回归分析大作业

柴大开 3210103517

实验问题

本案例收集了 1990-2020 年的国内生产总值 **x1**、出口量 **x2**、总消费量 **x3** 以及总进口量 **y** 的数据（单位均为亿元），希望通过 x1,x2,x3 的历年数据去预测 y 的数据。以下为 1990-2020 年的相关数据：

```
read.table("F:/Desktop/相关数据.txt")
```

```
##           V1       V2       V3       V4
## 1    18872.9   2985.8  12012.6   5560.1
## 2    22005.6   3827.1  13626.2   7225.8
## 3    27194.5   4676.3  16240.5   9119.6
## 4    35673.2   5284.8  20815.9  11271.0
## 5    48637.5  10421.8  28297.5  20381.9
## 6    61339.9  12451.8  36228.1  23499.9
## 7    71813.6  12576.4  43123.3  24133.9
## 8    79715.0  15160.7  47549.7  26967.2
## 9    85195.5  15223.5  51502.8  26849.7
## 10   90564.4  16159.8  56667.5  29896.2
## 11  100280.1  20634.4  63749.1  39273.3
## 12  110863.1  22024.4  68662.2  42183.6
## 13  121717.0  26947.9  74227.9  51378.2
## 14  137422.0  36287.9  79735.7  70483.5
## 15  161840.2  49103.3  89395.3  95539.1
## 16  187318.9  62648.1 101872.5 116921.8
## 17  219438.5  77597.9 115364.2 140974.7
## 18  270092.3  93627.1 137737.4 166924.1
## 19  319244.6 100394.9 158899.0 179921.5
## 20  348517.7  82029.7 174538.1 150648.1
## 21  412119.3 107022.8 201581.8 201722.3
## 22  487940.2 123240.6 244747.2 236402.0
## 23  538580.0 129359.3 275444.3 244160.2
## 24  592963.2 137131.4 306664.2 258168.9
## 25  643563.1 143883.8 338031.9 264241.8
## 26  688858.2 141166.8 371921.5 245502.9
## 27  746395.1 138419.3 410806.8 243386.5
## 28  832035.9 153309.4 456518.6 278099.2
## 29  919281.1 164128.8 506135.1 305010.1
## 30  986515.2 172373.6 552632.3 315627.3
## 31 1013567.0 179278.8 560811.6 322215.2
```

以上的数据从左到右分别为国内生产总值 (V1)、出口量 (V2)、总消费量 (V3) 以及总进口量 (V4).（数据来源于《中国统计年鉴-2022 年》。）

数据处理

首先获得各变量的样本均值、样本标准差数据，代码如下：

```
yx=read.table("F:/Desktop/相关数据.txt")
x1=yx[,1]
x2=yx[,2]
x3=yx[,3]
y=yx[,4]
mean(x1);mean(x2);mean(x3);mean(y)
```

```
## [1] 334824.7
```

```
## [1] 72883.17
```

```
## [1] 181146.5
```

```
## [1] 133990
```

```
sd(x1);sd(x2);sd(x3);sd(y)
```

```
## [1] 319151.4
```

```
## [1] 61019.01
```

```
## [1] 173127.5
```

```
## [1] 111078.6
```

实验步骤

本实验主要通过最小二乘法、多重共线性诊断、岭估计法与主成分估计法来进行数据分析。

1、最小二乘法

首先使用最小二乘法进行分析，结果如下：

```
economy=data.frame(y,x1,x2,x3)
lm.sol=lm(y~.,data=economy)
summary(lm.sol)
```

```
##
## Call:
## lm(formula = y ~ ., data = economy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12399.6  -1774.2   -419.6   2614.7   7258.1
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2427.82757 1559.41480   1.557   0.1311
## x1             0.15528    0.09493   1.636   0.1135
## x2             1.75391    0.09260  18.940   <2e-16 ***
## x3            -0.26641    0.14827  -1.797   0.0836 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4548 on 27 degrees of freedom
## Multiple R-squared:  0.9985, Adjusted R-squared:  0.9983
## F-statistic:  5957 on 3 and 27 DF,  p-value: < 2.2e-16
```

得到回归方程：

$$\hat{y} = 2427.82757 + 0.15528x_1 + 1.75391x_2 - 0.26641x_3$$

根据回归方程，x_1 和 x_2 前的回归系数都是正的，而 x_3 的回归系数为负。这与经济意义不符合（全国总消费量与总进口量理论上来讲是成正相关），所以说回归系数的符号与实际不符。

