

課題 11.E.3 略解例

表 2.E.1 のデータを、グループ変数 XA 、 XB と XC をダミー変数として設定するときの重回帰モデルの Stan スクリプトをリスト 11.E.3.1 のように用意した。

リスト 11.E.3.1 重回帰モデルの Stan スクリプト

```
data{
  int N;
  real Y[N];
  real X[N];
  real XA[N];
  real XB[N];
  real XC[N];
}
parameters{
  real b;
  real bA;
  real bB;
  real bC;
  real<lower = 0.0> sgm;
}
transformed parameters{
  real Yreg[N];
  for (i in 1:N)
    Yreg[i] = b * X[i] + bA * XA[i] + bB * XB[i] + bC * XC[i];
}
model{
  for (i in 1:N)
    Y[i] ~ normal(Yreg[i], sgm);
}
```

リスト 11.E.3.1 の Stan スクリプトを用いる Python スクリプトをリスト 11.E.3.2 のように用意した。

リスト 11.E.3.2 Stan スクリプト (リスト 11.E.3.1) を用いる Python スクリプト

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

```
Data = [# ID SA SB
        ['A', 11, 10],
        ['A', 12, 11],
        ['A', 11, 12],
        ['A', 10, 11],
        ['A', 11, 11],
        ['B', 13, 12],
```

```

['B', 14, 13],
['B', 13, 14],
['B', 12, 13],
['B', 13, 13],
['C', 15, 14],
['C', 16, 15],
['C', 15, 16],
['C', 14, 15],
['C', 15, 15]
]

```

```
N = len(Data)
```

```
Y = []
```

```
X = []
```

```
XA = [0] * N
```

```
XB = [0] * N
```

```
XC = [0] * N
```

```
pos = 0
```

```
for v in Data:
```

```
    Y.append(v[2])
```

```
    X.append(v[1])
```

```
    if v[0] == 'A':
```

```
        XA[pos] = 1
```

```
    elif v[0] == 'B':
```

```
        XB[pos] = 1
```

```
    else:
```

```
        XC[pos] = 1
```

```
    pos += 1
```

```
DataStan = {'N': N, 'Y': Y, 'X': X, 'XA': XA, 'XB': XB, 'XC': XC}
```

```
sm = pystan.StanModel(file = 'MRegEx3.stan')
```

```
fit = sm.sampling(data = DataStan, n_jobs = 1)
```

```
print(fit)
```

```
smp_b = fit['b']
```

```
bp025 = np.percentile(smp_b, 2.5)
```

```
bp975 = np.percentile(smp_b, 97.5)
```

```
plt.hist(smp_b)
```

```
plt.xlabel('b')
```

```
plt.title('95% CI = [{0:.3f}, {1:.3f}]'.format(bp025, bp975))
```

```
plt.show()
```

```
smp_Yregs = fit['Yreg']
```

```
Yreg = []
```

```
for i in range(N):
```

```
    Yreg.append([])
```

```
for v in smp_Yregs:
```

```
    pos = 0
```

```

for u in v:
    Yreg[pos].append(u)
    pos += 1

meanYreg = [0] * N
for i in range(N):
    meanYreg[i] = np.median(Yreg[i])

plt.plot(meanYreg, Y, 'bo')
plt.xlabel('Yreg')
plt.ylabel('Y')
minYreg = np.amin(meanYreg)
maxYreg = np.amax(meanYreg)
plt.plot([minYreg, maxYreg], [minYreg, maxYreg],
         'b--', label = 'y = x')
plt.legend()
plt.show()

