Chapter 7: Problems with Predictors

Problems with Predictors

- Errors in predictors
- Change of scale
- Collinearity

Errors in Predictors

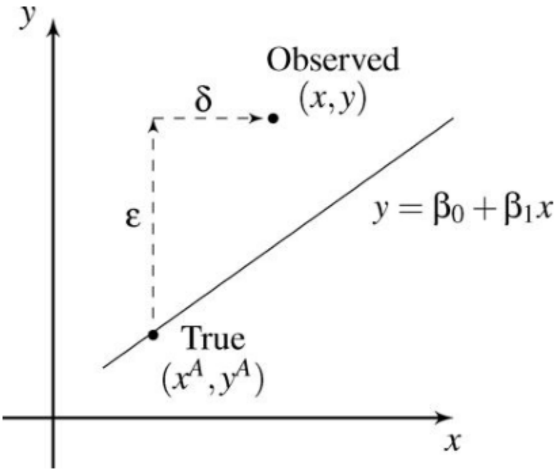

Figure: Measurement error: true vs. observed data

Errors in Predictors

Consider simple regression as example.

The X we observe is not the X that generates the y.

$$\begin{aligned} y_i^O &= y_i^A + \epsilon_i \\ x_i^O &= x_i^A + \delta_i \end{aligned}$$

The true relationship is:

$$y_i^A = \beta_0 + \beta_1 x_i^A$$

We get:

$$y_i^O = \beta_0 + \beta_1 x_i^O + (\epsilon_i - \beta_1 \delta_i)$$

Notations and Assumptions

Assume $E(\epsilon_i) = E(\delta_i) = 0$

Let

$$\begin{aligned} var(\epsilon_i) &= \sigma_\epsilon^2 \\ var(\delta_i) &= \sigma_\delta^2 \\ \sigma_x^2 &= \sum \left(x_i^A - \bar{x}^A\right)^2 / n \\ \sigma_{x\delta} &= cov(x^A, \delta) \end{aligned}$$

Effect on the fit

We use least squares to estimate β_1. It turns out

$$E(\hat{\beta}_1) = \beta_1 \times \left[\frac{\sigma_x^2 + \sigma_{x\delta}}{\sigma_x^2 + \sigma_\delta^2 + 2\sigma_{x\delta}} \right]$$

Scenario 1. x^A and δ are unrelated, i.e., $\sigma_{x\delta} = 0$. Then

$$E(\hat{\beta}_1) = \beta_1 \times \left[\frac{1}{1 + \sigma_\delta^2 / \sigma_x^2} \right] \leq \beta_1$$

- Shrinks toward 0
- If $\sigma_x^2 \gg \sigma_\delta^2$, the error can be ignored.

Simulation Example

Model: $\beta_1 = 1$

$$y^O = x^A + \epsilon$$

```
## No error in X
> xA <- 10*runif(50)
> yA <- xA
> yO <- yA + rnorm(50)
>  summary(lm(yO ~ xA))
Coefficients:
            Estimate Std.Error t value Pr(>|t|)
(Intercept) -0.23841   0.28125  -0.848    0.401
xA           1.06733   0.05414  19.715   <2e-16
```

Model:

$$y^O = x^O + \delta + \epsilon$$

```
> xO <- xA + rnorm(50)
> summary(lm(yO ~ xO))
Coefficients:
            Estimate Std.Error t value Pr(>|t|)
(Intercept)  0.56790   0.33005   1.721   0.0918
xO           0.89873   0.06198  14.501   <2e-16

## Larger errors
> xO_2 <- xA + 5*rnorm(50)
> summary(lm(yO ~ xO_2))
Coefficients:
            Estimate Std.Error t value Pr(>|t|)
(Intercept)  4.34652   0.49175   8.839 1.23e-11
xO_2         0.07710   0.07035   1.096    0.279
```

Effect on the fit

Scenario 2. Consider two possibilities:

- x^A is fixed, but measured as x^O. If measurement is repeated, x^A is the same, but x^O will change.
- x^O is fixed, while x^A changes at every repetition. In this case,

$$\sigma_{x\delta} = cov(X^O - \delta, \delta) = -\sigma_\delta^2$$

Hence $E(\hat{\beta}_1) = \beta_1$.

```
## Observed X are fixed
> xO <- seq(0, 10, length=50)
> xA <- xO + 5*rnorm(50)
> yO <- xA + rnorm(50)
> summary(lm(yO ~ xO))
Coefficients:
            Estimate Std.Error t value Pr(>|t|)
(Intercept)   1.0942    1.3962   0.784 0.437060
xO            0.8581    0.2406   3.567 0.000832
```

Change of Scale

$$x_j \to \frac{x_j + a}{b}$$

- Predictors of similar magnitude are easier to compare.
- Numerical stability
- Easy interpretation

Consequences

- Rescaling x_j leaves the t and F tests and $\hat{\sigma}^2$ and R^2 unchanged.

$$\hat{\beta}_j \to b\hat{\beta}_j$$

- Rescaling y leaves the t and F tests and R^2 unchanged but both $\hat{\sigma}$ and $\hat{\beta}$ rescaled by b; $\hat{\beta}_0$ is both shifted by a and rescaled by b.

