日期: /

ZHONG Qiaoyang

24112456g

1. (a) $\hat{\beta}^{ridge} = \underset{\beta}{\arg\min} \|Y - Z\beta\|_2^2 + \lambda\|\beta\|_2^2$

$$\frac{\partial RHS}{\partial \beta} = -2Z^T(Y - Z\beta) + 2\lambda\beta = 0$$

$$\Rightarrow \quad \hat{\beta}^{ridge} = (Z^TZ + \lambda I)^{-1}Z^TY$$

$$\underline{\|\hat{\beta}^{ridge}\|_2^2 = Y^TZ(Z^TZ + \lambda I)^{-2}Z^TY}$$

we consider $(Z^TZ + \lambda I)$ eigenvalues,

eigenvalues $(Z^TZ) \geq 0$,

and as $\lambda \to 0$, <u>eigenvalues of $(Z^TZ + \lambda I)^{-2}$ increase</u>

so <u>$\|\hat{\beta}^{ridge}\|_2^2$ increase, strictly</u> $\square$

(b) $\hat{\beta}^{lasso} = \underset{\beta}{\arg\min} \|Y - Z\beta\|_2^2 + \|\beta\|_1 \cdot \lambda$

$$\|Y - Z\hat{\beta}_1^{lasso}\|_2^2 + \|\hat{\beta}_1^{lasso}\|_1 \cdot \lambda_1 \leq \|Y - Z\hat{\beta}_2^{lasso}\|_2^2 + \|\hat{\beta}_2^{lasso}\|_1 \cdot \lambda_1$$

$$\|Y - Z\hat{\beta}_2^{lasso}\|_2^2 + \|\hat{\beta}_2^{lasso}\|_1 \lambda_2 \leq \|Y - Z\hat{\beta}_1^{lasso}\|_2^2 + \|\hat{\beta}_1^{lasso}\|_1 \cdot \lambda_2$$

add this two inequality, we get

$$\underline{\lambda_1\left(\|\hat{\beta}_1^{lasso}\|_1 - \|\hat{\beta}_2^{lasso}\|_1\right) \leq \lambda_2\left(\|\hat{\beta}_1^{lasso}\|_1 - \|\hat{\beta}_2^{lasso}\|_1\right)}$$

since $\lambda_1 > \lambda_2 > 0$, we have

$\|\hat{\beta}_1^{lasso}\|_1 \leq \|\hat{\beta}_2^{lasso}\|_1$. $\square$ same as ridge.

日期: /

2. $\hat{f}(x) = \sum c_i \cdot x_i^T x$ $\hat{\beta}^{ridge} = \underset{\beta}{\arg\min} \| Y - Z\beta \|_2^2 + \lambda \|\beta\|_2^2$

assume that $C = \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix}$, $Z = \begin{pmatrix} x_1^T \\ \vdots \\ x_n^T \end{pmatrix}$

$\sum c_i x_i^T = C^T \cdot Z$

$\forall x$, $\hat{f}(x) = C^T Z x = \hat{\beta}^T x \Rightarrow \hat{\beta} = Z^T C$

$\hat{\beta} = (Z^T Z + \lambda I)^{-1} Z^T Y = Z^T (Z Z^T + \lambda I)^{-1} Y = Z^T C$

$\Rightarrow$ $\boxed{C = (Z Z^T + \lambda I)^{-1} Y}$.

assignment1_q3

March 7, 2025

[29]:
```
# (a)
import numpy as np
np.random.seed(123)

def getData(n):
    # annotation: the wrong version
    mu = np.array([-2.0, 3.0]).reshape((2, 1)) # mu = np.array([-2.0, 3.0])
    A = np.array([[1.0, 2.0], [3.0, 4.0]])
    X = np.random.normal(size= n*2)
    X = X.reshape((2, n))
    X = A.dot(X) + mu # X = (X + mu).dot(A)
    return X
```

[30]:
```
# (b)
np.random.seed(123)
X = getData(10)
X
```

[30]:
```
array([[-4.44340291, -1.19207249,  1.26575775, -4.78409871, -3.46656417,
        -1.21726601, -0.01481908,  1.94465955,  1.27404405, -2.0943676 ],
       [-2.97243642,  5.61320046,  9.814494  , -4.07449213, -0.51172859,
         6.21690451,  4.5436826 , 10.46040647, 10.81402437,  1.94452439]])
```

