

単回帰分析 — lm関数を用いる方法

オブジェクト名 <- lm(基準変数 ~ 説明変数, データフレーム名)
summary(オブジェクト名)

lm() でモデルを指定し, 結果を「オブジェクト名」に保存する. その内容をsummary() で表示する.

```
> setwd("i:¥¥Rdocuments¥¥scripts¥¥")
> d1 <- read.table("回帰分析データ.csv", header=TRUE, sep=",")
> head(d1)
  id stress kyoufu support utsu work result
1  1     20    2.2      17   18     0      0
2  2     23    4.8      18   21     1      0
3  3     30    5.8      12   29     1      1
4  4     25    5.2      18   29     0      1
5  5     26    2.0       8   22     1      0
6  6     21    5.0      26   19     1      0
>
>
> # 記述統計量
> dtmp <- d1[,c(-1)]
> ntmp <- nrow(dtmp)
> mtmp <- colMeans(dtmp)
> stmp <- apply(dtmp, 2, sd)
> ctmp <- cor(dtmp)
> ktmp <- round(data.frame(ntmp, mtmp, stmp, ctmp), 2)
> colnames(ktmp) <- c("N", "Mean", "SD", colnames(ctmp))
> ktmp
```

1	id	stress	kyoufu	support	utsu	work	result
2	1	20	2.2	17	18	0	0
3	2	23	4.8	18	21	1	0
4	3	30	5.8	12	29	1	1
5	4	25	5.2	18	29	0	1
6	5	26	2	8	22	1	0
7	6	21	5	26	19	1	0
8	7	14	2.2	24	12	0	0
9	8	22	4.4	17	19	1	0
10	9	26	4.2	11	27	0	1
11	10	26	4.2	18	18	1	0
12	11	21	2	27	18	0	0
13	12	24	4.8	19	29	1	1
14	13	24	3.4	23	19	1	0
15	14	23	2.2	10	24	1	0
16	15	23	4.4	24	20	0	0
17	16	35	7.2	12	31	0	1
18	17	25	3.2	17	19	1	0
19	18	33	3.4	14	26	0	1
20	19	30	4.4	20	30	1	1
21	20	19	5.6	18	12	1	0

```
      N Mean SD stress kyoufu support utsu work result
stress 245 22.94 5.25 1.00 0.40 -0.34 0.62 0.03 0.44
kyoufu 245 4.05 1.17 0.40 1.00 -0.03 0.31 0.10 0.20
support 245 18.42 4.96 -0.34 -0.03 1.00 -0.51 -0.03 -0.39
utsu    245 20.29 6.49 0.62 0.31 -0.51 1.00 0.02 0.76
work    245 0.50 0.50 0.03 0.10 -0.03 0.02 1.00 -0.08
result  245 0.26 0.44 0.44 0.20 -0.39 0.76 -0.08 1.00
>
>
```

```
> # 単回帰分析
> reg.1 <- lm(utsu ~ stress, data=d1)
> summary(reg.1)
```

Call:
lm(formula = utsu ~ stress, data = d1)

Residuals:

	Min	1Q	Median	3Q	Max
	-11.6277	-3.2181	-0.0374	3.0771	15.6674

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.73611	1.46777	1.864	0.0635 .
stress	0.76506	0.06238	12.264	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.114 on 243 degrees of freedom
Multiple R-squared: 0.3823, Adjusted R-squared: 0.3798
F-statistic: 150.4 on 1 and 243 DF, p-value: < 2.2e-16

```
>
>
> # 偏回帰係数の信頼区間
> confint(reg.1)
      2.5 %      97.5 %
(Intercept) -0.1550688 5.6272812
stress      0.6421862 0.8879398
>
>
```

```

> #散布図と回帰直線を描く
> plot(d1$stress, d1$utsu, xlim=c(0,40), ylim=c(0,40),
+      pch=20, las=1, xlab="ストレス", ylab="うつ傾向")
>
> #回帰直線
> abline(reg.1, lwd=2)
> (a <- reg.1$coefficients["(Intercept)"])
(Intercept)
2.736106
> (b <- reg.1$coefficients["stress"])
stress
0.765063
> segments(-1, 38, 0.5, 38, lwd=2, lty=1)
> text(1, 38, paste("Y = ", round(a, 2), " + ", round(b, 2), " X", sep=""), pos=4)
>
>
> # データの標準化
> d2 <- as.data.frame(scale(d1[, c("stress", "kyoufu", "support", "utsu")]))
>
> #共分散行列の確認
> cov(d2)

```

```

      stress      kyoufu      support      utsu
stress  1.0000000  0.40369513 -0.33786517  0.6183281
kyoufu  0.4036951  1.00000000 -0.02670051  0.3124069
support -0.3378652 -0.02670051  1.00000000 -0.5064285
utsu     0.6183281  0.31240691 -0.50642854  1.0000000

```

```

> #標準偏回帰係数の推定
> reg.2 <- lm(utsu ~ stress, data=d2)
> summary(reg.2)

```

```

Call:
lm(formula = utsu ~ stress, data = d2)

```

```

Residuals:
    Min       1Q   Median       3Q      Max
-1.79071 -0.49560 -0.00575  0.47388  2.41283

```

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.448e-16  5.031e-02   0.00      1
stress       6.183e-01  5.042e-02  12.26 <2e-16 ***
---

```

```

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 0.7875 on 243 degrees of freedom
Multiple R-squared:  0.3823,    Adjusted R-squared:  0.3798
F-statistic: 150.4 on 1 and 243 DF,  p-value: < 2.2e-16

```