Savings Example

```
> data(savings)
> result <- lm(sr ~ ., data=savings)
> summary(result)
Coefficients:
            Estimate Std.Error t value Pr(>|t|)
Intercept 28.5666100 7.3544986   3.884 0.000334
pop15     -0.4612050 0.1446425  -3.189 0.002602
pop75     -1.6915757 1.0835862  -1.561 0.125508
dpi       -0.0003368 0.0009311  -0.362 0.719296
ddpi       0.4096998 0.1961961   2.088 0.042468
Residual standard error: 3.803 on 45 degrees of freedom
Multiple R-Squared: 0.3385  Adjusted R-squared: 0.2797
F-statistic: 5.756 on 4 and 45 DF  p-value: 0.0007902
```

Savings Example

```
## Scale one predictor variable
> summary(lm(sr ~ pop15 + pop75 + I(dpi/1000)
  + ddpi, data=savings))
Coefficients:
            Estimate Std.Error t value Pr(>|t|)
(Intercept)  28.5666    7.3545   3.884 0.000334
pop15        -0.4612    0.1446  -3.189 0.002602
pop75        -1.6916    1.0836  -1.561 0.125508
I(dpi/1000) -0.3368    0.9311  -0.362 0.719296
ddpi          0.4097    0.1962   2.088 0.042468
Residual standard error: 3.803 on 45 degrees of freedom
Multiple R-Squared: 0.3385   Adjusted R-squared: 0.2797
F-statistic: 5.756 on 4 and 45 DF    p-value: 0.0007902
```

Standardizing variables

- Convert all variables to standard units (mean 0, variance 1)
- Can compare coefficients directly
- Helps numerical stability
- Interpretation may be easier or harder

```
## Standardize all variables
> sctemp <- data.frame(scale(savings))
> summary(lm(sr ~ ., data=sctemp))
Coefficients:
           Estimate Std.Error   t value Pr(>|t|)
Intercept-2.453e-16 1.200e-01 -2.04e-15   1.0000
pop15    -9.420e-01 2.954e-01    -3.189   0.0026
pop75    -4.873e-01 3.122e-01    -1.561   0.1255
dpi      -7.448e-02 2.059e-01    -0.362   0.7193
ddpi      2.624e-01 1.257e-01     2.088   0.0425
Residual standard error:   0.8487 on 45 degrees of freedom
Multiple R-Squared: 0.3385      Adjusted R-squared: 0.2797
F-statistic: 5.756 on 4 and 45 DF       p-value: 0.0007902
```

Collinearity

- Collinearity: X^TX close to singular
- Cause: some predictors are (almost) linear combinations of others.
- Detection:
  - Correlation matrix: large pairwise correlation
  - Regress x_j on other predictors – get R_j^2.
    R_j^2 close to 1 indicates a problem
  - Condition number of X^TX: $\kappa = \sqrt{\frac{\lambda_1}{\lambda_{p+1}}}$
    where λ_1 is the largest eigenvalue and λ_{p+1} is the minimum eigenvalue of X^TX.
  - Rule of Thumb: $\kappa > 30$ are signs of collinearity.

Consequences of Collinearity

- Imprecise estimate of β
- Inflated standard error
- t-test fails to reveal significant predictors
- Sensitivity to measurement errors
- Numerical instability

Collinearity Continued

Why? Let $S_{x_j} = \sum_i (x_{ij} - \bar{x}_j)^2$, then

$$var(\hat{\beta}_j) = \sigma^2 \left(\frac{1}{1 - R_j^2} \right) \frac{1}{S_{x_j}}$$

- Variance inflation factor (VIF): $\frac{1}{1-R_j^2}$
- Spread of x_j

- Rule of Thumb: VIF $>$ 10 are signs of collinearity.