[ ]:
```
# (c)
np.random.seed(123)
X = getData(10000)
mu_hat = np.mean(X, axis=1)
sigma_hat = 1/(10000-1) * (
    (X - mu_hat.reshape((2, 1))) @
    (X - mu_hat.reshape((2, 1))).T
)
print("mu_hat:", np.round(mu_hat, 4))
print("sigma_hat:\n", np.round(sigma_hat, 4))
```

```
mu_hat: [-1.9695  3.0706]
sigma_hat:
 [[ 5.0391 11.0854]
 [11.0854 25.1818]]
```

[81]:

```python
# (d)
import torch
np.random.seed(123)
n = 1000
X = getData(n)
X = torch.tensor(X).T # dim: n x 2

X = X.unsqueeze(axis=-1) # dim: n x 2 x 1
X = X.permute(0, 2, 1) # dim: n x 1 x 2
X = X.repeat(1, n, 1) # dim: n x n x 2

U = X.clone().permute(1, 0, 2) # dim: n x n x 2
K = np.exp(-torch.norm(X - U, dim=-1)**2) # dim: n x n
K = K.numpy()
eigenvalues = np.linalg.eigh(K).eigenvalues
eigenvalues.sort()
eigenvalues = np.round(eigenvalues[::-1][:3], 4)
eigenvalues
```

[81]:

```
array([105.5731,  91.1879,  78.3936])
```

日期: /

4. (a) since $(x-1)^2$ is convex, $\lambda|x|$ is convex

the objection function is convex,

so there exists only one optimizer.

That is $(x^*(\lambda)-1)^2+\lambda|x^*(\lambda)| < (x-1)^2+\lambda|x|$. for $\forall x \neq x^*$.

(b) if $x^*(\lambda)=0$, $\frac{\partial f}{\partial x} = 2(x-1)+\lambda \, \text{sign}(x)$

Necessary: when $x_1 \to 0^-$, we must have $\left.\frac{\partial f}{\partial x}\right|_{x_1} \leqslant 0$

when $x_2 \to 0^+$, we must have $\left.\frac{\partial f}{\partial x}\right|_{x_2} \geqslant 0$

$$\begin{cases} \left.\frac{\partial f}{\partial x}\right|_{x_1} = 2(x_1-1)-\lambda \to -2-\lambda \leqslant 0 & \text{as } x_1 \to 0^-, \\ \left.\frac{\partial f}{\partial x}\right|_{x_2} = 2(x_2-1)+\lambda \to -2+\lambda \geqslant 0 & \text{as } x_2 \to 0^+, \end{cases}$$

$\Rightarrow \lambda \geqslant 2$.

Then we verify $\lambda \geqslant 2$ is sufficient:

when $\lambda \geqslant 2$, f. is convex, so $x^*(\lambda)$ is minimizer.

