - 2X^\top Y$$

$$0 = 2X^\top X\theta - 2X^\top Y$$

$$X^\top Y = X^\top X\theta$$

$$\theta = (X^\top X)^{-1} X^\top Y$$





Example: Least Squares

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & \|X\theta - Y\|_2^2 \\ \text{subject to} \quad & \theta_1^2 + \theta_2^2 \leq 1 \end{aligned}$$

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & \|X\theta - Y\|_2^2 \\ \text{subject to} \quad & \|\theta\|_2^2 - 1 \leq 0 \end{aligned}$$

- Lagrangian: $L(x, \lambda) = f(x) + \sum_{i=1}^{n} \lambda_i g_i(x)$

$$L(\theta, \lambda) = \|X\theta - Y\|_2^2 + \lambda(\|\theta\|_2^2 - 1)$$

- Stationarity $\nabla_x L(x^*, \lambda^*, \mu^*) = 0$

$$\nabla_\theta L(\theta, \lambda) = 2X^\top X\theta - 2X^\top Y + 2\lambda\theta$$
$$0 = X^\top X\theta - X^\top Y + \lambda\theta$$
$$X^\top Y = (X^\top X + \lambda I)\theta$$

- Primal feasibility: $g_i(x^*) \leq 0,\ a_i^\top x^* - b_i = 0$
- Dual feasibility: $\lambda^* \geq 0$

$$\|\theta\|_2^2 - 1 \leq 0$$
$$\lambda \geq 0$$

- Complementary slackness: $\lambda_i^* g_i(x^*) = 0,\ i = 1, \ldots, n$

$$\lambda(\|\theta\|_2^2 - 1) = 0$$
$$\lambda = 0 \text{ or } \|\theta\|_2^2 = 1$$

Example: Least Squares

- Case 1: If $\lambda = 0$, then
  - $\lambda \geq 0$ is satisfied automatically
  - $X^\top Y = (X^\top X)\theta \Rightarrow \theta = (X^\top X)^{-1} X^\top Y$
  - If $\|\theta\|_2^2 - 1 \leq 0$ happens to be true, we are done
  - Otherwise, try case 2

- Case 2: If $\|\theta\|_2^2 = 1$, then
  - $\|\theta\|_2^2 - 1 \leq 0$ is satisfied automatically
  - $X^\top Y = (X^\top X + \lambda I)\theta \Rightarrow \theta = (X^\top X + \lambda I)^{-1} X^\top Y$
  - Solve $\|\theta\|_2^2 = 1$ and $\theta = (X^\top X + \lambda I)^{-1} X^\top Y$ for θ and λ
  - If $\lambda \geq 0$, we are done

KKT conditions:

- $X^\top Y = (X^\top X + \lambda I)\theta$
- $\|\theta\|_2^2 - 1 \leq 0$
- $\lambda \geq 0$
- $\lambda = 0$ or $\|\theta\|_2^2 = 1$

Solving the Optimality Conditions

$$\begin{aligned} \text{minimize} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \le 0, i = 1, \dots, n \\ & a_j^\top x = b_j, j = 1, \dots, m \end{aligned}$$

- Equations to solve: KKT conditions
  - Stationarity $\nabla_x L(x^*, \lambda^*, \mu^*) = 0$
  - Primal feasibility: $g_i(x^*) \le 0$, $a_i^\top x^* - b_i = 0$
  - Dual feasibility: $\lambda^* \ge 0$
  - Complementary slackness: $\lambda_i^* g_i(x^*) = 0$, $i = 1, \dots, n$

- Use numerical equation solvers, or do it by hand (as much as possible)

- For convex problems, KKT conditions are necessary and sufficient
- For non-convex problems, KKT conditions are just necessary