Car Example

- Car drivers adjust the seat position for comfort
- Response: seat position
- Predictors: age, weight, height with and without shoes, seated height, arm length, thigh length, lower leg length

```
> data(seatpos)
> result <- lm(hipcenter ~ ., data=seatpos)
> summary(result)
```

```
Coefficients:
            Estimate Std.Error t value Pr(>|t|)
(Intercept)436.43213 166.57162   2.620   0.0138
Age          0.77572   0.57033   1.360   0.1843
Weight       0.02631   0.33097   0.080   0.9372
HtShoes     -2.69241   9.75304  -0.276   0.7845
Ht           0.60134  10.12987   0.059   0.9531
Seated       0.53375   3.76189   0.142   0.8882
Arm         -1.32807   3.90020  -0.341   0.7359
Thigh       -1.14312   2.66002  -0.430   0.6706
Leg         -6.43905   4.71386  -1.366   0.1824
Residual standard error:  37.72 on 29 degrees of freedom
Multiple R-Squared: 0.6866    Adjusted R-squared: 0.6001
F-statistic:  7.94 on 8 and 29 DF     p-value: 1.306e-05
```

Collinearity: 1

```
## Correlation matrix
> round(cor(seatpos)[2:7, 2:7], 2)
        Weight HtShoes   Ht Seated  Arm Thigh
Weight    1.00    0.83 0.83   0.78 0.70  0.57
HtShoes   0.83    1.00 1.00   0.93 0.75  0.72
Ht        0.83    1.00 1.00   0.93 0.75  0.73
Seated    0.78    0.93 0.93   1.00 0.63  0.61
Arm       0.70    0.75 0.75   0.63 1.00  0.67
Thigh     0.57    0.72 0.73   0.61 0.67  1.00
```

Collinearity: 3

```
## Condition number
> X <- model.matrix(result)[, -1]
> e <- eigen(t(X) %*% X)
> e$val
[1] 3.653671e+06 2.147948e+04 9.043225e+03
[4] 2.989526e+02 1.483948e+02 8.117397e+01
[7] 5.336194e+01 7.298209e+00
> round(sqrt(e$val[1]/e$val), 3)
[1]   1.000  13.042  20.100 110.551 156.912
[6] 212.156 261.667 707.549
```

Collinearity: 2

```
## Variance inflation factor
> library(faraway)
> round(vif(X), 3)
    Age  Weight HtShoes      Ht  Seated
  1.998   3.647 307.429 333.138   8.951
    Arm   Thigh     Leg
  4.496   2.763   6.694
```

Consequence of Collinearity

```
## Sensitivity to measurement errors
> junk <- lm(hipcenter + 10*rnorm(38) ~ ., data=seatpos)
> summary(junk)
Coefficients:
             Estimate Std.Error t value Pr(>|t|)
(Intercept)431.13413 176.13709   2.448   0.0207
Age           0.60041   0.60308   0.996   0.3277
Weight       -0.10886   0.34998  -0.311   0.7580
HtShoes      -3.86967  10.31311  -0.375   0.7102
Ht            1.33472  10.71159   0.125   0.9017
Seated        0.79736   3.97792   0.200   0.8425
Arm          -0.01702   4.12417  -0.004   0.9967
Thigh        -1.54993   2.81278  -0.551   0.5858
Leg          -4.73289   4.98456  -0.950   0.3502
```

```
## Correlation of variables measuring length
> round(cor(X[, 3:8]), 2)
        HtShoes   Ht Seated  Arm Thigh  Leg
HtShoes    1.00 1.00   0.93 0.75  0.72 0.91
Ht         1.00 1.00   0.93 0.75  0.73 0.91
Seated     0.93 0.93   1.00 0.63  0.61 0.81
Arm        0.75 0.75   0.63 1.00  0.67 0.75
Thigh      0.72 0.73   0.61 0.67  1.00 0.65
Leg        0.91 0.91   0.81 0.75  0.65 1.00
```

```
## Using a subset of predictor variables
> result2 <- lm(hipcenter ~ Age + Weight + Ht,
    data=seatpos)
> summary(result2)
Coefficients:
            Estimate Std.Error t value Pr(>|t|)
Intercept 528.297729 135.31295   3.904 0.000426
Age         0.519504  0.408039   1.273 0.211593
Weight      0.004271  0.311720   0.014 0.989149
Ht         -4.211905  0.999056  -4.216 0.000174

Residual standard error: 36.49 on 34 degrees of freedom
Multiple R-Squared: 0.6562   Adjusted R-squared: 0.6258
F-statistic: 21.63 on 3 and 34 DF    p-value: 5.125e-08
```

What to do about collinearity

- If you mostly care about prediction, drop highly correlated predictors
- Variable selection may be used (Ch 8)
- If interpretation is important and you must keep all predictors, do not use least squares. Use some other estimation method, e.g., ridge regression (Ch 9)