we have when $\boxed{\lambda \geqslant 2}$, $x^*(\lambda)=0$. □

日期:　　/

5、

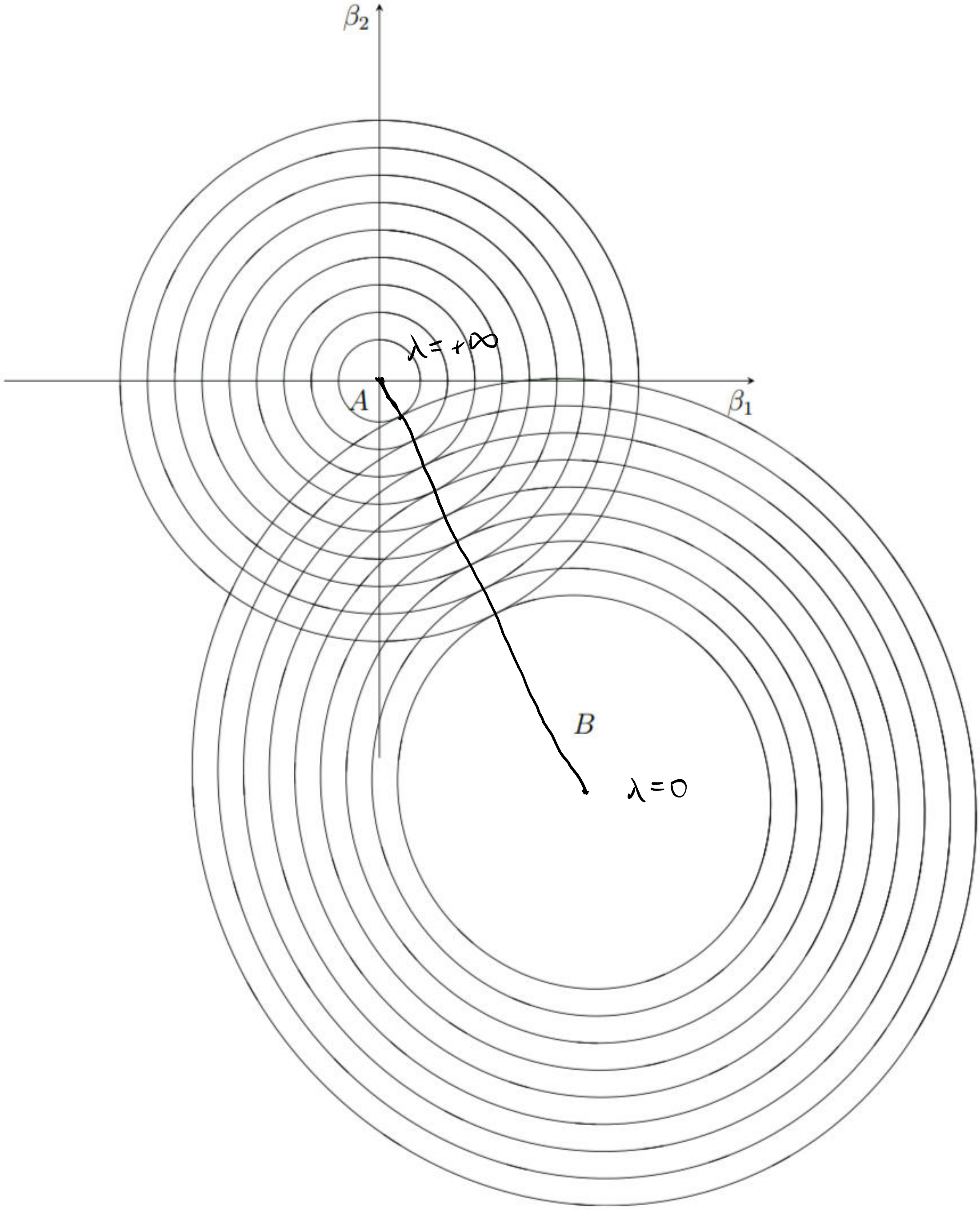

Figure 1

日期: /

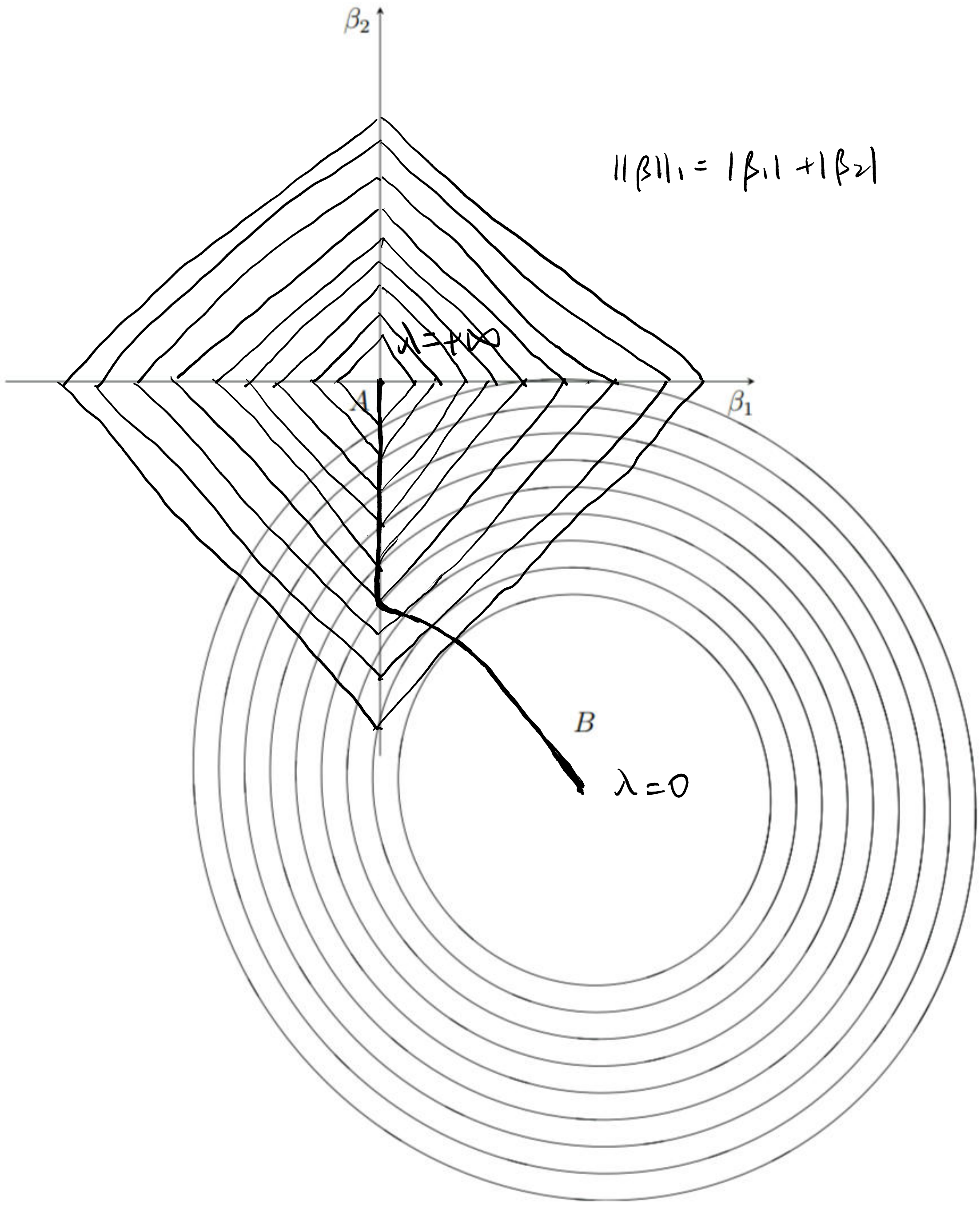

Figure 2

obviously $\underline{k(x, y) = k(y, x)}$

6. (a) Proove $\sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j k(x_i, x_j) \geq 0$, for $\forall c$, $\forall x$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j k(x_i, x_j) = \sum_i \sum_j c_i c_j (x_i^3 x_j^3 + x_i x_j)$$

$$= \sum_i c_i x_i^3 \sum_j c_j x_j^3 + \sum_i c_i x_i \sum_j c_j x_j$$

$$\underline{= \left(\sum_i c_i x_i^3\right)^2 + \left(\sum_i c_i x_i\right)^2 \geq 0}$$

so $k(x_i, y)$ is positivive semi-definite.

(b) $\underline{G(x, y) = G(y, x)}$ obviously

$\bar{G}(x, y) \triangleq \langle x, y \rangle$ is positive semi-definite,

so $\bar{G} \circ \bar{G} \circ \bar{G}(x, y) = \langle x, y \rangle^3$ is positive semi-definite,

and $G(x, y) = \langle x, y \rangle^3 + \langle x, y \rangle$ is PSD. $\square$

(c) $\underline{k(x, y) = k(y, x)}$

$\forall\ c, x,$ $$\sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j \frac{1 + x_i x_j}{1 - x_i x_j}$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j \left(-1 + \frac{2}{1 - x_i x_j}\right)$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j \left(-1 + 2\sum_{k=0}^{+\infty} (x_i x_j)^k\right)$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{n} \left(1 + 2\sum_{k=1}^{+\infty} x_i^k x_j^k\right) c_i c_j = \underline{\left(\sum_{i=1}^{n} c_i\right)^2 + 2\sum_{k=1}^{+\infty} \left(\sum_{i=1}^{n} x_i^k c_i\right)^2 \geq 0}$$

(d) prove $\exp(i(x-u))$ is positive semi-definite

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j \exp(i(x_i - x_j)) = \sum_{i=1}^{n} c_i \exp(i x_i) \cdot \sum_{j=1}^{n} \overline{c_j \exp(+i x_j)}$$

$$= \left| \sum_i c_i \exp(i x_i) \right|^2 \geq 0$$

so $\exp(i(u-x))$ is PSD too,

$K(u,x)$ is PSD. □