2、多重共线性诊断

接下来从多重共线性的角度考虑自变量之间的关系，考察样本的相关系数矩阵：

```
X=cbind(x1,x2,x3)
rho=cor(X)
rho
```

```
##           x1        x2        x3
## x1 1.0000000 0.9655464 0.9983580
## x2 0.9655464 1.0000000 0.9516582
## x3 0.9983580 0.9516582 1.0000000
```

从数据中可以看出 x_1, x_2, x_3 三者之间均互相存在着高度的线性相关性。

那么接下来进行多重共线性诊断：

```
library(DAAG)
vif(lm.sol)
```

```
##      x1      x2      x3
## 1331.50   46.31  955.74
```

```
eigen(rho)
```

```
## eigen() decomposition
## $values
## [1] 2.9438407073 0.0557273830 0.0004319097
##
## $vectors
##            [,1]       [,2]       [,3]
## [1,] -0.5813331  0.2961093  0.7578727
## [2,] -0.5720453 -0.8111161 -0.1218804
## [3,] -0.5786329  0.5043906 -0.6409166
```

```
kappa(rho,exact = TRUE)
```

```
## [1] 6815.871
```

3、岭估计法

接下来用岭估计方法寻找岭回归方程。

首先对数据进行标准化，代码如下：

```
yxs=scale(yx)
x1=yxs[,1]
x2=yxs[,2]
x3=yxs[,3]
y=yxs[,4]
economy2=data.frame(x1,x2,x3,y)
```

然后进行岭估计：

```
library(MASS)
```

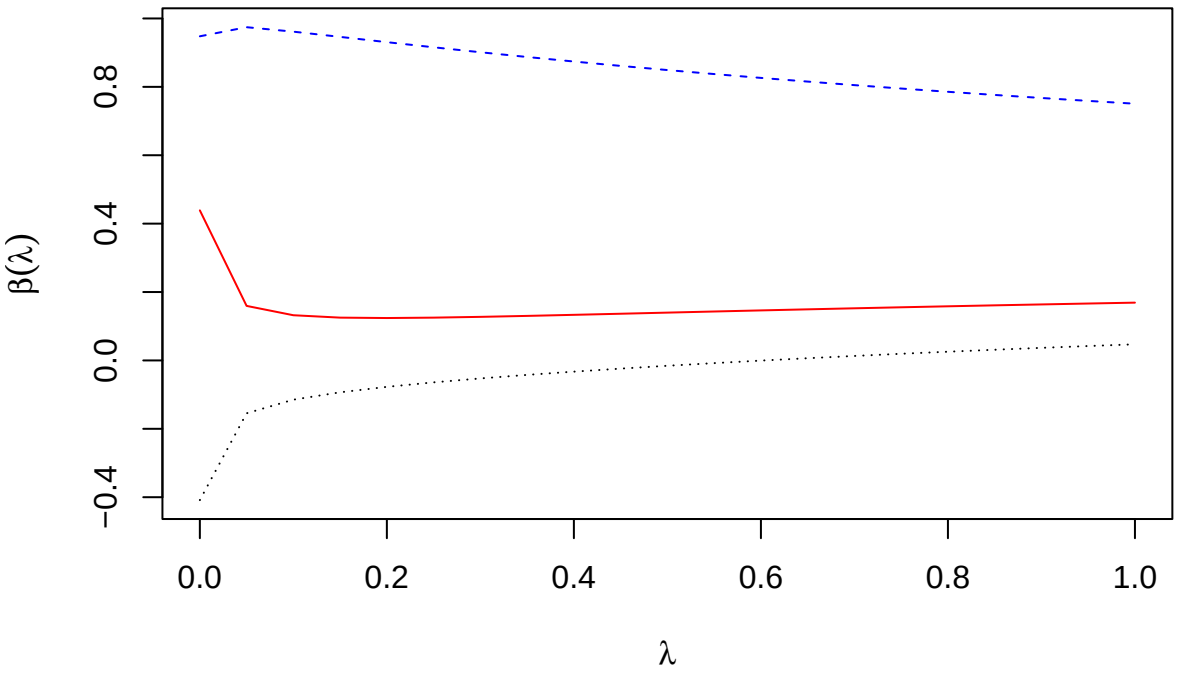
```
##
## 载入程辑包：'MASS'

## The following object is masked from 'package:DAAG':
##
##     hills
```

```
rr.sol=lm.ridge(y~0+x1+x2+x3,data=economy2,lambda=seq(0,1,by=0.05))
rr.sol
```

```
##              x1        x2            x3
## 0.00 0.4461390 0.9634790 -0.4152220141
## 0.05 0.1620989 0.9905061 -0.1571155080
## 0.10 0.1343163 0.9773316 -0.1167177128
## 0.15 0.1272051 0.9617615 -0.0947190669
## 0.20 0.1260097 0.9460967 -0.0785694062
## 0.25 0.1271971 0.9308402 -0.0652168935
## 0.30 0.1295027 0.9161370 -0.0535341734
## 0.35 0.1323683 0.9020238 -0.0429967939
## 0.40 0.1355136 0.8884979 -0.0333204106
## 0.45 0.1387857 0.8755403 -0.0243326431
## 0.50 0.1420963 0.8631256 -0.0159196435
## 0.55 0.1453925 0.8512260 -0.0080009850
## 0.60 0.1486422 0.8398139 -0.0005167305
## 0.65 0.1518254 0.8288619  0.0065797314
## 0.70 0.1549305 0.8183445  0.0133259020
## 0.75 0.1579505 0.8082370  0.0197527826
## 0.80 0.1608819 0.7985165  0.0258865639
## 0.85 0.1637235 0.7891616  0.0317497618
## 0.90 0.1664755 0.7801522  0.0373620080
## 0.95 0.1691391 0.7714696  0.0427406175
## 1.00 0.1717160 0.7630963  0.0479010068
```

