

8. 回帰分析の応用

honocat

2025-12-31

```
HR09 <- read_csv('data/hr-data.csv') |>
  mutate(experience = as.numeric(status == '現職' | status == '元職')) |>
  filter(year == 2009) |>
  na.omit()
```

回帰分析で使うテクニック

線形変換

選挙費用を説明変数、得票率を結果変数とする回帰式を推定する。

選挙費用を 1 円単位で測定した `exp` を使った回帰式は、次のように求めることができる。

```
fit1 <- lm(voteshare ~ exp,
           data = HR09)
broom::tidy(fit1, conf.int = TRUE)
```

```
# A tibble: 2 x 7
  term          estimate std.error statistic  p.value  conf.low conf.high
<chr>          <dbl>     <dbl>     <dbl>    <dbl>    <dbl>    <dbl>
1 (Intercept)  7.71         0.758      10.2 2.98e- 23 6.22      9.19
2 exp          0.00000308 0.0000000961 32.0 3.64e-160 0.00000289 0.00000327
```

よって、

$$\widehat{\text{得票率}} = 7.71 + 0.00000308 \cdot \text{選挙費用(1 円)}$$

である。

これに対し、選挙費用を 100 万円単位で測定した `expm` を使うと、

```
fit2 <- lm(voteshare ~ I(exp / 10 ^ 6), data = HR09)
broom::tidy(fit2, conf.int = TRUE)
```

A tibble: 2 x 7

term	estimate	std.error	statistic	p.value	conf.low	conf.high
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1 (Intercept)	7.71	0.758	10.2	2.98e- 23	6.22	9.19
2 I(exp/10^6)	3.08	0.0961	32.0	3.64e-160	2.89	3.27

よって、

$$\widehat{\text{得票率}} = 7.71 + 3.08 \cdot \text{選挙費用(100 万円)}$$

である。どちらがわかりやすいか？

標準化

z 値で標準化した変数を使って回帰分析を行う。変数 x の z 値は、

$$z_x = \frac{x - \bar{x}}{u_x}$$

で求められる。ただし、 u_x は x の不偏分散の平方根である。

例として、選挙費用(測定単位：100 万円)を標準化し、得票率を説明してみよう。

```
HR09 <- HR09 |>
  mutate(z_expm = (expm - mean(expm, na.rm = TRUE)) / sd(expm, na.rm = TRUE))
summary(HR09$z_expm)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-1.2221	-0.8650	-0.2624	0.0000	0.6002	3.8514

これで、expm の z 値 z_expm が得られた。この変数を利用して回帰式を求める。

```
fit4 <- lm(voteshare ~ z_expm,
  data = HR09)
broom::tidy(fit4, conf.int = TRUE)
```

A tibble: 2 x 7

term	estimate	std.error	statistic	p.value	conf.low	conf.high
------	----------	-----------	-----------	---------	----------	-----------

	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	(Intercept)	26.5	0.480	55.3	5.93e-322	25.6	27.5
2	z_expm	15.4	0.480	32.0	3.64e-160	14.4	16.3

この結果は、選挙費用(100 万円)が 1 標準偏差増えるごとに、得票率が平均して 15.37 ポイント上昇することを示している。切片の 26.52 は、選挙費用が平均値を取ったときの得票率の予測値(平均値)である。

中心化

議員経験と選挙費用で得票率を説明するモデルを考える。回帰式を求めると、

```
fit5 <- lm(voteshare ~ experience * expm, data = HR09)
broom::tidy(fit5)
```

```
# A tibble: 4 x 5
  term          estimate std.error statistic    p.value
  <chr>          <dbl>    <dbl>    <dbl>    <dbl>
1 (Intercept)   -2.09      0.714    -2.93 3.48e- 3
2 experience     46.2      1.57     29.4 2.62e-141
3 expm           4.86     0.165     29.5 7.50e-142
4 experience:expm -4.76     0.206    -23.1 1.60e- 96
```

```
broom::glance(fit5)
```

```
# A tibble: 1 x 12
  r.squared adj.r.squared sigma statistic    p.value    df logLik   AIC   BIC
  <dbl>      <dbl> <dbl>    <dbl>    <dbl> <dbl> <dbl> <dbl> <dbl>
1   0.707      0.706  12.0     895. 2.72e-296     3 -4371. 8752. 8777.
# i 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