```

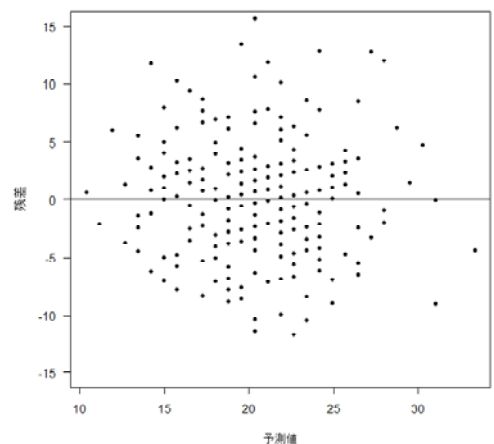
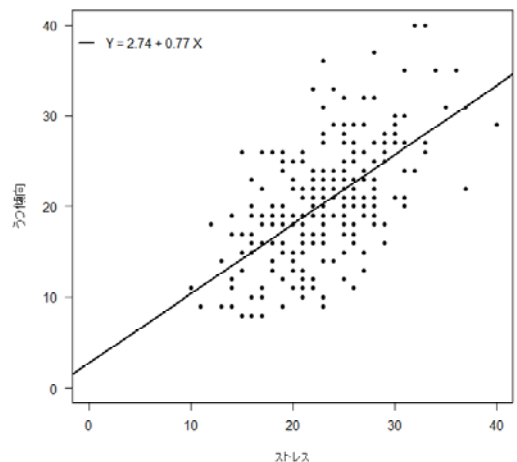
>
> # (標準) 偏回帰係数の信頼区間
> confint(reg.2)
      2.5 %      97.5 %
(Intercept) -0.09910683 0.09910683
stress      0.51901839 0.71763782

```

```

> # 残差プロット
> plot(reg.1$fitted.values, reg.1$residuals, pch=20, las=1,
+      ylim=c(-15, 15), xlab="予測値", ylab="残差", main="")
> segments(5, 0, 40, 0)
>
>

```



重回帰分析 — lm関数を使う方法

偏回帰係数の推定と信頼区間

```
オブジェクト名1 <- lm(基準変数 ~ 説明変数1 + 説明変数2 + ..., データフレーム名1)
summary(オブジェクト名1)
confint(オブジェクト名1)
```

標準偏回帰係数の推定と信頼区間

```
データフレーム名2 <- as.data.frame(scale(データフレーム名1))
オブジェクト名2 <- lm(基準変数 ~ 説明変数1 + 説明変数2 + ..., データフレーム名2)
summary(オブジェクト名2)
confint(オブジェクト名2)
```

多重共線性の診断

```
VIF <- diag(solve(cor(説明変数間の相関係数行列)))
VIF      # VIF(variance inflaion factor)
1/VIF    # torelance
```

または

```
library(car)
vif(lmオブジェクト名)
```

あらかじめcarパッケージをインストールしておく必要がある。

ステップワイズ分析

```
library(MASS)
オブジェクト名 <- stepAIC(lmオブジェクト名)
summary(オブジェクト名)
```

MASSパッケージは最初からインストールされている。

```
> setwd("i:¥¥Rdocuments¥¥scripts¥¥")
> d1 <- read.table("回帰分析データ.csv", header=TRUE, sep=",")
> head(d1)
```

```
  id stress kyoufu support utsu work result
1  1    20    2.2    17   18     0      0
2  2    23    4.8    18   21     1      0
3  3    30    5.8    12   29     1      1
4  4    25    5.2    18   29     0      1
5  5    26    2.0     8   22     1      0
6  6    21    5.0    26   19     1      0
```

```
>
>
> # 記述統計量
> dtmp <- d1[, c(-1)]
> ntmp <- nrow(dtmp)
> mtmp <- colMeans(dtmp)
> stmp <- apply(dtmp, 2, sd)
> ctmp <- cor(dtmp)
> ktmp <- round(data.frame(ntmp, mtmp, stmp, ctmp), 2)
> colnames(ktmp) <- c("N", "Mean", "SD", colnames(ctmp))
> ktmp
```

```
      N Mean  SD stress kyoufu support  utsu  work result
stress 245 22.94 5.25  1.00  0.40 -0.34  0.62  0.03  0.44
kyoufu 245  4.05 1.17  0.40  1.00 -0.03  0.31  0.10  0.20
support 245 18.42 4.96 -0.34 -0.03  1.00 -0.51 -0.03 -0.39
utsu    245 20.29 6.49  0.62  0.31 -0.51  1.00  0.02  0.76
work    245  0.50 0.50  0.03  0.10 -0.03  0.02  1.00 -0.08
result  245  0.26 0.44  0.44  0.20 -0.39  0.76 -0.08  1.00
```

```
> #偏回帰係数の推定
> result.1 <- lm(utsu ~ stress + kyoufu, data=d1)
> summary(result.1)
```

1	id	stress	kyoufu	support	utsu	work	result
2	1	20	2.2	17	18	0	0
3	2	23	4.8	18	21	1	0
4	3	30	5.8	12	29	1	1
5	4	25	5.2	18	29	0	1
6	5	26	2	8	22	1	0
7	6	21	5	26	19	1	0
8	7	14	2.2	24	12	0	0
9	8	22	4.4	17	19	1	0
10	9	26	4.2	11	27	0	1
11	10	26	4.2	18	18	1	0
12	11	21	2	27	18	0	0
13	12	24	4.8	19	29	1	1
14	13	24	3.4	23	19	1	0
15	14	23	2.2	10	24	1	0
16	15	23	4.4	24	20	0	0
17	16	35	7.2	12	31	0	1
18	17	25	3.2	17	19	1	0
19	18	33	3.4	14	26	0	1
20	19	30	4.4	20	30	1	1
21	20	19	5.6	18	12	1	0

```

Call:
lm(formula = utsu ~ stress + kyoufu, data = d1)