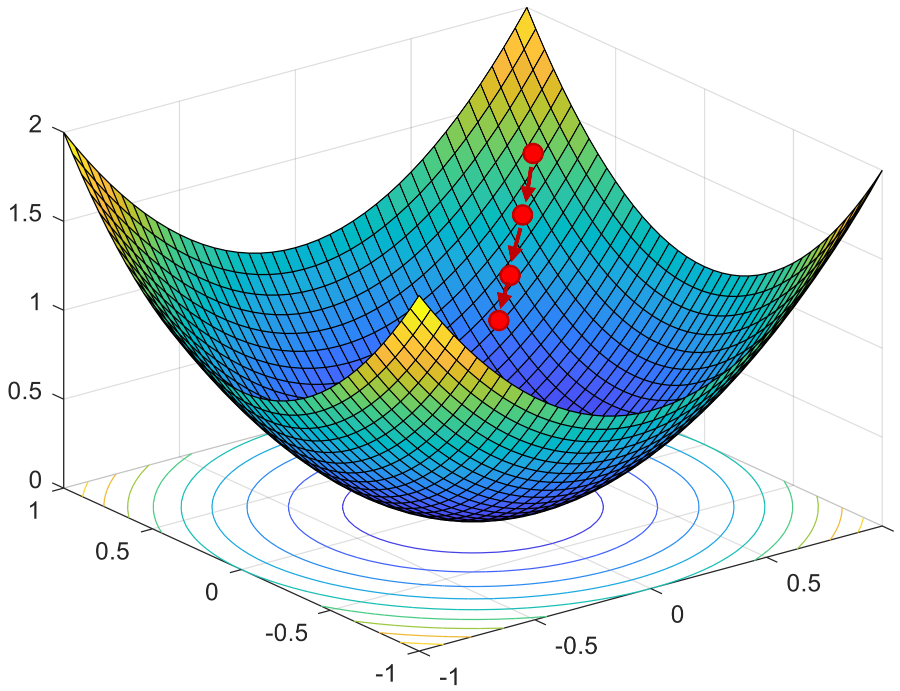
Outline

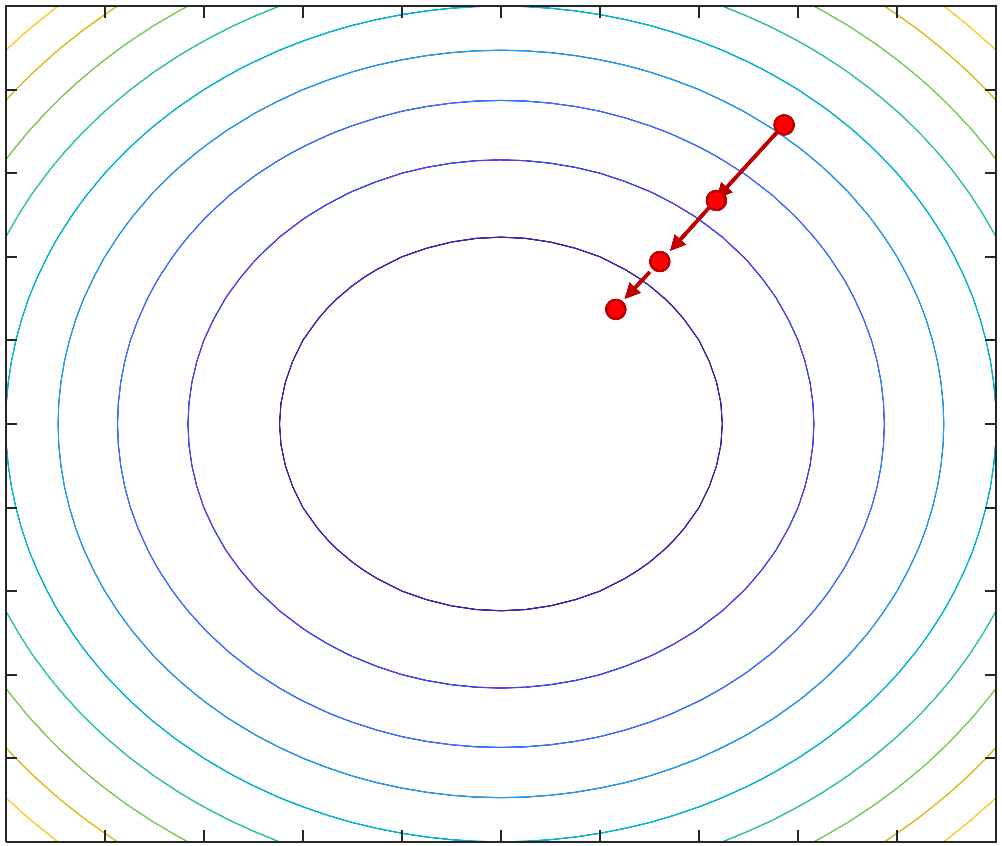
- Solving convex optimization problems
  - Solving the optimality conditions

  - **Gradient methods for approximating solutions to convex optimization problems**
    - **Unconstrained case**

Numerical Solution: Gradient Methods

- Start from x^0 and construct a sequence x^k such that $x^k \to x^*$
  - Calculate x^{k+1} from x^k by "going down the gradient"
  - Unconstrained case: $x^{k+1} = x^k - \alpha^k \nabla f(x)$, $\alpha^k > 0$



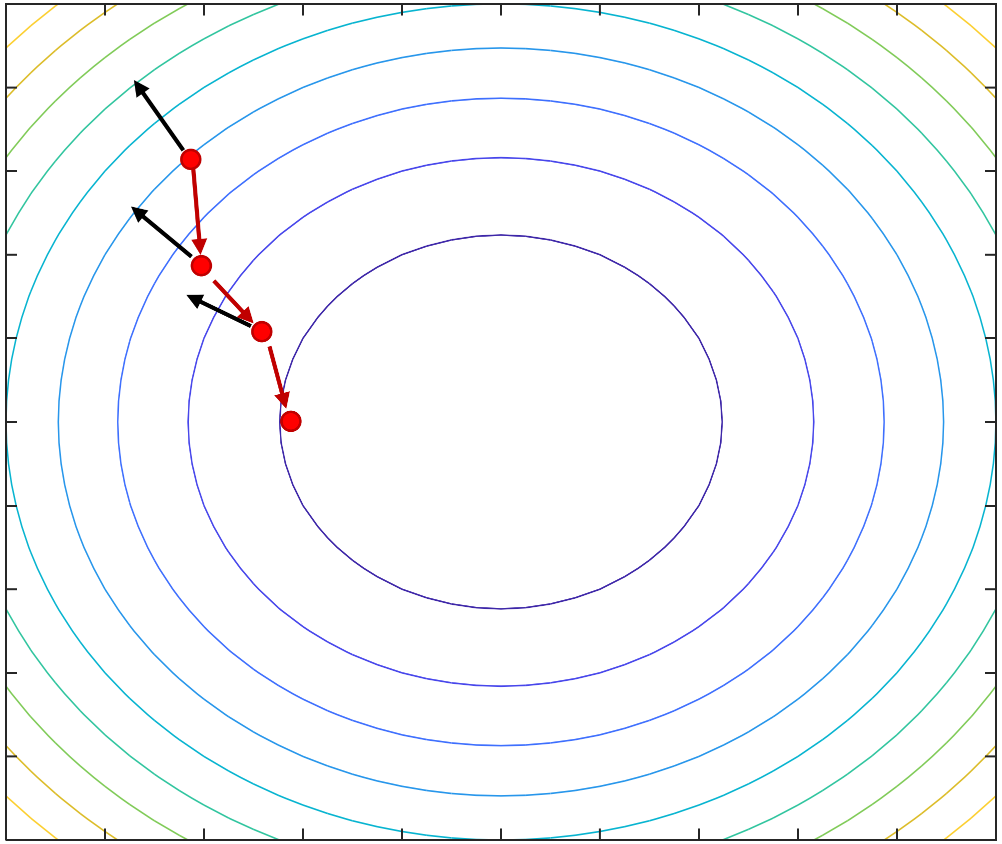


Numerical Solution: Gradient Methods

- Start from x^0 and construct a sequence x^k such that $x^k \rightarrow x^*$
  - Calculate x^{k+1} from x^k by “going down the gradient”
  - Unconstrained case: $x^{k+1} = x^k - \alpha^k \nabla f(x)$, $\alpha^k > 0$
- More generally, $x^{k+1} = x^k + \alpha^k d^k$ for some d such that

