先端データ解析論　第 3 回レポート

電子情報学専攻　48-176403 石毛真修

2017 年 5 月 9 日

大問 1.

線形モデル $f_\theta(\mathbf{x}) = \sum_{j=1}^{b} \theta_j \phi_j(\mathbf{x})$ に対する重み付き最小二乗法

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \frac{1}{2} \sum_{i=1}^{n} \tilde{\omega_i} \left(f_\theta(\mathbf{x})_i - y_i\right)^2 = (\Phi^{\mathrm{T}} \tilde{W} \Phi)^{-}1 \Phi^{\mathrm{T}} \tilde{W} \mathbf{y}$$

証明

$$\Phi = \begin{bmatrix} \phi_1(\mathbf{x}_1) & \phi_2(\mathbf{x}_1) & ... & \phi_b(\mathbf{x}_1) \\ \phi_1(\mathbf{x}_2) & \phi_2(\mathbf{x}_2) & ... & \phi_b(\mathbf{x}_2) \\ ... & & & \\ \phi_1(\mathbf{x}_n) & \phi_2(\mathbf{x}_n) & ... & \phi_b(\mathbf{x}_n) \end{bmatrix}$$

とする。

誤差関数 $\frac{1}{2} \sum_{i=1}^{n} \tilde{\omega_i} \left(f_\theta(\mathbf{x})_i - y_i\right)^2$ を J とおくと

$$\begin{aligned} J &= \frac{1}{2} (\hat{\mathbf{y}} - \mathbf{y})^{\mathrm{T}} \tilde{W} (\hat{\mathbf{y}} - \mathbf{y}) \\ &= \frac{1}{2} (\Phi\theta - \mathbf{y})^{\mathrm{T}} \tilde{W} (\Phi\theta - \mathbf{y}) \\ &= \frac{1}{2} \left[\theta^{\mathrm{T}} \Phi^{\mathrm{T}} \tilde{W} \Phi\theta - \theta^{\mathrm{T}} \Phi^{\mathrm{T}} \tilde{W} \mathbf{y} - \mathbf{y}^{\mathrm{T}} \tilde{W} \Phi\theta + \mathbf{y}^{\mathrm{T}} \tilde{W} \mathbf{y}\right] \end{aligned}$$

この J を最小化したいので、$\frac{dJ}{d\theta} = 0$ とすると、

$$\begin{aligned} \frac{dJ}{d\theta} &= \frac{1}{2} \left[2\Phi^{\mathrm{T}} \tilde{W} \Phi\theta - \Phi^{\mathrm{T}} \tilde{W} \mathbf{y} - (y^{\mathrm{T}} \tilde{W} \Phi)^{\mathrm{T}}\right] \\ &= \frac{1}{2} \left[2\Phi^{\mathrm{T}} \tilde{W} \Phi\theta - 2\Phi^{\mathrm{T}} \tilde{W} \mathbf{y}\right] \\ &= \Phi^{\mathrm{T}} \tilde{W} \Phi\theta - \Phi^{\mathrm{T}} \tilde{W} \mathbf{y} \end{aligned}$$

$$\therefore \hat{\theta} = (\Phi^{\mathrm{T}} \tilde{W} \Phi)^{-}1 \Phi^{\mathrm{T}} \tilde{W} \mathbf{y}$$

大問 2.

微分可能で対称な損失関数 $\rho(r)$ に対して $\tilde{r}$ で接する二次上界は、

$$\tilde{\rho}(r) = \frac{\tilde{\omega}}{2}r^2 + const$$

証明

$\rho(r)$ は対称なので、それに接する二次上界 $\tilde{\rho}(r)$ も対称になる。よって、

$$\tilde{\rho}(r) = ar^2 + b$$

接しているという条件から、接点の傾きが同じであり、かつ共通の解を持つので、

$$\rho'(\tilde{r}) = 2a\tilde{r}$$
$$\rho(\tilde{r}) = a\tilde{r}^2 + b$$

これを解いて、

$$a = \frac{1}{2}\frac{\rho'(\tilde{r})}{\tilde{r}} \qquad b = \rho(\tilde{r}) - \frac{1}{2}\rho'(\tilde{r})\tilde{r}$$

よって、

$$\tilde{\rho}(r) = \frac{1}{2}\frac{\rho'(\tilde{r})}{\tilde{r}}r^2 + \left(\rho(\tilde{r}) - \frac{1}{2}\rho'(\tilde{r})\tilde{r}\right) = \frac{\tilde{\omega}}{2}r^2 + const$$

大問 3.

まず、次のようなデータ直線 $y = x$ に正規分布から生成されるノイズを載せたデータを作成する。

```
import matplotlib
import matplotlib.pyplot as plt
import numpy as np

n, N = 10, 1000
x = np.linspace(-3, 3, n)
X = np.linspace(-4, 4, N)                C

# Original data
np.random.seed(5)
y = x + 0.2 * np.random.normal(size=n)
y[2], y[n -2], y[n - 1] = [-4] * 3
```

次に、適当に初期化したパラメータ thetas = [theta1, theta2] を元に、繰り返し最小二乗法を行うことで Tukey 回帰を行う。なお、コード中の変数 Phi は計画行列である。

```
# Tukey Regression
Phi = np.array([[1, x[i]] for i in range(n)])
thetas = np.random.normal(1, 0.1, 2)
ETA = 2
def w_tukey(r):
    return (1 - (r**2)/(ETA**2))**2 if np.absolute(r) <= ETA else 0
# Iteration
for step in range(20):                                              C
    errors = y - np.matmul(Phi, thetas)
    _W = np.diag([w_tukey(err) for err in errors])
    Q = Phi.T @ _W @ Phi
    if np.linalg.det(Q) == 0:
        break
    thetas = np.linalg.inv(Q) @ Phi.T @ _W @ y
```

最後にこれをプロットする。

```
# Prediction
def pred(x):
    return thetas[0] + thetas[1] * x

plt.scatter(x, y)
plt.plot(X, pred(X), 'b-')              C
axes = plt.gca()
axes.set_ylim([-4.5, 3])
plt.show()
```

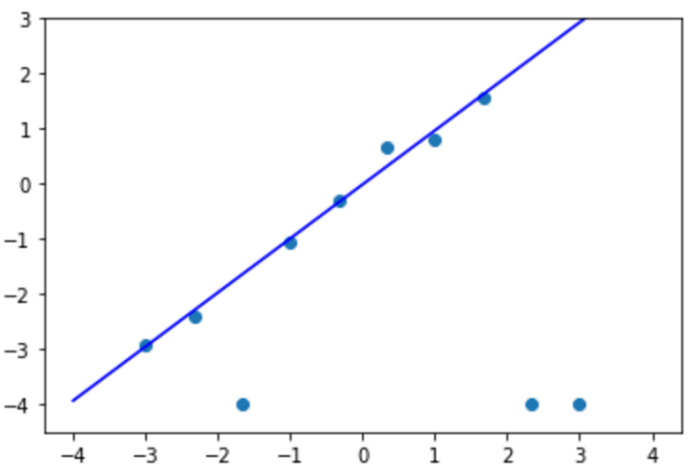

図 1: Tukey 回帰の結果