Residuals:
    Min       1Q   Median       3Q      Max
-11.3092  -3.2097  -0.0432   3.1134  16.0238

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.90963    1.58558   1.204   0.230
stress       0.72759    0.06806  10.690 <2e-16 ***
kyoufu       0.41623    0.30522   1.364   0.174
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.105 on 242 degrees of freedom
Multiple R-squared:  0.387,    Adjusted R-squared:  0.382
F-statistic: 76.4 on 2 and 242 DF,  p-value: < 2.2e-16

> confint(result.1)
              2.5 %    97.5 %
(Intercept) -1.2136587 5.0329271
stress       0.5935196 0.8616657
kyoufu      -0.1850019 1.0174659
>
>
> #標準偏回帰係数の推定
> d2 <- as.data.frame(scale(d1[,c("stress", "kyoufu", "utsu", "work")]))
> #共分散行列の確認
> round(cov(d2), 2)
      stress kyoufu utsu work
stress  1.00   0.40 0.62 0.03
kyoufu  0.40   1.00 0.31 0.10
utsu    0.62   0.31 1.00 0.02
work    0.03   0.10 0.02 1.00
> result.2 <- lm(utsu ~ stress + kyoufu, data=d2)
> summary(result.2)

Call:
lm(formula = utsu ~ stress + kyoufu, data = d2)

Residuals:
    Min       1Q   Median       3Q      Max
-1.74165 -0.49430 -0.00665  0.47948  2.46771

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.235e-16  5.023e-02   0.000   1.000
stress      5.880e-01  5.501e-02  10.690 <2e-16 ***
kyoufu      7.502e-02  5.501e-02   1.364   0.174
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7861 on 242 degrees of freedom
Multiple R-squared:  0.387,    Adjusted R-squared:  0.382
F-statistic: 76.4 on 2 and 242 DF,  p-value: < 2.2e-16

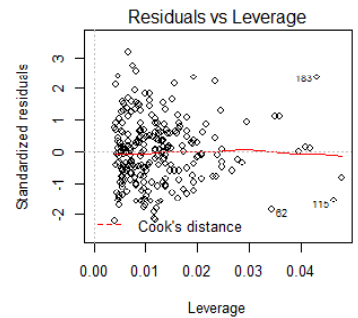
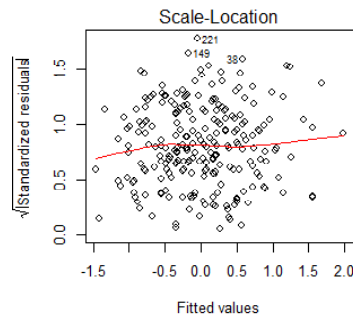
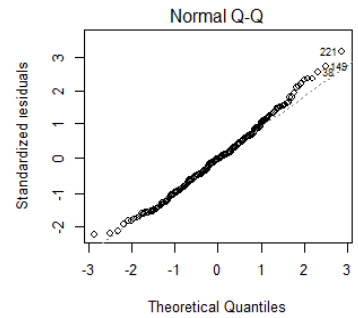
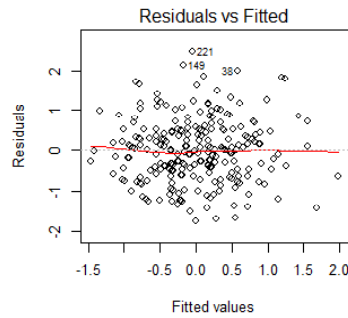
> confint(result.2)
              2.5 %    97.5 %
(Intercept) -0.09893404 0.09893404
stress       0.47968584 0.69640299
kyoufu      -0.03334233 0.18337482
>
>
> #多重共線性の確認
> # carパッケージのvifを使う方法
> library(car)
> VIF <- vif(result.2)
> TOLERANCE <- 1/VIF
> round(data.frame(VIF, TOLERANCE), 3)
      VIF TOLERANCE

```

```

stress 1.195      0.837
kyoufu 1.195      0.837
>
> #説明変数だけのデータフレームを作成
> d3 <- d1[,c("stress", "kyoufu")]
> VIF <- diag(solve(cor(d3)))
> tolerance <- 1/VIF
> data.frame(VIF, tolerance)
      VIF tolerance
stress 1.1947 0.8370302
kyoufu 1.1947 0.8370302
>
>
>
> #残差プロットなどを描く
> #図を4枚同時に並べる指定
> par(mfrow=c(2, 2))
> plot(result.2)
> par(mfrow=c(1, 1))
>
> #ステップワイズ分析
> #MASSパッケージの読み込み
> library(MASS)
> result.3 <- stepAIC(result.2)
Start:  AIC=-114.92
utsu ~ stress + kyoufu

```



```

      Df Sum of Sq    RSS    AIC
- kyoufu  1      1.149 150.71 -115.043
<none>                 149.56 -114.919
- stress  1     70.624 220.19  -22.162

```

```

Step:  AIC=-115.04
utsu ~ stress

```

```

      Df Sum of Sq    RSS    AIC
<none>                 150.71 -115.043
- stress  1     93.288 244.00    0.998
> summary(result.3)

```

```

Call:
lm(formula = utsu ~ stress, data = d2)

```

```

Residuals:
    Min       1Q   Median       3Q      Max
-1.79071 -0.49560 -0.00575  0.47388  2.41283

```

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.448e-16  5.031e-02   0.00    1
stress      6.183e-01  5.042e-02  12.26 <2e-16 ***
---

```

```

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 0.7875 on 243 degrees of freedom
Multiple R-squared:  0.3823,    Adjusted R-squared:  0.3798
F-statistic: 150.4 on 1 and 243 DF,  p-value: < 2.2e-16