$$\nabla f\left(x^k\right) \cdot d^k < 0$$

- Tuning parameters: descent direction d^k, and step size α^k



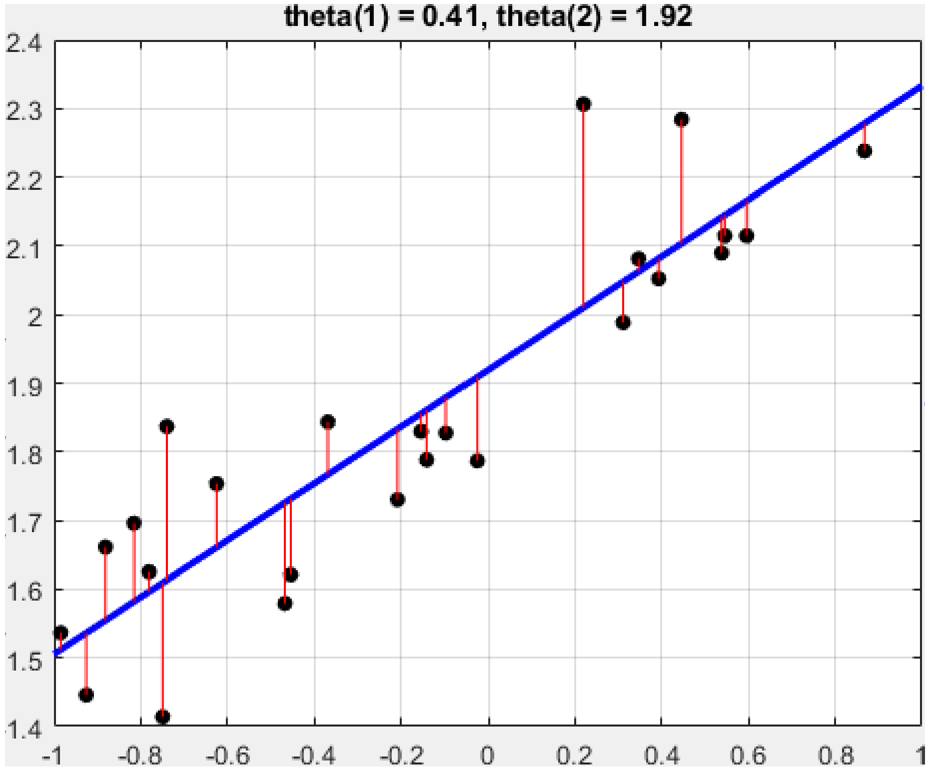
Descent Direction

- Steepest descent: $d^k = -\nabla f\left(x^k\right)$
  - $x^{k+1} = x^k - \alpha^k \nabla f(x)$
  - Simple but sometimes leads to slow convergence

Steepest Descent (Gradient Descent) Example

- Line fitting: $f(\theta) = \|X\theta - Y\|_2^2$
  - $\dfrac{\partial f}{\partial \theta} = 2X^\top X\theta - 2X^\top Y$

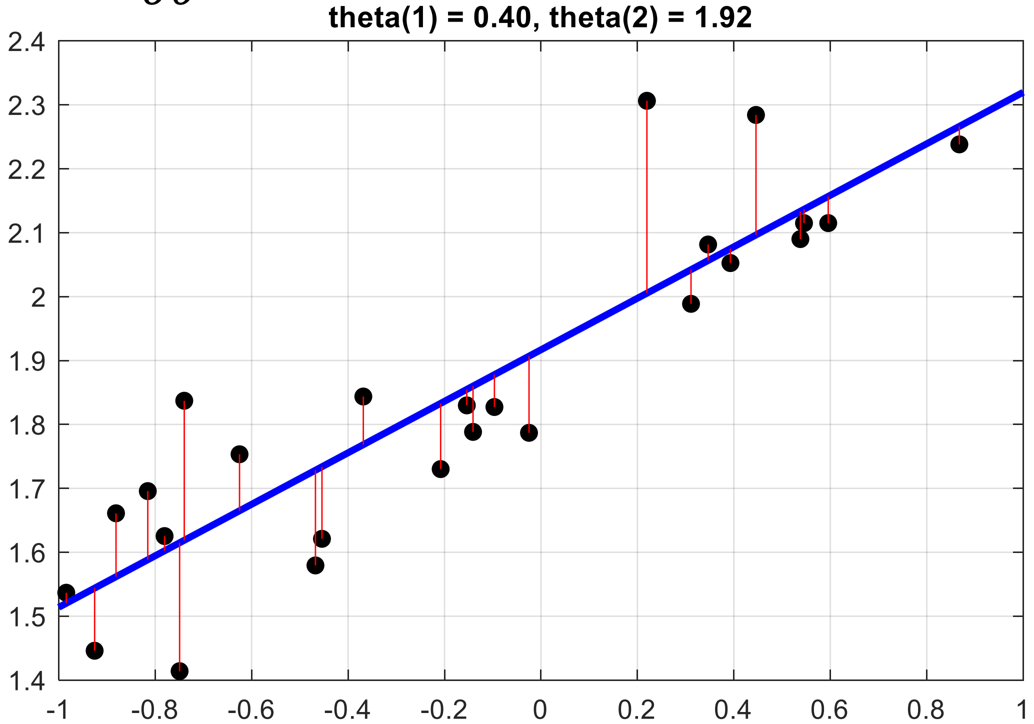
```
theta_last = [-2; -2];
dtheta = inf;
maxIter = 500;

for k = 1:maxIter
    if (norm(dtheta) <= 0.001)
        break;
    end

    alpha = 0.1/k;
    theta = theta_last - alpha*(2*X'*X*theta_last - 2*X'*Y);
    dtheta = theta_last - theta;
    theta_last = theta;
end
```



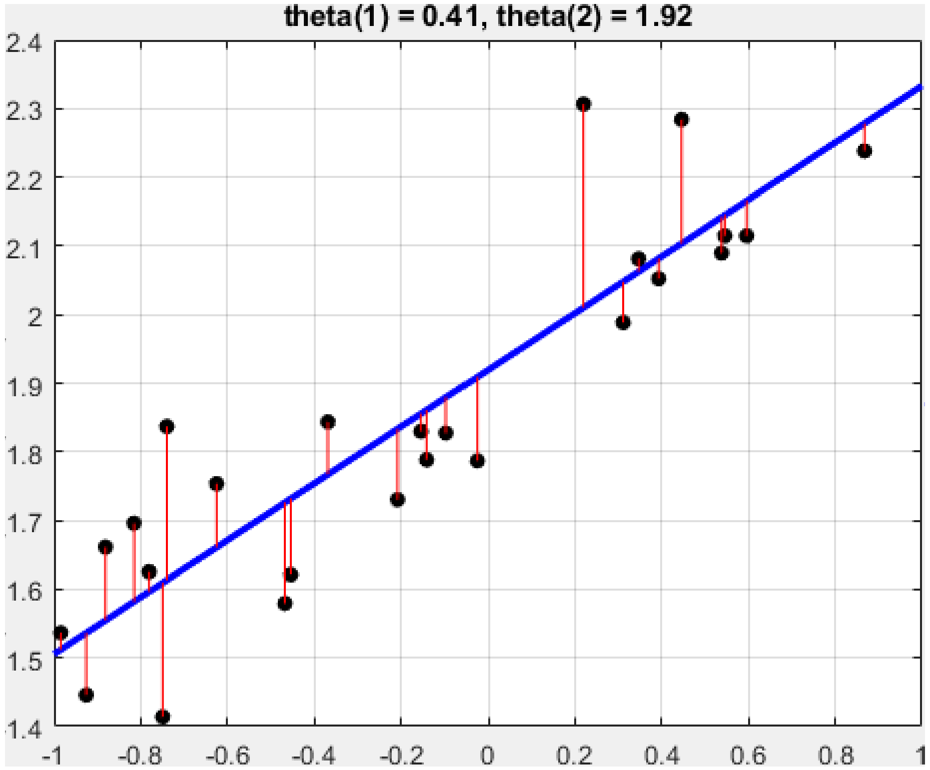
$$\theta^{k+1} = \theta^k - \underbrace{\frac{0.1}{k}}_{\alpha^k}(2X^\top X\theta - 2X^\top Y)$$

Steepest Descent (Gradient Descent) Example

- Line fitting: $f(\theta) = \|X\theta - Y\|_2^2$
  - $\frac{\partial f}{\partial \theta} = 2X^\top X\theta - 2X^\top Y$