となる。このとき、係数の推定値は何を表しているだろうか。特に、相互作用を表す係数には注意が必要である。説明変数を中心化してから、同様の回帰式を求めてみる。

```
HR09 <- HR09 |>
  mutate(c_experience = experience - mean(experience),
         c_expm       = expm - mean(expm, na.rm = TRUE))
fit5_c <- lm(voteshare ~ c_experience * c_expm,
            data = HR09)
broom::tidy(fit5_c, conf.int = TRUE)
```

```
# A tibble: 4 x 7
```

	term	estimate	std.error	statistic	p.value	conf.low	conf.high
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	(Intercept)	34.5	0.501	69.0	0	33.6	35.5
2	c_experience	17.1	1.01	16.9	4.27e- 57	15.1	19.1
3	c_expm	2.93	0.110	26.6	5.99e-121	2.71	3.14
4	c_experience:c_expm	-4.76	0.206	-23.1	1.60e- 96	-5.16	-4.35

```
broom::glance(fit5_c)
```

```
# A tibble: 1 x 12
```

	r.squared	adj.r.squared	sigma	statistic	p.value	df	logLik	AIC	BIC
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	0.707	0.706	12.0	895.	2.72e-296	3	-4371.	8752.	8777.

```
# i 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

まず、残差の標準偏差(sigma の値、すなわち 12.0)と R^2 (1)が説明変数を中心化する前のモデルと全く同じことを確認してほしい。これは変数の中心化を行っても、回帰式の実質的な内容に変化がないことを示している。

次に、係数の意味を考えよう。切片である 34.54 が表しているのは、すべての説明変数が平均値を取ったときの得票率の予測である。中心化する前のモデルでは、すべての説明変数が 0 (これは非現実的でデータを代表しない値)のときの予測値が示されていたが、説明変数を中心化することによって、実質的に意味のある切片(データを代表するケースの予測値)を得ることができた。

c_experience の係数は、選挙費用が平均値のとき、議員経験がある候補者のほうが、議員経験がない候補者より 17.07 ポイント高い得票率を得ると期待されることを示す。中心化する前のモデルでは選挙費用が 0 の候補者(そのような候補者は存在しない)の傾きが示されていたのに対し、ここではデータ全体を代表する傾きが示されている。

c_expm の係数は、議員経験が平均値のとき、選挙費用を 1 単位(100 万円)増やすごとに得票率が平均 2.93 ポイント上昇することが期待されることを示す。中心化する前のモデルでは議員経験がない候補者の傾きが示されていたのに対し、ここではデータ全体を代表する(平均的な候補者の)傾きが示されている。

そして、c_experience と c_expm の交差項の係数は、議員経験がある候補者となない候補者の間には、選挙費用 1 単位が得票率に与える影響(傾き)の差が 4.76 であることを示す。この値は中心化する前のモデルと同じである。

対数変換

対数変換した変数を回帰分析で利用するときは、事前に変数を変換しても良いが、分析と同時に変換することもできる。例えば、次のようにする。

```
fit6 <- lm(log(voteshare) ~ expm,
           data = HR09)
broom::tidy(fit6, conf.int = TRUE)
```

```
# A tibble: 2 x 7
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1 (Intercept)	1.17	0.0522	22.4	2.09e- 92	1.07	1.27
2 expm	0.218	0.00661	32.9	1.61e-166	0.205	0.231

係数の値を元の測定単位に戻したいときは、`exp()` で計算すればよい。対数変換していない説明変数である `expm1` 単位の増加すなわち 100 万円の支出増は、得票率を

```
exp(coef(fit6)[2]) - 1
```

```
expm
0.2430331
```

だけ変化させる。つまり、得票率を 0.2430331 ポイント増加させる。

二次関数の推定

教育の収益率について考える。労働経済学においてミンサー方程式 (Mincer equation, Mincer earnings function) と呼ばれる、次のような式がある。

$$\log(\text{賃金}_i) = \beta_0 + \beta_1 \text{修学年数}_i + \beta_2 \text{就業経験}_i + \beta_3 \text{就業経験}_i^2 + \epsilon_i$$

ただし、「就業経験」は

$$\text{就業経験} = \text{年齢} - \text{就学年数} - 6$$

である。最後の項の 6 は、小学校に入学する年齢である。

このミンサー方程式を回帰分析で推定してみよう。ここで確かめたいのは、

1. 就学年数が賃金を上昇させるか(そして、どのくらい上昇させるか)
2. 就業経験が賃金を上昇させるか(そして、どのくらい上昇させるか)