```

```

> confint(result.3)
              2.5 %      97.5 %
(Intercept) -0.09910683 0.09910683
stress      0.51901839 0.71763782
>
>
>

```

重回帰分析 — glm関数を使う方法

オブジェクト名 <- glm(基準変数 ~ 説明変数, family=gaussian, データフレーム名)
summary(オブジェクト名)

glm() でモデルを指定し、結果を「オブジェクト名」に保存する。その内容をsummary()で表示する。

```
> setwd("i:\\Rdocuments\\scripts\\")
> d1 <- read.table("回帰分析データ.csv", header=TRUE, sep=",")
> head(d1)
```

```
  id stress kyoufu support utsu work result
1  1     20    2.2      17   18     0      0
2  2     23    4.8      18   21     1      0
3  3     30    5.8      12   29     1      1
4  4     25    5.2      18   29     0      1
5  5     26    2.0       8   22     1      0
6  6     21    5.0      26   19     1      0
```

```
> # 記述統計量
```

```
> dtmp <- d1[, c(-1)]
> ntmp <- nrow(dtmp)
> mtmp <- colMeans(dtmp)
> stmp <- apply(dtmp, 2, sd)
> ctmp <- cor(dtmp)
> ktmp <- round(data.frame(ntmp, mtmp, stmp, ctmp), 2)
> colnames(ktmp) <- c("N", "Mean", "SD", colnames(ctmp))
> ktmp
```

```
      N Mean  SD stress kyoufu support  utsu  work result
stress 245 22.94 5.25  1.00  0.40 -0.34  0.62  0.03  0.44
kyoufu 245  4.05 1.17  0.40  1.00 -0.03  0.31  0.10  0.20
support 245 18.42 4.96 -0.34 -0.03  1.00 -0.51 -0.03 -0.39
utsu    245 20.29 6.49  0.62  0.31 -0.51  1.00  0.02  0.76
work    245  0.50 0.50  0.03  0.10 -0.03  0.02  1.00 -0.08
result  245  0.26 0.44  0.44  0.20 -0.39  0.76 -0.08  1.00
```

```
> #偏回帰係数の推定
```

```
> result.1 <- glm(utsu ~ stress + kyoufu + support, family=gaussian, data=d1)
> summary(result.1)
```

Call:

```
glm(formula = utsu ~ stress + kyoufu + support, family = gaussian,
    data = d1)
```

Deviance Residuals:

```
      Min       1Q   Median       3Q      Max
-11.0197  -3.3419   0.1646   3.0142  11.7647
```

Coefficients:

```
      Estimate Std. Error t value Pr(>|t|)
(Intercept) 13.22868    2.14218   6.175 2.78e-09 ***
stress       0.55796    0.06634   8.411 3.61e-15 ***
kyoufu       0.67134    0.28009   2.397  0.0173 *
support     -0.45934    0.06424  -7.150 1.03e-11 ***
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for gaussian family taken to be 21.58741)

```
Null deviance: 10288.0 on 244 degrees of freedom
Residual deviance: 5202.6 on 241 degrees of freedom
AIC: 1453.9
```

Number of Fisher Scoring iterations: 2

1	id	stress	kyoufu	support	utsu	work	result
2	1	20	2.2	17	18	0	0
3	2	23	4.8	18	21	1	0
4	3	30	5.8	12	29	1	1
5	4	25	5.2	18	29	0	1
6	5	26	2	8	22	1	0
7	6	21	5	26	19	1	0
8	7	14	2.2	24	12	0	0
9	8	22	4.4	17	19	1	0
10	9	26	4.2	11	27	0	1
11	10	26	4.2	18	18	1	0
12	11	21	2	27	18	0	0
13	12	24	4.8	19	29	1	1
14	13	24	3.4	23	19	1	0
15	14	23	2.2	10	24	1	0
16	15	23	4.4	24	20	0	0
17	16	35	7.2	12	31	0	1
18	17	25	3.2	17	19	1	0
19	18	33	3.4	14	26	0	1
20	19	30	4.4	20	30	1	1
21	20	19	5.6	18	12	1	0

```

> confint(result.1)
Waiting for profiling to be done...
              2.5 %      97.5 %
(Intercept)  9.0300832 17.4272733
stress       0.4279414  0.6879808
kyoufu       0.1223773  1.2203088
support      -0.5852586 -0.3334245
>
>
> #標準偏回帰係数の推定
> d2 <- as.data.frame(scale(d1[,c("stress", "kyoufu", "support", "utsu")]))
> #共分散行列の確認
> round(cov(d2), 2)
      stress kyoufu support  utsu
stress    1.00   0.40   -0.34   0.62
kyoufu    0.40   1.00   -0.03   0.31
support  -0.34  -0.03   1.00  -0.51
utsu      0.62   0.31  -0.51   1.00

> result.2 <- glm(utsu ~ stress + kyoufu + support, family=gaussian, data=d2)
> summary(result.2)

Call:
glm(formula = utsu ~ stress + kyoufu + support, family = gaussian,
    data = d2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.69706  -0.51466   0.02535   0.46419   1.81181

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.334e-17  4.571e-02   0.000   1.0000
stress       4.509e-01  5.361e-02   8.411 3.61e-15 ***
kyoufu       1.210e-01  5.048e-02   2.397  0.0173 *
support     -3.508e-01  4.907e-02  -7.150 1.03e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.5119875)

Null deviance: 244.00  on 244  degrees of freedom
Residual deviance: 123.39  on 241  degrees of freedom
AIC: 537.23

Number of Fisher Scoring iterations: 2

> confint(result.2)
Waiting for profiling to be done...
              2.5 %      97.5 %
(Intercept) -0.08959726  0.08959726
stress       0.34586463  0.55602983
kyoufu       0.02205570  0.21993258
support      -0.44701225 -0.25466489
>
>
>
>

```

重回帰分析 — sem関数を使う方法

パッケージの読み込み