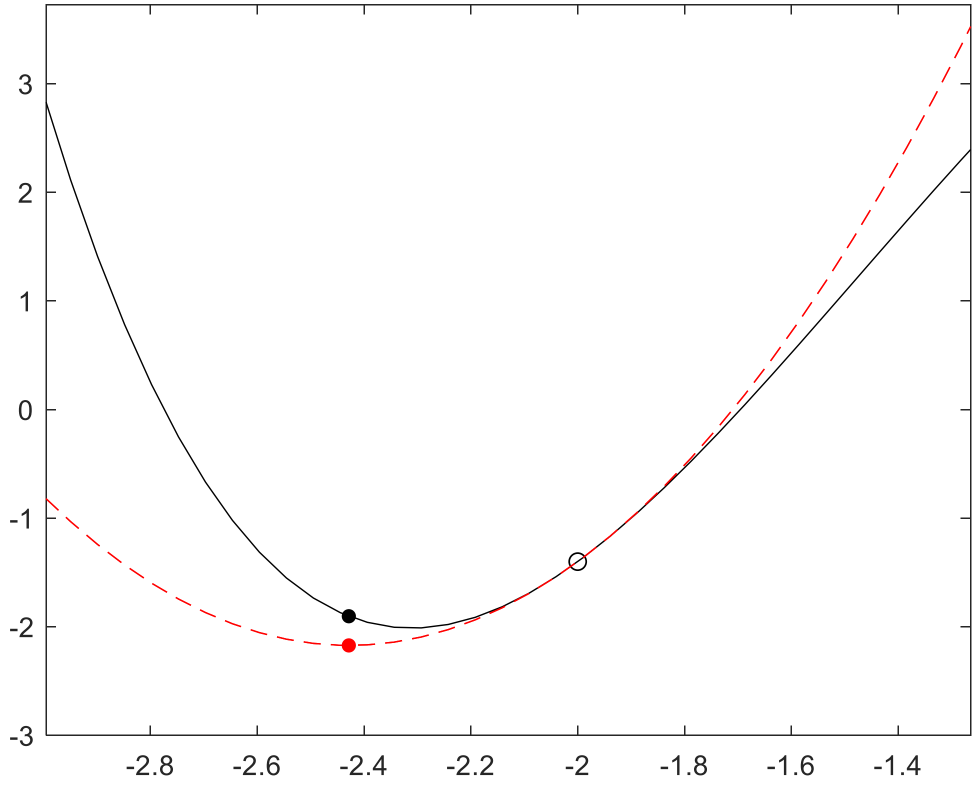


Previously: $\theta = (X^\top X)^{-1} X^\top Y$



$$\theta^{k+1} = \theta^k - \underbrace{\frac{0.1}{k}}_{\alpha^k}(2X^\top X\theta - 2X^\top Y)$$

Descent Direction



- Steepest descent: $d^k = -\nabla f\left(x^k\right)$
  - $x^{k+1} = x^k - \alpha^k \nabla f(x)$
  - Simple but sometimes leads to slow convergence

- Newton's method: $d^k = \left(\nabla^2 f\left(x^k\right)\right)^{-1} \nabla f\left(x^k\right)$
  - Minimize the quadratic approximation:

$$f^k(x) = f\left(x^k\right) + \nabla f\left(x^k\right)^{\mathsf{T}}\left(x - x^k\right) + \frac{1}{2}\left(x - x^k\right)^{\mathsf{T}} \nabla^2 f\left(x^k\right)\left(x - x^k\right)$$

  - Set gradient to zero to obtain next iterate

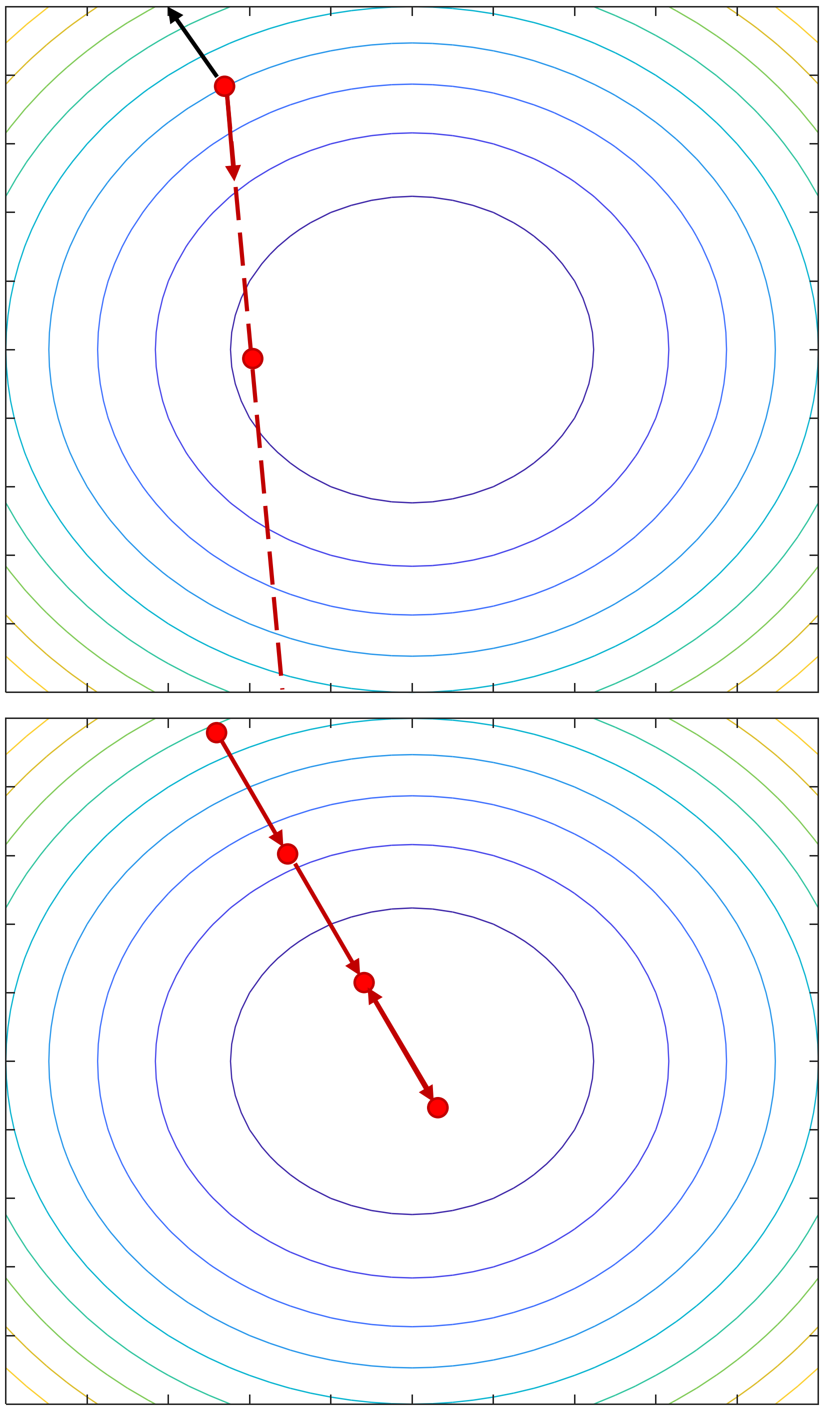
$$\nabla f^k(x) = \nabla f\left(x^k\right) + \nabla^2 f\left(x^k\right)\left(x - x^k\right) = 0$$

$$\Rightarrow x^{k+1} = x^k - \left(\nabla^2 f\left(x^k\right)\right)^{-1} \nabla f\left(x^k\right)$$

