Problems with the Predictors

Chapter 7 of LMWR2

Subrata Paul

6/3/2020

Errors in the Predictors

Our standard regression model allows for errors in the response by including the ϵ term, but what if the predictors are measured with error?

- i.e., What if the observed X is not the one used to generate Y?
- e.g., what if the predictor were the amount of exposure to secondhand tobacco smoke? This would be very difficult to measure.

We (generally) don't want to treat X as a random variable.

- This is possible for observational data, but not experimental.
- Regression inference proceeds on a fixed value of X (though we may not be able to measure X accurately).

Account for errors in predictors

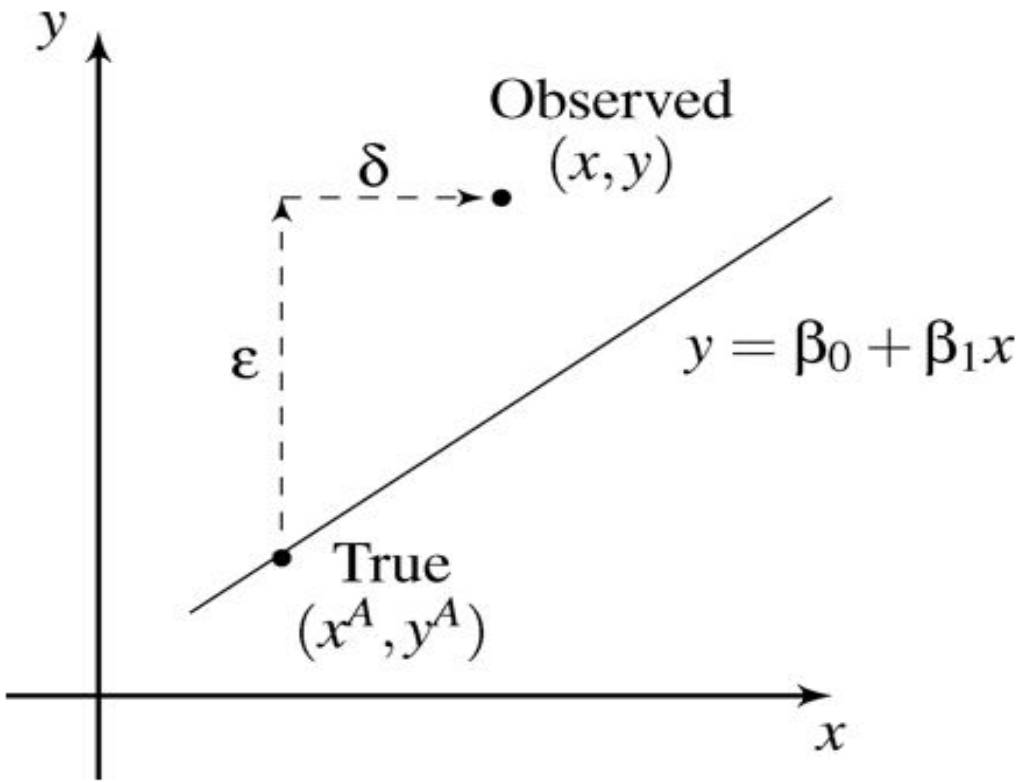
Suppose that we observe (x_i^o, y_i^o) for $i = 1, 2, \ldots, n$, which are related to the true values (x_i^a, y_i^a):

$$y_i^o = y_i^a + \epsilon_i$$

$$x_i^o = x_i^a + \delta_i$$

where the errors ϵ and δ are independent.

In graphics



Problems

The true underlying relationship is

$$y_i^a = \beta_0 + \beta_1 x_i^a,$$

but we only see (x_i^o, y_i^o). They are related though the equation:

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

Assume that $E(\epsilon) = E(\delta) = 0$ and $var(\epsilon) = \sigma_\epsilon^2 I and var(\delta) = \sigma_\delta^2 I$. Define

$$\sigma_x^2 = \frac{\sum (x_i^a - \overline{x^a})^2}{n}, \quad \sigma_{x\delta} = cov(x^a, \delta).$$

Problems

For observational data, σ_x^2 is (essentially) the sample variance of X^a, while for a controlled experiment, it is just a numerical measure of the spread of the design.