画出岭迹图：

```
matplot(rr.sol$lambda,t(rr.sol$coef),type="l",col=c("red","blue","black"),xlab=expression(lambda),ylab=
```



根据岭迹图选择岭参数$k = 0.85$, 得到标准化变量的岭回归方程为：

$$\hat{v} = 0.1637u_1 + 0.7892u_2 + 0.0317u_3$$

转换成原始变量的岭回归方程：

$$\frac{\hat{y} - 133990}{111078.6} = 0.1637 \times \frac{x_1 - 334824.7}{319151.4} + 0.7892 \times \frac{x_2 - 72883.17}{61019.01} + 0.0317 \times \frac{x_3 - 181146.5}{173127.5}$$

即得到

$$\hat{y} = 6517.799 + 0.057x_1 + 1.437x_2 + 0.020x_3$$

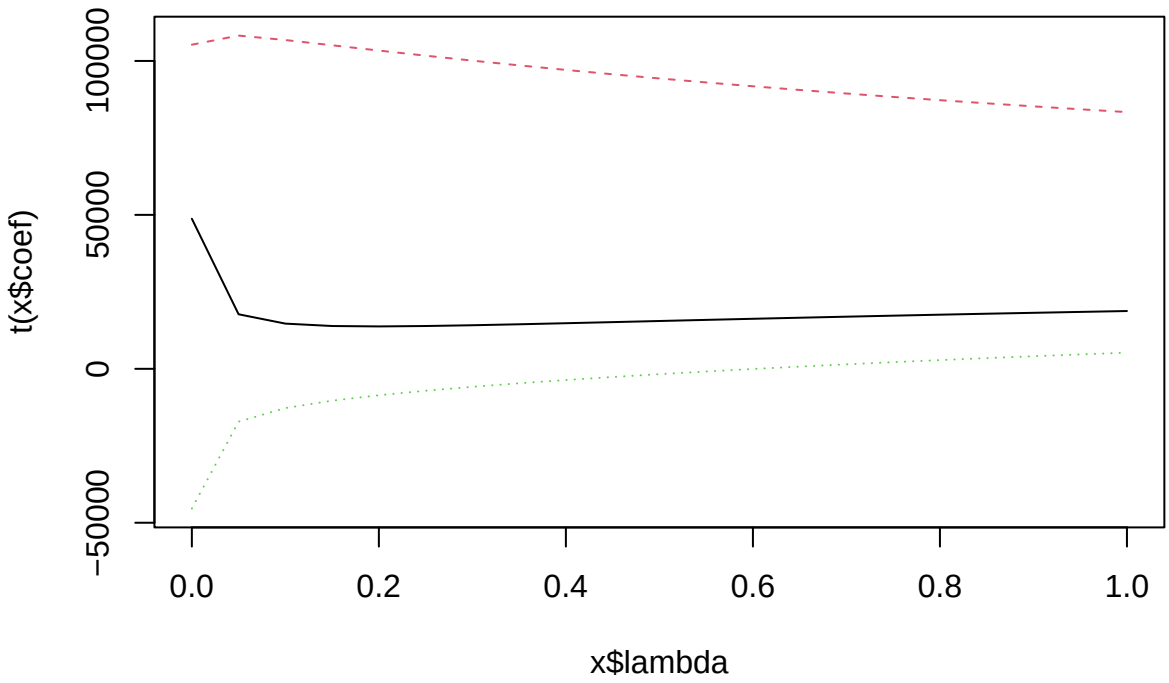
由于本案例中各个变量的量纲一致（均为亿元），故可以直接对原始数据进行岭估计：

```
rr.sol=lm.ridge(y~.,data=economy,lambda=seq(0,1,by=0.05))
rr.sol
```

```
##                       x1       x2            x3
## 0.00 2427.828 0.15527583 1.753911 -0.2664064390
## 0.05 1944.125 0.05641750 1.803111 -0.1008053080
## 0.10 2234.507 0.04674792 1.779128 -0.0748860831
## 0.15 2572.206 0.04427293 1.750784 -0.0607717521
## 0.20 2912.896 0.04385686 1.722268 -0.0504101300
```

```
## 0.25 3246.812 0.04427013 1.694495 -0.0418431580
## 0.30 3571.085 0.04507260 1.667730 -0.0343475250
## 0.35 3884.931 0.04606994 1.642038 -0.0275867424
## 0.40 4188.336 0.04716465 1.617416 -0.0213783750
## 0.45 4481.601 0.04830348 1.593828 -0.0156118235
## 0.50 4765.157 0.04945571 1.571228 -0.0102140431
## 0.55 5039.484 0.05060295 1.549566 -0.0051334319
## 0.60 5305.070 0.05173396 1.528792 -0.0003315343
## 0.65 5562.391 0.05284188 1.508855  