  - Fast convergence, but matrix inverse required
  - Alternatively, use an algorithm to minimize a quadratic function

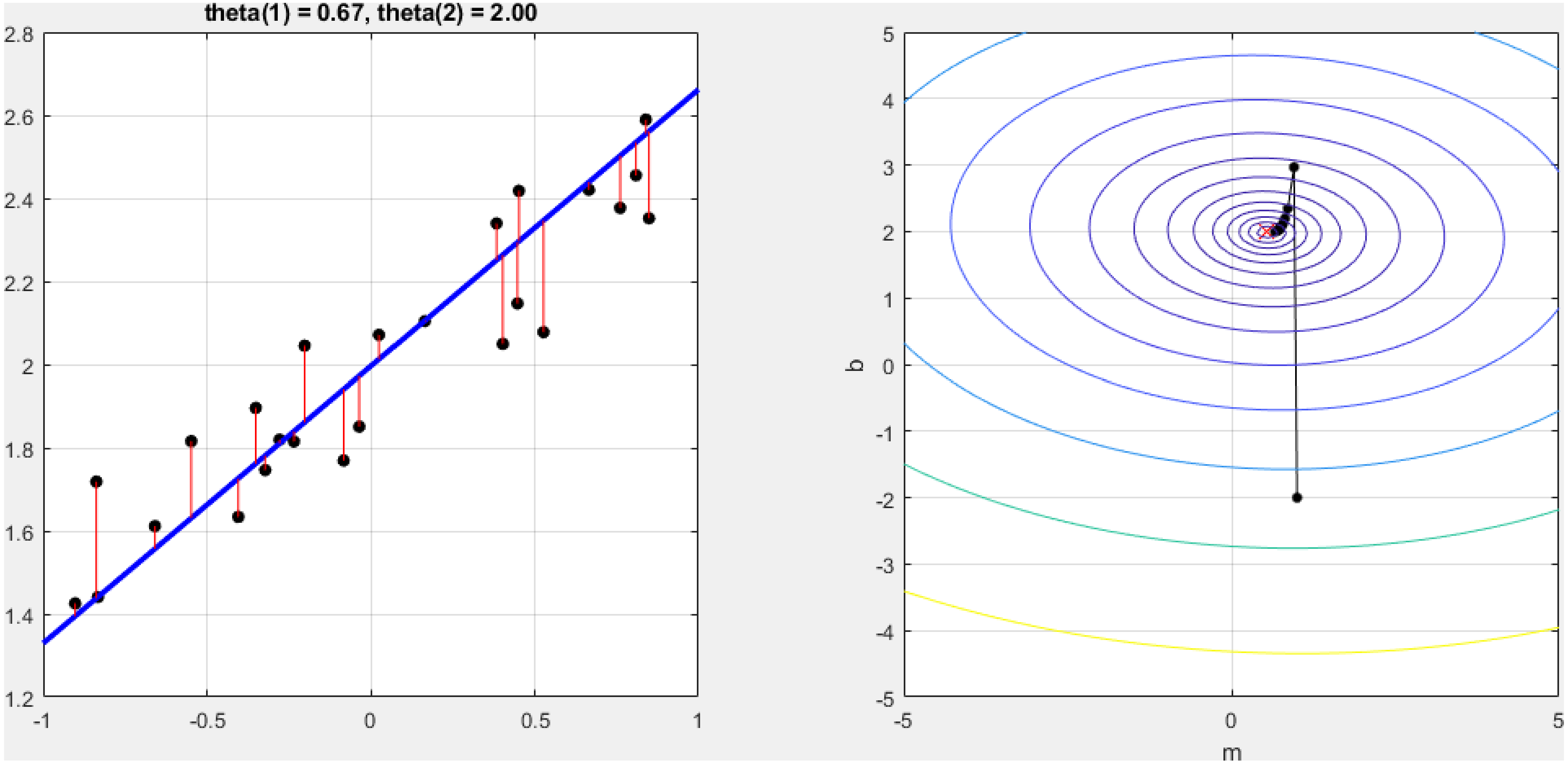
Step Size

- Recall $x^{k+1} = x^k + \alpha^k d^k$, with $\nabla f(x^k)^\top d^k < 0$
- Line search: choose $\alpha^k = \min_{\alpha \geq 0} f(x^k + \alpha^k d^k)$
  - Requires minimization
- Constant step size: $\alpha^k = \alpha$
  - May not converge
- Diminishing step size: $\alpha^k \to 0$
  - Still need to explore all regions $\sum \alpha^k = \infty$
  - For example: $\alpha^k = \frac{\alpha^0}{k}$