```
library(sem)
```

モデルの設定

specifyEquations を使う方法

```
モデル名 <- specifyEquations()
```

```
予測式
```

```
V(変数名.) = 分散, 誤差分散を表すパラメタ (または設定値)
```

```
C(変数名1, 変数名2) = 共分散を表すパラメタ (または設定値)
```

specifyModel を使う方法

```
モデル名 <- specifyModel()
```

```
各パラメタ (または設定値) を設定する式
```

パラメタ値の推定

```
semオブジェクト名 <- sem(モデル名, 共分散行列 (またはデータ行列), N=標本サイズ)
```

```
summary(semオブジェクト名)
```

標準化解の推定

```
stdCoef(semオブジェクト名)
```

あらかじめsemパッケージをインストールしておく必要がある。
 (行をあけると、モデル設定が終了したと判断されてしまう)
 (逆に、行をあけないと、モデル設定が終了したことになる)

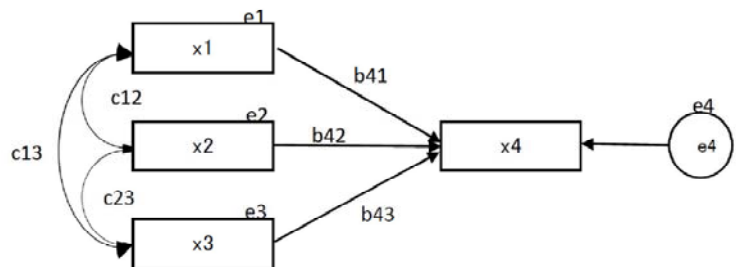
モデル部分のスキプトの例 (重回帰モデル $x_4 = b_{41} * x_1 + b_{42} * x_2 + b_{43} * x_3 + e_4$)

```
# specifyEquations を使う方法
```

```
seq.1 <- specifyEquations()
x4 = b41*x1 + b42*x2 + b43*x3      # 予測式
V(x4) = ev4                        # 内生変数の誤差分散
V(x1) = v1                          # 外生変数の分散
V(x2) = v2
V(x3) = v3
C(x1, x2) = c12                    # 外生変数の共分散
C(x1, x3) = c13
C(x2, x3) = c23
```

```
# specifyModel を使う方法
```

```
smd.1 <- specify.model()
x4 <- x1, b41, NA
x4 <- x2, b42, NA
x4 <- x3, b43, NA
x4 <-> x4, ev4, NA
x1 <-> x1, v1, NA
x2 <-> x2, v2, NA
x3 <-> x3, v3, NA
x1 <-> x2, c12, NA
x1 <-> x3, c13, NA
x2 <-> x3, c23, NA
```



```
# 1行目: x4に対するx1の偏回帰係数をb41とする (NAは、「初期値を特に指定しない」の意. 以下同様)
# 2行目: x4に対するx2の偏回帰係数をb42とする
# 3行目: x4に対するx3の偏回帰係数をb43とする
# 4行目: x4の誤差分散をev4とする
# 5行目: x1の分散をv1とする
# 6行目: x2の分散をv2とする
# 7行目: x3の分散をv3とする
# 8行目: x1とx2の共分散をc12とする
# 9行目: x1とx3の共分散をc13とする
#10行目: x2とx3の共分散をc23とする
```

```
> setwd("i:¥¥Rdocuments¥¥scripts¥¥")
> d1 <- read.table("回帰分析データ.csv", header=TRUE, sep=",")
> head(d1)
  id stress kyoufu support utsu work result
1  1     20    2.2      17   18     0      0
2  2     23    4.8      18   21     1      0
3  3     30    5.8      12   29     1      1
4  4     25    5.2      18   29     0      1
5  5     26    2.0       8   22     1      0
6  6     21    5.0      26   19     1      0
```

1	id	stress	kyoufu	support	utsu	work	result
2	1	20	2.2	17	18	0	0
3	2	23	4.8	18	21	1	0
4	3	30	5.8	12	29	1	1
5	4	25	5.2	18	29	0	1
6	5	26	2	8	22	1	0
7	6	21	5	26	19	1	0
8	7	14	2.2	24	12	0	0
9	8	22	4.4	17	19	1	0
10	9	26	4.2	11	27	0	1
11	10	26	4.2	18	18	1	0
12	11	21	2	27	18	0	0
13	12	24	4.8	19	29	1	1
14	13	24	3.4	23	19	1	0
15	14	23	2.2	10	24	1	0
16	15	23	4.4	24	20	0	0
17	16	35	7.2	12	31	0	1
18	17	25	3.2	17	19	1	0
19	18	33	3.4	14	26	0	1
20	19	30	4.4	20	30	1	1
21	20	19	5.6	18	12	1	0

```
> # 記述統計量
> dtmp <- d1[,c(-1)]
> ntmp <- nrow(dtmp)
> mtmp <- colMeans(dtmp)
> stmp <- apply(dtmp, 2, sd)
> ctmp <- cor(dtmp)
> ktmp <- round(data.frame(ntmp, mtmp, stmp, ctmp), 2)
> colnames(ktmp) <- c("N", "Mean", "SD", colnames(ctmp))
> ktmp
      N Mean  SD stress kyoufu support  utsu  work result
stress 245 22.94 5.25   1.00   0.40  -0.34  0.62  0.03  0.44
kyoufu 245  4.05 1.17   0.40   1.00  -0.03  0.31  0.10  0.20
support 245 18.42 4.96  -0.34  -0.03   1.00 -0.51 -0.03 -0.39
utsu    245 20.29 6.49   0.62   0.31  -0.51  1.00  0.02  0.76
work    245  0.50 0.50   0.03   0.10  -0.03  0.02  1.00 -0.08
result  245  0.26 0.44   0.44   0.20  -0.39  0.76 -0.08  1.00
```