```

リスト 11.E.3.2 のスクリプトを実行すると、図 11.E.3.1 のフォームが表示される。

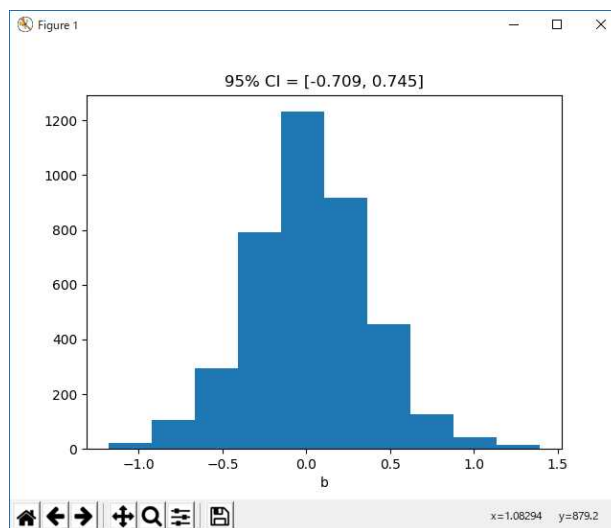


図 11.E.3.1 傾き b の事後分布

傾き b の事後分布における 95%CI は $[-0.709, 0.745]$ であり、0 を含んでおり、図 11.E.3.1 に示されている分布は 0 を中央に含んで分布しているので、傾き b は 0 であると考えられる。

図 11.E.3.1 のフォームを閉じるとお、図 11.E.3.2 のフォームが表示される。

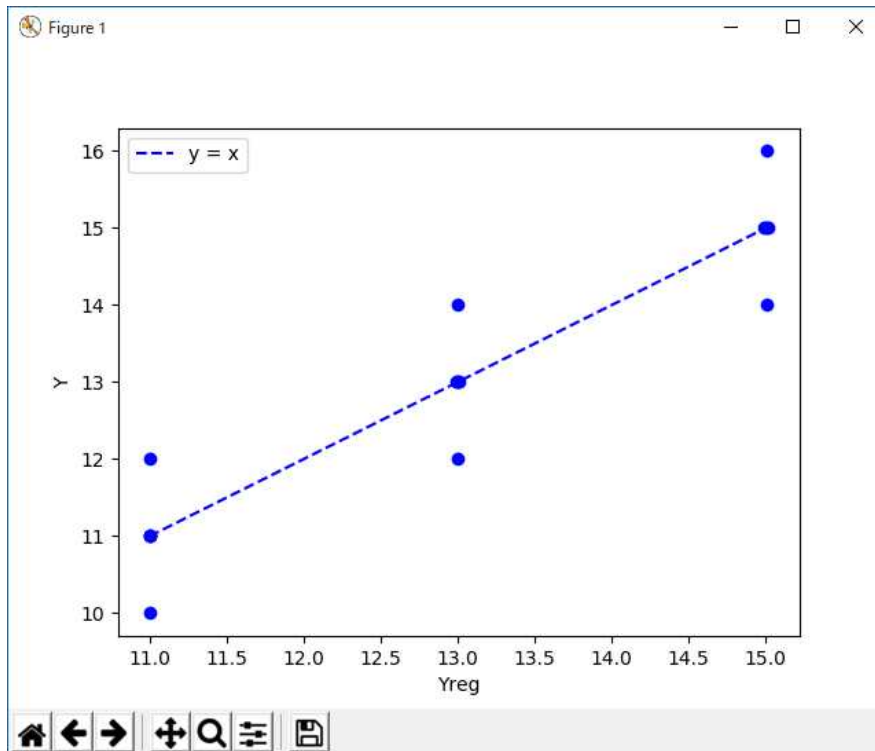


図 11.E.3.2 予測値の事後分布の中央値 Yreg とデータ Y。
 予測値とデータが一致すると点(Yreg, Y)は直線 $y = x$ 上にある。

図 11.E.3.2 のグラフは、予測値（事後分布の中央値）とデータ値の関係を表すものである。予測値とデータ値が一致すれば、直線 $y = x$ 上に点（予測値、データ値）がプロットされる。直線 $y = x$ と点の散布図との関係に特に偏りは認められないので、モデルとデータとの関係に特に問題はないと考えられる。

独立変数 SA の偏回帰係数 b が、図 11.E.3.1 に示されている事後分布から 0 であると考えられ、さらに、モデルとデータとの関係も図 11.E.3.2 に示されている関係から特に問題はないと考えられるが、これはグループの要因を表すダミー変数 XA 、 XB と XC を導入すると、単回帰分析（課題 11.E.1）での変数 SA と従属変数 SB の間の 1 次式の関係は消滅することを示している。