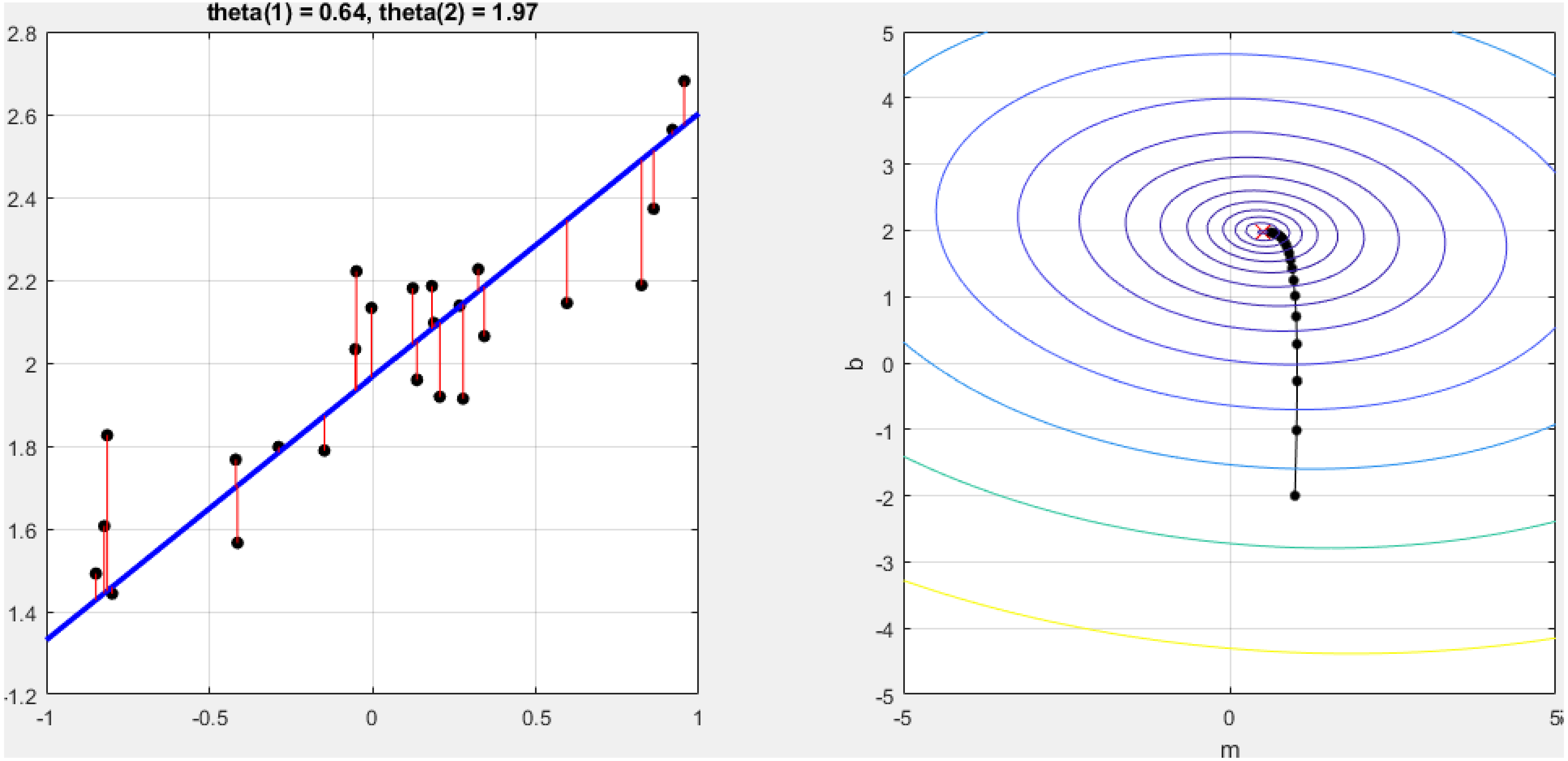
Step Size Example

- Steepest descent, $\alpha^k = \alpha^0/k$



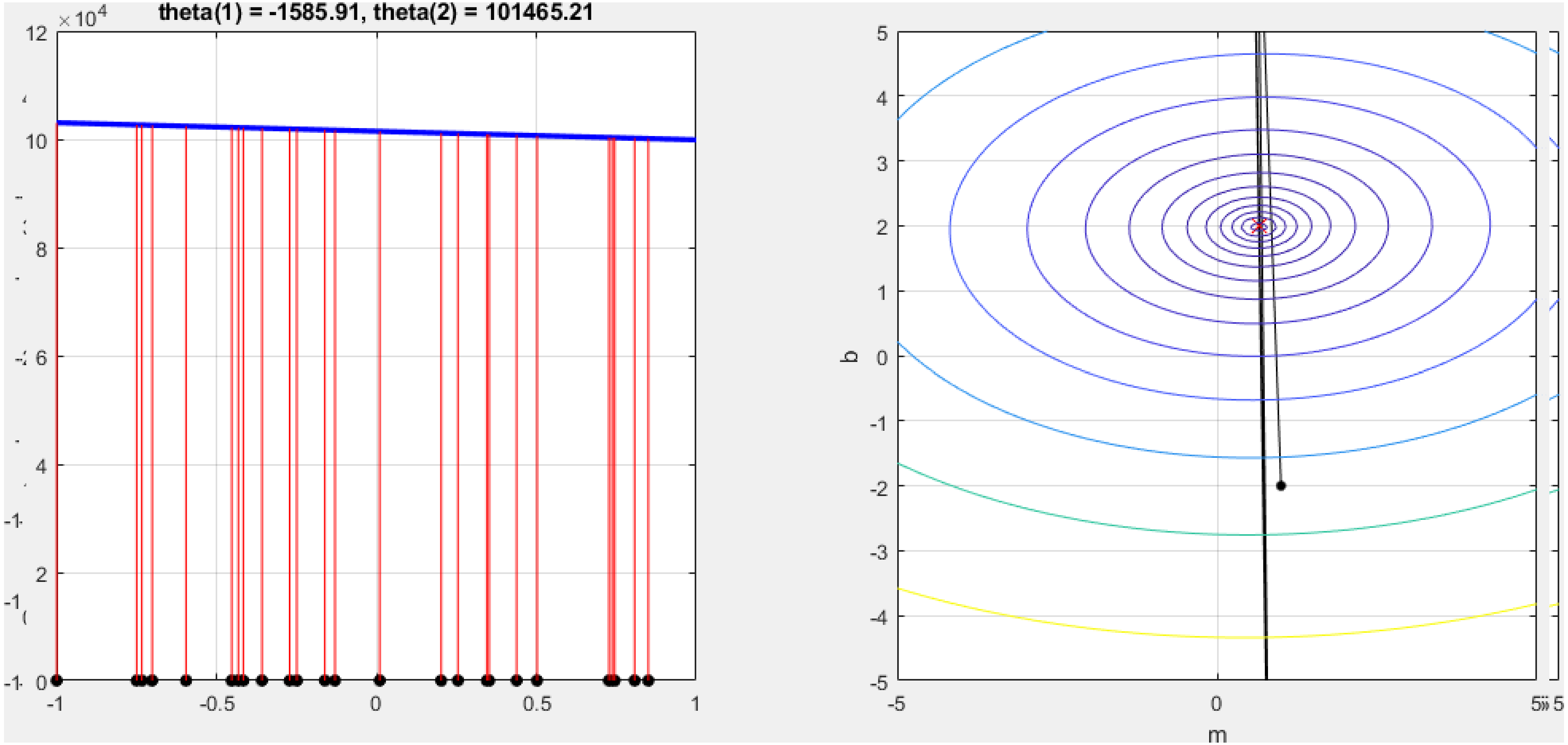
Step Size Example

- Steepest descent, $\alpha^k = \alpha^0$ (small steps)