```
> #変数名の変更 (式を書くのにはこのほうが簡単)
> # x1:ストレス x2:失敗恐怖 x3:ソーシャルサポート x4:うつ傾向
> colnames(d1) <- c("id", "x1", "x2", "x3", "x4", "work", "result")
> head(d1)
```

```
  id x1 x2 x3 x4 work result
1  1 20 2.2 17 18     0      0
2  2 23 4.8 18 21     1      0
3  3 30 5.8 12 29     1      1
4  4 25 5.2 18 29     0      1
5  5 26 2.0  8 22     1      0
6  6 21 5.0 26 19     1      0
```

```
> # 共分散行列・相関係数行列
> cov.d1 <- cov(d1[,c("x1", "x2", "x3", "x4")])
> cor.d1 <- cor(d1[,c("x1", "x2", "x3", "x4")])
>
>
> #sem パッケージの読み込み
> library(sem)
>
> # 適合度指標出力の指定
> opt <- options(fit.indices = c("GFI", "AGFI", "RMSEA", "NFI", "NNFI", "CFI", "RNI", "IFI", "SRMR", "AIC", "AICc", "BIC", "CAIC"))
>
> # 構造方程式モデリング
> #  $x_4 = b_{41} * x_1 + b_{42} * x_2 + b_{43} * x_3 + e_4$ 
>
> # specifyEquations を使う方法
> # モデルの設定
> seq.1 <- specifyEquations()
1:  $x_4 = b_{41} * x_1 + b_{42} * x_2 + b_{43} * x_3$ 
2:  $V(x_4) = ev_4$ 
3:  $V(x_1) = v_1$ 
4:  $V(x_2) = v_2$ 
5:  $V(x_3) = v_3$ 
```

```
6: C(x1, x2) = c12
7: C(x1, x3) = c13
8: C(x2, x3) = c23
9:
Read 8 items
```

```
> # モデルの当てはめ（共分散行列）
```

```
> sem.seq.1 <- sem(seq.1, cov.d1, N=nrow(d1))
> summary(sem.seq.1)
```

```
Model Chisquare = 0    Df = 0 Pr(>Chisq) = NA
Goodness-of-fit index = 1
AIC = 20
AICc = 0.9401709
BIC = 0
CAIC = 0
```

```
Normalized Residuals
      Min.      1st Qu.      Median      Mean      3rd Qu.      Max.
-1.861e-15  0.000e+00  0.000e+00 -1.675e-17  0.000e+00  8.713e-16
```

```
R-square for Endogenous Variables
      x4
0.4943
```

```
Parameter Estimates
      Estimate Std Error z value Pr(>|z|)
b41  0.5579611 0.06592870  8.4630984 2.603658e-17 x4 <--- x1
b42  0.6713431 0.27836253  2.4117582 1.587581e-02 x4 <--- x2
b43 -0.4593416 0.06384842 -7.1942506 6.280474e-13 x4 <--- x3
ev4  21.3219891 1.93040219 11.0453610 2.308379e-28 x4 <--> x4
v1   27.5413182 2.49347379 11.0453610 2.308379e-28 x1 <--> x1
v2   1.3695590 0.12399405 11.0453610 2.308379e-28 x2 <--> x2
v3   24.5971228 2.22691886 11.0453610 2.308379e-28 x3 <--> x3
c12  2.4793409 0.42400604  5.8474189 4.992591e-09 x2 <--> x1
c13 -8.7938274 1.75878193 -4.9999532 5.734423e-07 x3 <--> x1
c23 -0.1549716 0.37169984 -0.4169266 6.767321e-01 x3 <--> x2
```

```
Iterations = 0
>
```

```
> # 標準化解の表示
```

```
> (sc.sem.sew.1 <- stdCoeef(sem.seq.1))
      Std. Estimate
b41 b41  0.45094723 x4 <--- x1
b42 b42  0.12099414 x4 <--- x2
b43 b43 -0.35083857 x4 <--- x3
ev4 ev4  0.50569259 x4 <--> x4
v1  v1  1.00000000 x1 <--> x1
v2  v2  1.00000000 x2 <--> x2
v3  v3  1.00000000 x3 <--> x3
c12 c12  0.40369513 x2 <--> x1
c13 c13 -0.33786517 x3 <--> x1
c23 c23 -0.02670051 x3 <--> x2
>
```

```
> # モデルの当てはめ（相関係数行列）
```

```
> sem.seq.1 <- sem(seq.1, cor.d1, N=nrow(d1))
> summary(sem.seq.1)
```

```
Model Chisquare = -5.417888e-14    Df = 0 Pr(>Chisq) = NA
Goodness-of-fit index = 1
AIC = 20
AICc = 0.9401709
BIC = -5.417888e-14
CAIC = -5.417888e-14
```

```
Normalized Residuals
      Min.      1st Qu.      Median      Mean      3rd Qu.      Max.
-1.475e-15  0.000e+00  0.000e+00  9.014e-18  0.000e+00  1.547e-15
```

R-square for Endogenous Variables

x4

0.4943

Parameter Estimates

	Estimate	Std Error	z value	Pr(> z)	
b41	0.45094723	0.05328394	8.4630984	2.603658e-17	x4 <--- x1
b42	0.12099414	0.05016844	2.4117582	1.587581e-02	x4 <--- x2
b43	-0.35083857	0.04876652	-7.1942506	6.280474e-13	x4 <--- x3
ev4	0.50569259	0.04578326	11.0453610	2.308379e-28	x4 <--> x4
v1	1.00000000	0.09053575	11.0453610	2.308379e-28	x1 <--> x1
v2	1.00000000	0.09053575	11.0453610	2.308379e-28	x2 <--> x2
v3	1.00000000	0.09053575	11.0453610	2.308379e-28	x3 <--> x3
c12	0.40369513	0.06903817	5.8474189	4.992591e-09	x2 <--> x1
c13	-0.33786517	0.06757367	-4.9999532	5.734423e-07	x3 <--> x1
c23	-0.02670051	0.06404126	-0.4169266	6.767321e-01	x3 <--> x2

Iterations = 0