0.0042215556
## 0.70 5811.904 0.05392256 1.489709  0.0085498985
## 0.75 6054.043 0.05497365 1.471309  0.0126733851
## 0.80 6289.214 0.05599392 1.453614  0.0166088191
## 0.85 6517.799 0.05698292 1.436585  0.0203706467
## 0.90 6740.155 0.05794073 1.420184  0.0239714638
## 0.95 6956.614 0.05886777 1.404378  0.0274223796
## 1.00 7167.490 0.05976467 1.389135  0.0307332854
```

```
plot(rr.sol)
```



同样根据岭迹图选择岭参数 $k = 0.85$, 得到岭回归方程：

$$\hat{y} = 6517.799 + 0.057x_1 + 1.437x_2 + 0.020x_3$$

与上面的结果一致！

上面已经完成了最小二乘估计与多重共线性诊断。

4、主成分估计法

为了消除多重共线性的影响，接下来我们进行主成分估计：

```
economy.pr=princomp(~x1+x2+x3,data=economy,cor=TRUE)
summary(economy.pr,loadings=TRUE)
```

```
## Importance of components:
##                            Comp.1     Comp.2       Comp.3
## Standard deviation     1.7157624 0.23606648 0.0207824374
## Proportion of Variance 0.9812802 0.01857579 0.0001439699
## Cumulative Proportion  0.9812802 0.99985603 1.0000000000
##
## Loadings:
##    Comp.1 Comp.2 Comp.3
## x1  0.581  0.296  0.758
## x2  0.572 -0.811 -0.122
## x3  0.579  0.504 -0.641
```

第三个特征根 $\lambda_3 = 0.0207824374^2 = 0.0004319 \approx 0$

对应三个标准正交化特征向量为：

$$\phi_1 = (0.581, 0.296, 0.758)'$$
$$\phi_2 = (0.572, -0.811, -0.122)'$$
$$\phi_3 = (0.579, 0.504, -0.641)'$$

三个主成分分别为：

$$z_1 = 0.581x_1^* + 0.296x_2^* + 0.758x_3^*$$
$$z_2 = 0.572x_1^* - 0.811x_2^* - 0.122x_3^*$$
$$z_3 = 0.579x_1^* + 0.504x_2^* - 0.641x_3^*$$