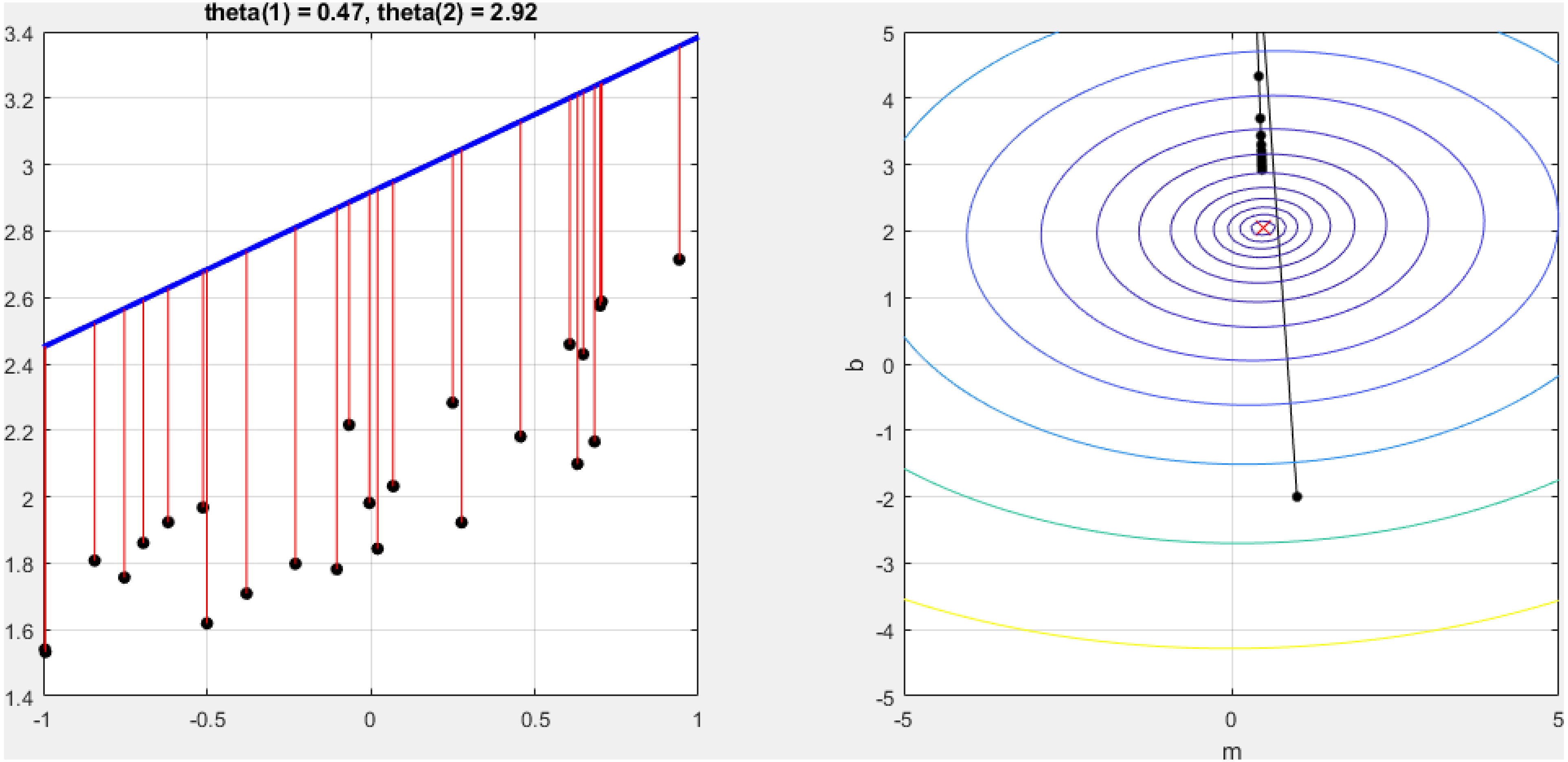
Step Size Example

- Steepest descent, $\alpha^k = \alpha^0$ (large steps)



Step Size Example

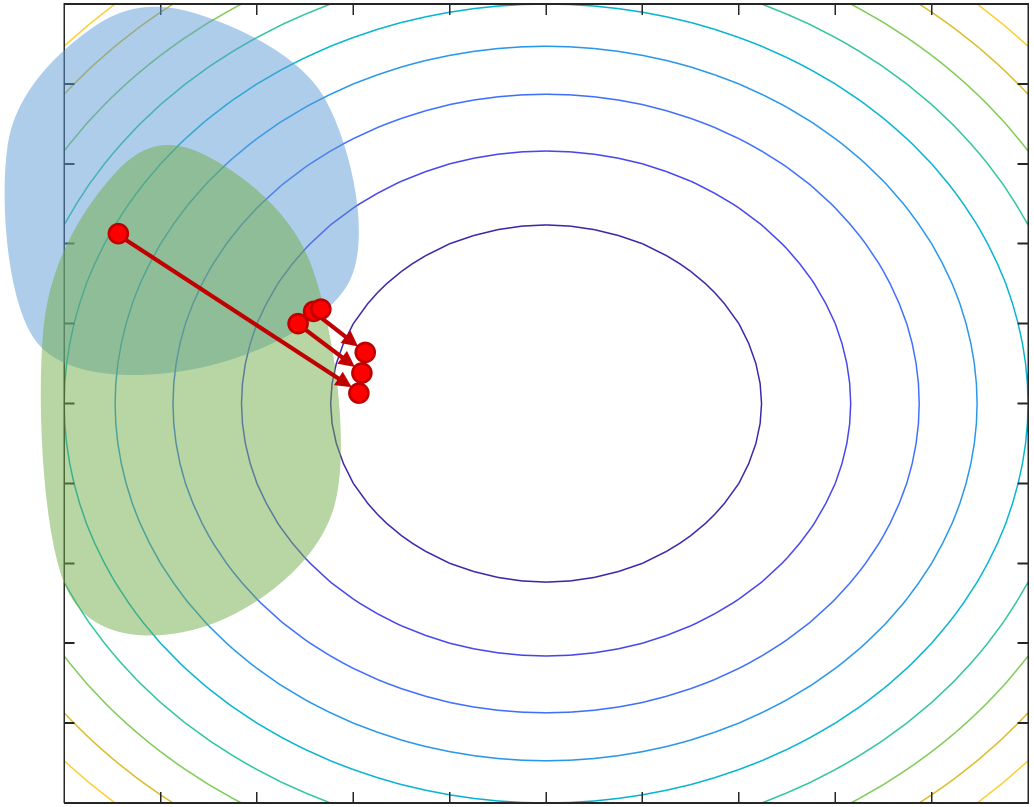
- Steepest descent, $\alpha^k = \alpha^0 / k^2$ (steps do not sum to ∞: $\sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6}$)



Dealing with Constraints

- Idea 1: Apply descent step, and project point to feasible set
  - Proximal gradient methods
  - Difficulty: Computing the projected point

- Idea 2: Set penalty to ∞ for constraint violation
  - Barrier functions



Introduction to cvx

- cvx: MATLAB software for disciplined convex programming
  - http://cvxr.com/cvx/download/
  - http://cvxr.com/cvx/doc/install.html