```
> (sc.sem.seq.1 <- stdCoef(sem.seq.1))
```

	Std. Estimate	
b41 b41	0.45094723	x4 <--- x1
b42 b42	0.12099414	x4 <--- x2
b43 b43	-0.35083857	x4 <--- x3
ev4 ev4	0.50569259	x4 <--> x4
v1 v1	1.00000000	x1 <--> x1
v2 v2	1.00000000	x2 <--> x2
v3 v3	1.00000000	x3 <--> x3
c12 c12	0.40369513	x2 <--> x1
c13 c13	-0.33786517	x3 <--> x1
c23 c23	-0.02670051	x3 <--> x2

```
> # specifyModel を使う方法
```

```
> # モデルの設定
```

```
> smd.1 <- specifyModel()
```

```
1: x4 <- x1, b41, NA
2: x4 <- x2, b42, NA
3: x4 <- x3, b43, NA
4: x4 <-> x4, ev4, NA
5: x1 <-> x1, v1, NA
6: x2 <-> x2, v2, NA
7: x3 <-> x3, v3, NA
8: x1 <-> x2, c12, NA
9: x1 <-> x3, c13, NA
10: x2 <-> x3, c23, NA
11:
```

Read 10 records

NOTE: it is generally simpler to use specifyEquations() or cfa()
see ?specifyEquations

```
> # モデルの当てはめ (相関係数行列)
```

```
> sem.smd.1 <- sem(smd.1, cor.d1, N=nrow(d1))
```

```
> summary(sem.smd.1)
```

Model Chisquare = -5.417888e-14 Df = 0 Pr(>Chisq) = NA

Goodness-of-fit index = 1

AIC = 20

AICc = 0.9401709

BIC = -5.417888e-14

CAIC = -5.417888e-14

Normalized Residuals

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-1.475e-15	0.000e+00	0.000e+00	9.014e-18	0.000e+00	1.547e-15

R-square for Endogenous Variables

x4

0.4943

Parameter Estimates

	Estimate	Std Error	z value	Pr(> z)		
b41	0.45094723	0.05328394	8.4630984	2.603658e-17	x4 <---	x1
b42	0.12099414	0.05016844	2.4117582	1.587581e-02	x4 <---	x2
b43	-0.35083857	0.04876652	-7.1942506	6.280474e-13	x4 <---	x3
ev4	0.50569259	0.04578326	11.0453610	2.308379e-28	x4 <-->	x4
v1	1.00000000	0.09053575	11.0453610	2.308379e-28	x1 <-->	x1
v2	1.00000000	0.09053575	11.0453610	2.308379e-28	x2 <-->	x2
v3	1.00000000	0.09053575	11.0453610	2.308379e-28	x3 <-->	x3
c12	0.40369513	0.06903817	5.8474189	4.992591e-09	x2 <-->	x1
c13	-0.33786517	0.06757367	-4.9999532	5.734423e-07	x3 <-->	x1
c23	-0.02670051	0.06404126	-0.4169266	6.767321e-01	x3 <-->	x2

Iterations = 0

> (sc.sem.smd.1 <- stdCoef(sem.smd.1))

	Std. Estimate		
b41 b41	0.45094723	x4 <---	x1
b42 b42	0.12099414	x4 <---	x2
b43 b43	-0.35083857	x4 <---	x3
ev4 ev4	0.50569259	x4 <-->	x4
v1 v1	1.00000000	x1 <-->	x1
v2 v2	1.00000000	x2 <-->	x2
v3 v3	1.00000000	x3 <-->	x3
c12 c12	0.40369513	x2 <-->	x1
c13 c13	-0.33786517	x3 <-->	x1
c23 c23	-0.02670051	x3 <-->	x2

主成分得点を用いた重回帰分析

重回帰分析において、説明変数間の相関が高いときに、無相関ないくつかの主成分を説明変数にして、重回帰分析を行う。説明変数が無相関になるという利点がある。

この方法と、「主成分回帰分析」と言われるものとは、解釈するものが異なる。主成分回帰分析では、いくつかの主成分を採用したとき、結局もとの変数がどう基準変数に影響するかを解釈する。つまり、もとの説明変数に対する回帰係数を解釈する。これに対し、主成分得点を用いた回帰分析では、もとの説明変数ではなく、主成分得点がどう基準変数に影響するかに関心があり、主成分得点に対する回帰係数を解釈する。

具体的な手続きには、説明変数を主成分分析して主成分得点を求め、それを説明変数にして重回帰分析を行う。

回帰分析

lm, glm, semのいずれを用いてもよい

主成分分析

```
library(psych)
オブジェクト名 <- principal(データ名, nfactors=主成分数, rotate="none")
print(オブジェクト名)
スコア名 <- オブジェクト名$scores
```

あらかじめpsychパッケージをインストールしておく必要がある。
主成分数は変数数以下にする。