であるが、就業経験の二乗項も説明変数に含まれている。これは、経験が浅いうちは、経験が賃金に与える正の効果が大きい、年齢が上がると経験を積んでも賃金が上がりにくくなる(場合によっては下がる)ことが想定されるからである。つまり、経験が賃金に及ぼす影響は逓減すると想定されている。

```
myd <- read_csv('data/fake_income.csv')
glimpse(myd)
```

Rows: 1,000

Columns: 3

\$ income <dbl> 33.9522, 124.6808, 140.4259, 79.6559, 145.1946, 1000.2209, ~

\$ education <dbl> 12, 16, 12, 14, 16, 16, 15, 14, 15, 16, 18, 12, 13, 15, 14, ~

\$ experience <dbl> 3, 18, 16, 14, 14, 11, 10, 16, 20, 18, 19, 6, 20, 19, 10, 5~

この回帰分析における結果変数は所得 income (万円)の自然対数、説明変数は就学年数 education、就業経験年数 experience と就業経験年数の二乗である。

```
fit_mincer <- lm(log(income) ~
                  education + experience + I(experience ^ 2),
                  data = myd)
broom::tidy(fit_mincer, conf.int = TRUE)
```

A tibble: 4 x 7

term	estimate	std.error	statistic	p.value	conf.low	conf.high
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1 (Intercept)	2.07	0.215	9.60	6.09e-21	1.65	2.49
2 education	0.138	0.0143	9.61	5.61e-21	0.109	0.166
3 experience	0.202	0.0162	12.5	2.92e-33	0.170	0.233
4 I(experience^2)	-0.00637	0.000683	-9.34	6.32e-20	-0.00771	-0.00503

結果変数が自然対数になっていて、就業年数(education)の係数の推定値が 0.14 なので、教育の収益率は 14% ほどであることがわかる。また、 t 値(statistic の列の値)が 2 よりも大きいので、5% の有意水準でこの効果は統計的に有意であることがわかる。

問：では、この効果は実質的に重要だろうか？

就学年数の影響とは異なり、就業経験の効果は推定値を見ただけではわかりにくい。就業経験の効果は、推定値を見ただけではわかりにくい。なぜなら、就業経験年数を動かすと、就業経験年数の二乗も一緒に動いてしまうからだ。そこで、修学年数を固定し、就業経験年数と就業経験年数の二乗を動かすと、結果変数である年収の対数がどのように動くか図示してみる。

まず、データ内の就学年数の平均値を求める。

```
(mean_educ <- mean(myd$education))
```

```
[1] 13.832
```

次に、就業経験年数の最小値と最大値を求め、その範囲の値を取る長さ 1,000 のベクトルを作る。

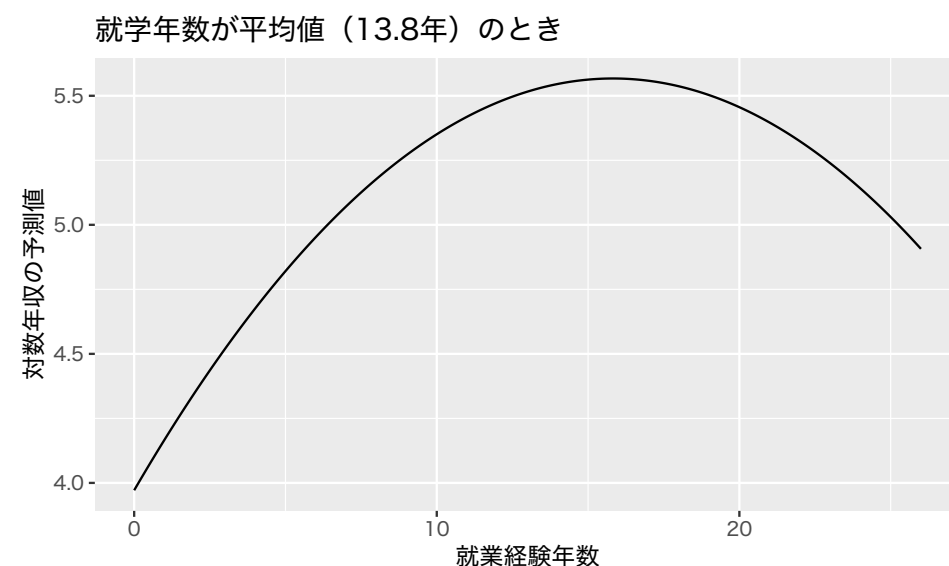
```
exper_vec <- with(myd,
  seq(from = min(experience),
    to   = max(experience),
    length.out = 1000))
```