- User must make sure the program is convex

- Also useful later on in the course

Coding example in cvx

$$\min_x x^\top P x + q^\top x + r$$

$$\text{subject to} \quad -1 \leq x \leq 1$$

$$\text{where} \quad P = \begin{bmatrix} 13 & 12 & -2 \\ 12 & 17 & 6 \\ -2 & 6 & 12 \end{bmatrix}, q = \begin{bmatrix} -22 \\ -14.5 \\ 13 \end{bmatrix}, r = 1$$

```
P = [13 12 -2; 12 17 6; -2 6 12];
q = [-22; -14.5; 13];
r = 1;
n = 3;
x_lower = -1;
x_upper = 1;

% Construct and solve the model
cvx_begin
    variable x(n)
    minimize ( (1/2)*quad_form(x,P) + q'*x + r )
    x >= x_lower;
    x <= x_upper;
cvx_end

fprintf('The computed optimal solution is (%.1f, %.1f, %.1f)\n', x(1), ...
  x(2), x(3))
```

```
Status: Solved
Optimal value (cvx_optval): -21.625

The computed optimal solution is (1.0, 0.5, -1.0)
```

Adapted from S. Boyd and L. Vandenberghe, "Convex Optimization," *Optim. Methods Softw.*, vol. 25, no. 3, pp. 487–487, 2010.

Coding example in cvx

$$\min_x x^\top P x + q^\top x + r$$

subject to $-1 \leq x \leq 1$

where $P = \begin{bmatrix} 13 & 12 & -2 \\ 12 & 17 & 6 \\ -2 & 6 & 12 \end{bmatrix}, q = \begin{bmatrix} -22 \\ -14.5 \\ 13 \end{bmatrix}, r = 1$

- What happens if

$$P = \begin{bmatrix} \mathbf{0} & 12 & -2 \\ 12 & 17 & 6 \\ -2 & 6 & 12 \end{bmatrix}?$$

Adapted from S. Boyd and L. Vandenberghe, "Convex Optimization," *Optim. Methods Softw.*, vol. 25, no. 3, pp. 487–487, 2010.

Coding example in cvx

$$\min_x x^\top P x + q^\top x + r$$

subject to $\quad -1 \leq x \leq 1$

where $\quad P = \begin{bmatrix} \mathbf{0} & 12 & -2 \\ 12 & 17 & 6 \\ -2 & 6 & 12 \end{bmatrix}, q = \begin{bmatrix} -22 \\ -14.5 \\ 13 \end{bmatrix}, r = 1$

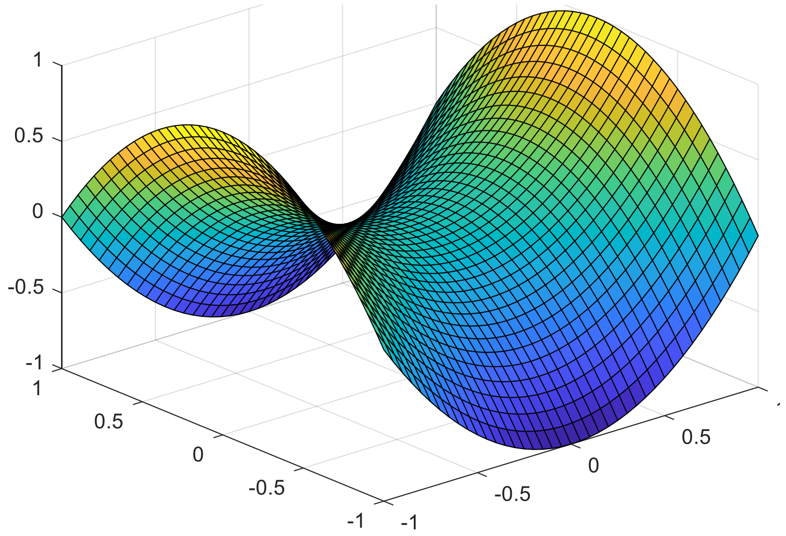
```
P = [0 12 -2; 12 17 6; -2 6 12];
q = [-22; -14.5; 13];
r = 1;
n = 3;
x_lower = -1;
x_upper = 1;

% Construct and solve the model
cvx_begin
    variable x(n)
    minimize ( (1/2)*quad_form(x,P) + q'*x + r )
    x >= x_lower;
    x <= x_upper;
cvx_end

fprintf('The computed optimal solution is (%.1f, %.1f, %.1f)\n', x(1), ...
  x(2), x(3))
```

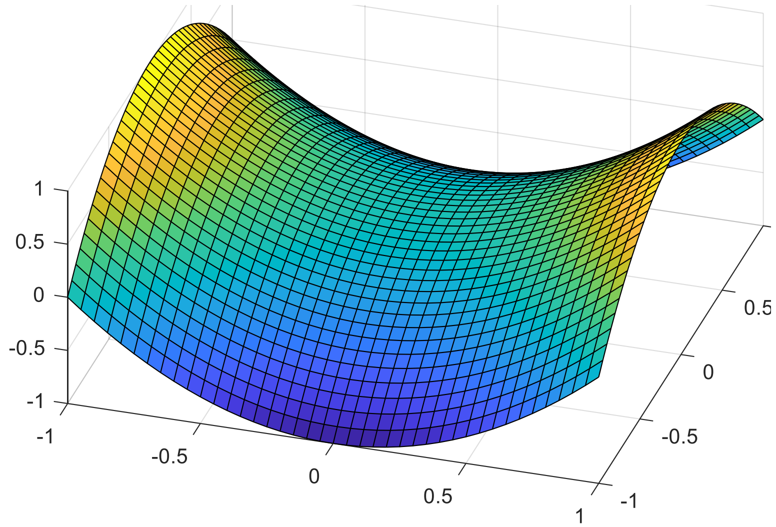
```
Error using cvx/quad_form (line 230)
The second argument must be positive or negative semidefinite.

Error in qp_cvx_example (line 19)
    minimize ( (1/2)*quad_form(x,P) + q'*x + r )
```

```
>> eig(P)

ans =

   -7.3059
   11.4985
   24.8074
```

Adapted from S. Boyd and L. Vandenberghe, “Convex Optimization,” *Optim. Methods Softw.*, vol. 25, no. 3, pp. 487–487, 2010.

Coding example in cvx





$$\min_x \; x^\top P x + q^\top x + r$$

subject to $\quad -1 \leq x \leq 1$

where $\quad P = \begin{bmatrix} 0 & 12 & -2 \\ 12 & 17 & 6 \\ -2 & 6 & 12 \end{bmatrix}, q = \begin{bmatrix} -22 \\ -14.5 \\ 13 \end{bmatrix}, r = 1$

```
Error using cvx/quad_form (line 230)
The second argument must be positive or negative semidefinite.

Error in qp_cvx_example (line 19)
    minimize ( (1/2)*quad_form(x,P) + q'*x + r )
```

```
>> eig(P)

ans =

   -7.3059
   11.4985
   24.8074
```

Adapted from S. Boyd and L. Vandenberghe, “Convex Optimization,” *Optim. Methods Softw.*, vol. 25, no. 3, pp. 487–487, 2010.