```
> setwd("i:¥¥Rdocuments¥¥scripts¥¥")
> d1 <- read.table("サポートデータ.csv", header=TRUE, sep=",")
> head(d1)
  support_s support_p kenkou
1         17         13     23
2         16         19     22
3         22         22     29
4         14         17     19
5         18         17     26
6         14         15     22
>
>
> # 記述統計量
> dtmp <- d1
> ntmp <- nrow(dtmp)
> mtmp <- colMeans(dtmp)
> stmp <- apply(dtmp, 2, sd)
> ctmp <- cor(dtmp)
> ktmp <- round(data.frame(ntmp, mtmp, stmp, ctmp), 2)
> colnames(ktmp) <- c("N", "Mean", "SD", colnames(ctmp))
> ktmp
      N Mean  SD support_s support_p kenkou
support_s 250 14.99 3.04      1.00      0.38      0.66
support_p 250 14.99 3.03      0.38      1.00      0.02
kenkou     250 20.00 5.01      0.66      0.02      1.00
>
>
> #標準偏回帰係数の推定
> d2 <- as.data.frame(scale(d1[,c("support_s", "support_p", "kenkou")]))
> #共分散行列の確認
> round(cov(d2), 2)
      support_s support_p kenkou
support_s      1.00      0.38      0.66
support_p      0.38      1.00      0.02
kenkou         0.66      0.02      1.00
> result.2 <- lm(kenkou ~ support_s + support_p, data=d2)
> summary(result.2)
```

	A	B	C
1	support_s	support_p	kenkou
2	17	13	23
3	16	19	22
4	22	22	29
5	14	17	19
6	18	17	26
7	14	15	22
8	15	19	17
9	14	13	20
10	14	14	20
11	17	12	27
12	18	19	22
13	18	13	24
14	9	14	6
15	15	16	23
16	19	18	24

Call:

```
lm(formula = kenkou ~ support_s + support_p, data = d2)
```

```

Residuals:
    Min       1Q   Median       3Q      Max
-1.87478 -0.43616  0.05371  0.50240  1.79567

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  8.444e-17  4.467e-02   0.000      1
support_s    7.675e-01  4.835e-02  15.872 < 2e-16 ***
support_p   -2.718e-01  4.835e-02  -5.621 5.11e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7063 on 247 degrees of freedom
Multiple R-squared:  0.5051,    Adjusted R-squared:  0.5011
F-statistic: 126.1 on 2 and 247 DF,  p-value: < 2.2e-16

> confint(result.2)
              2.5 %      97.5 %
(Intercept) -0.08798634  0.08798634
support_s    0.67226113  0.86274015
support_p   -0.36705158 -0.17657255
>
>
>
>
> # 説明変数の抽出
> d3 <- d2[,c("support_s", "support_p")]
>
>
> #主成分分析の実行
> # psychパッケージの読み込み
> library(psych)
> prin.1 <- principal(d3, nfactors=2, rotate="none", scores=TRUE)
> print(prin.1)

Principal Components Analysis
Call: principal(r = d3, nfactors = 2, rotate = "none", scores = TRUE)
Standardized loadings (pattern matrix) based upon correlation matrix
      PC1   PC2 h2    u2 com
support_s 0.83 -0.56 1 2.2e-16 1.7
support_p 0.83  0.56 1 2.2e-16 1.7

      PC1   PC2
SS loadings    1.38 0.62
Proportion Var    0.69 0.31
Cumulative Var    0.69 1.00
Proportion Explained 0.69 0.31
Cumulative Proportion 0.69 1.00

Mean item complexity = 1.7
Test of the hypothesis that 2 components are sufficient.

The root mean square of the residuals (RMSR) is 0
with the empirical chi square 0 with prob < NA

Fit based upon off diagonal values = 1

#> 主成分得点
> dPC <- prin.1$scores
> head(dPC)
      PC1      PC2
[1,] 0.002829912 -1.182324397
[2,] 0.996054277  0.887391725
[3,] 2.782506917  0.001807549
[4,] 0.201959837  0.886875872
[5,] 0.995707811 -0.294934736
[6,] -0.195260616  0.295454715
>

```

```
> # 主成分得点を説明変数にしたデータの作成
```

```
> d4 <- data.frame(dPC, d2[, "kenkou"])
```

```
> colnames(d4) <- c("pc1", "pc2", "kenkou")
```

```
>
```

```
>
```

```
> # 記述統計量
```

```
> dtmp <- d4
```

```
> ntmp <- nrow(dtmp)
```

```
> mtmp <- colMeans(dtmp)
```

```
> stmp <- apply(dtmp, 2, sd)
```

```
> ctmp <- cor(dtmp)
```

```
> ktmp <- round(data.frame(ntmp, mtmp, stmp, ctmp), 2)
```

```
> colnames(ktmp) <- c("N", "Mean", "SD", colnames(ctmp))
```

```
> ktmp
```

```
      N Mean SD  pc1  pc2 kenkou
```

```
pc1   250    0  1 1.00  0.00  0.41
```

```
pc2   250    0  1 0.00  1.00 -0.58
```

```
kenkou 250    0  1 0.41 -0.58  1.00
```

```
>
```

```
>
```

主成分得点間の相関は0になっている。

```
> # 主成分得点を用いた回帰分析
```

```
> res3 <- lm(kenkou ~ pc1+pc2, data=d4)
```

```
> summary(res3)
```

Call:

```
lm(formula = kenkou ~ pc1 + pc2, data = d4)
```

Residuals:

```
      Min       1Q   Median       3Q      Max
-1.87478 -0.43616  0.05371  0.50240  1.79567
```

Coefficients:

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.176e-18  4.467e-02   0.000      1
pc1           4.115e-01  4.476e-02   9.193 <2e-16 ***
pc2          -5.795e-01  4.476e-02 -12.946 <2e-16 ***
---

```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Residual standard error: 0.7063 on 247 degrees of freedom

Multiple R-squared: 0.5051, Adjusted R-squared: 0.5011

F-statistic: 126.1 on 2 and 247 DF, p-value: < 2.2e-16

```
>
```

```
>
```