就学年数を平均値に固定し、就業経験年数と就業経験年数の二乗を動かして予測値を求める。

```
pred_mean <- coef(fit_mincer)[1] +
  coef(fit_mincer)[2] * mean_educ +
  coef(fit_mincer)[3] * exper_vec +
  coef(fit_mincer)[4] * exper_vec ^ 2
```

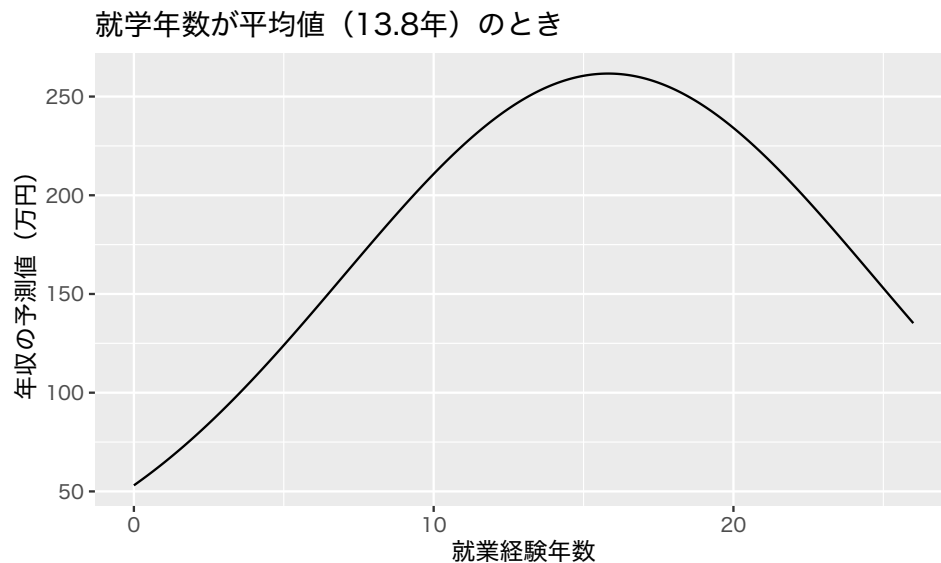
就業経験年数の変化に応じて年収の自然対数がどのように変化するか図示してみる。

```
dd <- tibble(exper      = exper_vec,
  pred_mean = pred_mean)
plt1 <- ggplot(dd, aes(x = exper, y = pred_mean)) +
  geom_line() +
  labs(x = '就業経験年数', y = '対数年収の予測値',
    title = '就学年数が平均値(13.8年)のとき')
plot(plt1)
```



これで、就業経験年数と対数年収の間にある非線形の関係が図示できた。対数年収はわかりにくいので、`exp()`でもとの単位に戻す。

```
plt2 <- ggplot(dd, aes(x = exper, y = exp(pred_mean))) +
  geom_line() +
  labs(x = ' 就業経験年数', y = ' 年収の予測値(万円) ',
       title = ' 就学年数が平均値(13.8 年)のとき')
plot(plt2)
```