因为第一个特征根的累计贡献率为 $0.9812802 \geq 0.85$，所以删去后两个主成分，只保留第一个主成分。

计算主成分得分（即主成分的观测向量）：

```
pre=predict(economy.pr)
pre
```

```
##          Comp.1       Comp.2        Comp.3
## 1  -1.82575801  0.145606178  0.0157274091
## 2  -1.80645785  0.141971400  0.0155089610
## 3  -1.77987528  0.143132792  0.0164721188
## 4  -1.74283232  0.156457321  0.0184852745
## 5  -1.64445417  0.121427601  0.0111948418
## 6  -1.57464469  0.129464181  0.0078910949
## 7  -1.53063779  0.158079242  0.0069726743
## 8  -1.47634094  0.143719882  0.0041412680
## 9  -1.45216415  0.159747522  0.0023668994
## 10 -1.41575330  0.167454907 -0.0060099207
## 11 -1.33106184  0.137127409 -0.0182919237
## 12 -1.28152765  0.142876629 -0.0140568045
## 13 -1.19560099  0.103067383 -0.0187981516
## 14 -1.05880019  0.007983642 -0.0205789376
```

```
## 15 -0.85864033 -0.113548120 -0.0240074723
## 16 -0.63999303 -0.235591252 -0.0369602145
## 17 -0.39221323 -0.367351754 -0.0405529825
## 18 -0.06965416 -0.469914275 -0.0350203836
## 19  0.15774872 -0.452335801 -0.0097481251
## 20  0.09006720 -0.130249897  0.0393507646
## 21  0.53789293 -0.327894221  0.0403615270
## 22  0.97949003 -0.347691716  0.0280171980
## 23  1.23585805 -0.291699319  0.0223142311
## 24  1.51669021 -0.252969759  0.0203229544
## 25  1.78130176 -0.203591886  0.0107133944
## 26  1.95441719 -0.023792443 -0.0019648264
## 27  2.16688176  0.182759934 -0.0038303546
## 28  2.62266003  0.197705937  0.0006428284
## 29  3.05588206  0.280734763  0.0025600506
## 30  3.41691813  0.370441812 -0.0268615127
## 31  3.56060185  0.326871908 -0.0063618804
```

进行主成分估计：

```
z1=pre[,1]
yxs=scale(yx)
u=yxs[,4]
economynew=data.frame(u,z1)
pc.sol=lm(u~0+z1,data=economynew)
summary(pc.sol)
```

```
##
## Call:
## lm(formula = u ~ 0 + z1, data = economynew)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.3029 -0.1336 -0.1060  0.1893  0.3725
##
## Coefficients:
##    Estimate Std. Error t value Pr(>|t|)
## z1  0.56097    0.02163   25.93   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2067 on 30 degrees of freedom
## Multiple R-squared:  0.9573, Adjusted R-squared:  0.9559
## F-statistic: 672.3 on 1 and 30 DF,  p-value: < 2.2e-16
```

得到主成分回归方程：$\hat{u} = 0.56097 z_1$

$= 0.56097(0.581x_1^* + 0.296x_2^* + 0.758x_3^*)$

$= 0.325924x_1^* + 0.166047x_2^* + 0.425215x_3^*$

转化为原始变量的回归方程，得到：

$$\frac{\hat{y} - 133990}{111078.6} = 0.325924 \times \frac{x_1 - 334824.7}{319151.4} + 0.166047 \times \frac{x_2 - 72883.17}{61019.01} + 0.425215 \times \frac{x_3 - 181146.5}{173127.5}$$

即

$$\hat{y} = 24558.434 + 0.113x_1 + 0.302x_2 + 0.273x_3$$

这里的 y, x_1, x_2, x_3 表示原始变量。

实验结论与心得

由以上的结果可以看出，最小二乘法、岭估计法与主成分估计法得到的三个回归方程之间仍然存在着不小的差距。主要的原因就是选取的自变量与因变量之间的时候没有考虑到变量选取的合理性。这提醒了我将来在分析数据时，对于自变量的选择要更加注意（首先进行自变量的合理选取与模型的诊断）。但是其中，与最小而成估计相比，岭估计和主成分估计都一定程度上缓解了多重共线性带来的影响，因此 x_3 的回归系数的点估计的符号也发生了变化。

通过这次案例分析，我更加熟练了 r 语言的基本操作，并且能将其应用于经济生活方面相关的问题，感觉还是很有意义的。在庞老师上课讲解相关代码的基础上，加上自己动手操作，才能更加熟练 r 语言的使用。毕竟对于统计方向来说，r 语言无疑是一个很重要的软件。除此之外，这次实验让我对多重共线性、岭估计方法和主成分估计等知识有了更加深刻的了解。回归分析是一门很强大的学科，还有很多理论知识和方法需要我继续学习和掌握，这对于将来更好地分析和解释经济生活方面的现象具有重要意义。