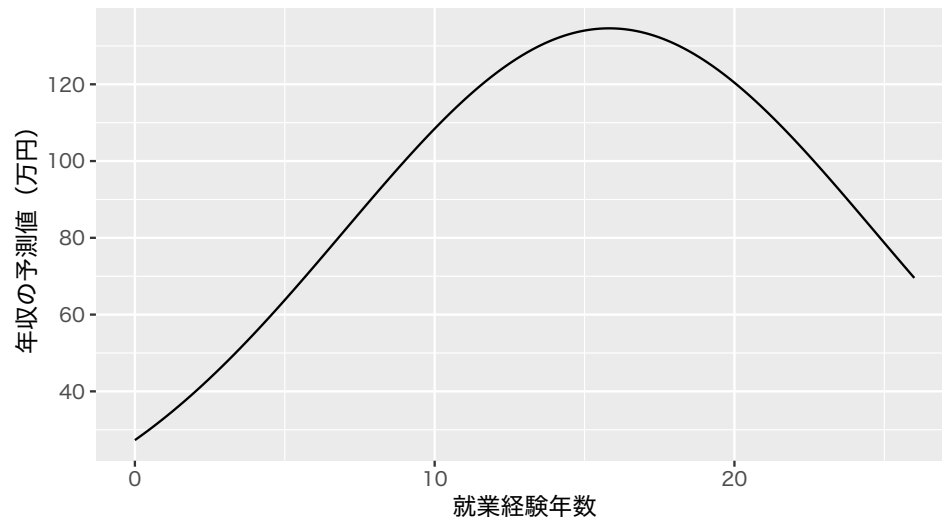


これで、就学年数が平均値のとき、就業経験年数と年収の間にある関係が図示できた。

修学年数が最小値のときは、

```
min_educ <- min(myd$education)
pred_min <- coef(fit_mincer)[1] +
  coef(fit_mincer)[2] * min_educ +
  coef(fit_mincer)[3] * exper_vec +
  coef(fit_mincer)[4] * exper_vec ^ 2
dd$pred_min <- pred_min
plt3 <- ggplot(dd, aes(x = exper, y = exp(pred_min))) +
  geom_line() +
  labs(x = ' 就業経験年数',
       y = ' 年収の予測値(万円) ',
       title = ' 就学年数が最小値(9 年)のとき')
plot(plt3)
```

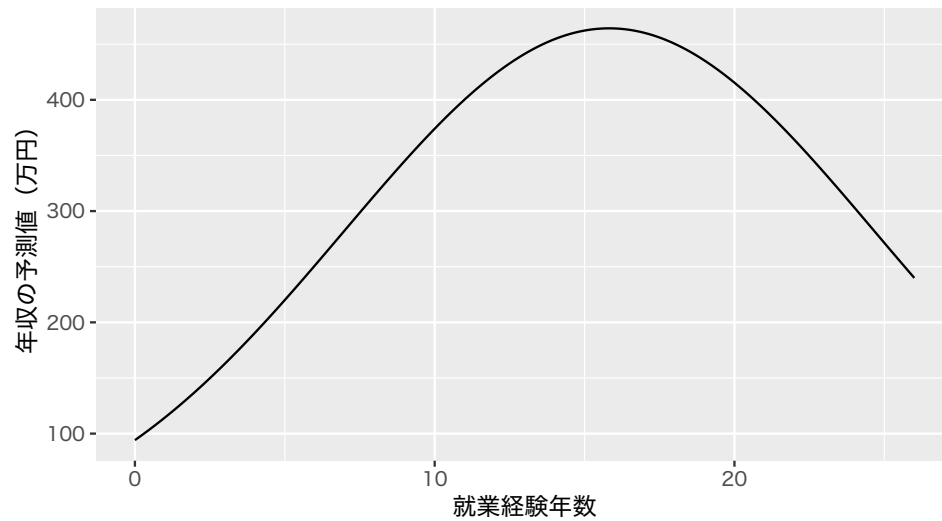
就学年数が最小値（9年）のとき



就学年数が最大値のときは、

```
max_educ <- max(myd$education)
pred_max <- coef(fit_mincer)[1] +
  coef(fit_mincer)[2] * max_educ +
  coef(fit_mincer)[3] * exper_vec +
  coef(fit_mincer)[4] * exper_vec ^ 2
dd$pred_max <- pred_max
plt4 <- ggplot(dd, aes(x = exper, y = exp(pred_max))) +
  geom_line() +
  labs(x = ' 就業経験年数',
       y = ' 年収の予測値(万円) ',
       title = ' 就学年数が最大値(18 年)のとき')
plot(plt4)
```

就学年数が最大値（18年） のとき



ひとつにまとめる。

```
dd_long <- dd |>
  pivot_longer(cols      = pred_mean : pred_max,
               names_to   = 'education',
               names_prefix = 'pred_',
               values_to   = 'predicted')
plt5 <- ggplot(dd_long,
               aes(x = exper, y = exp(predicted),
                   color = education)) +
  geom_line() +
  labs(x = ' 就業経験年数', y = ' 年収の予測値(万円) ') +
  scale_color_brewer(palette = 'Accent',
                    name     = ' 就学年数',
                    labels   = c(' 最大値(18年) ',
                                  ' 平均値(13.8年) ',
                                  ' 最小値(9年)'))
plot(plt5)
```