We can often assume $cov(x^a, \delta) = 0$.

The least squares estimator of β_1 is

$$\hat{\beta}_1 = \frac{\sum(x_i - \bar{x})y_i}{\sum(x_i - \bar{x})^2},$$

and we can derive that

$$E(\hat{\beta}_1) = \beta_1 \frac{\sigma_x^2 + \sigma_{x\delta}}{\sigma_x^2 + \sigma_\delta^2 + 2\sigma_{x\delta}}.$$

Special case I

If there is no relation between X^a and δ, then $\sigma_{x\delta} = 0$, and the expected value simplifies to

$$E(\hat{\beta}_1) = \beta_1 \frac{1}{1 + \sigma_\delta^2/\sigma_x^2}.$$

- In this case, the estimate will be biased toward zero, though this won't be a problem as long as $\sigma_\delta^2 \ll \sigma_x^2$.
- We typically see the same pattern for multiple predictors.

Special case II

In controlled experiments, there are two ways in which errors may arise.

- In the first case, we measure x, but instead of observing x^a, we observe x^o. If we measure x again, we will get a different x^o.

- In the second case, we fix x^o (e.g., we make a chemical solution with concentration x^o). However, the true concentration would be x^a. If we repeated this process, we would get the same x^o but a different x^a (since we are trying to make the solution at the same concentration).

    - In this case, $\sigma_{x\delta} = cov(X^o - \delta, \delta) = -\sigma_\delta^2$, and we would have that $E(\hat{\beta}_1) = \beta_1$.

    - This case essentially reverses the role of x^a and x^o, and if you get to observe the true X, then you will get an unbiased estimate of β_1.

Solution

When the error in X cannot be ignored, we should consider alternatives to the least squares estimation of β.

- Two possibilities are to consider the geometric mean functional relationship or the SIMEX method.
- Read the book for more details.

Change of Scale

Suppose we want to change the scale of the variables, e.g., $x_i < -(x_i + a)/b$.

- This may result in the estimated regression coefficient having a better scale (e.g., $\hat{\beta}_1 = 3.51 \; vs \; \hat{\beta}_1 = 0.00000351$).
- This may enhance numerical stability (really large or small values can cause problems).

Rescaling x_i leaves the t- and F-tests unchanged, as well as $\hat{\sigma}^2$ and R^2 unchanged. $\hat{\beta}_i \rightarrow b\hat{\beta}_i$.

Rescaling y leaves the t- and F-tests unchanged, and R^2 unchanged. $\hat{\sigma}^2$ and $\hat{\beta}$ will be multiplied by b.

Savings Example

The `savings` data has data related to 50 savings-related variables in 50 countries, averaged over the period 1960-1970. The data has the following variables:

- `sr` - savings rate. Personal saving divided by disposable income
- `pop15` - percent population under age of 15
- `pop75` - percent population over age of 75
- `dpi` - per-capita disposable income in dollars
- `ddpi` - percent growth rate of dpi

Consider the changes in the regression models when we rescale the `dpi` predictor by 1000. What changes and what doesn't? What if we rescaled the response (multiplying by 1000)?

Model fit

```
lm1 <- lm(sr ~ pop15 + pop75 + dpi + ddpi, data = savings)
sumary(lm1)
```

```
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept) 28.56608654  7.35451611  3.8842 0.0003338
## pop15       -0.46119315  0.14464222 -3.1885 0.0026030
## pop75       -1.69149768  1.08359893 -1.5610 0.1255298
## dpi         -0.00033690  0.00093111 -0.3618 0.7191732
## ddpi         0.40969493  0.19619713  2.0882 0.0424711
##
## n = 50, p = 5, Residual SE = 3.80267, R-Squared = 0.34
```

Model fit after scaling dpi by 1000:

```
lm2 <- lm(sr ~ pop15 + pop75 + I(dpi/1000) + ddpi, data = savings)
sumary(lm2)
```

```
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 28.56609    7.35452  3.8842 0.0003338
## pop15       -0.46119    0.14464 -3.1885 0.0026030
## pop75       -1.69150    1.08360 -1.5610 0.1255298
## I(dpi/1000) -0.33690    0.93111 -0.3618 0.7191732
## ddpi         0.40969    0.19620  2.0882 0.0424711
##
## n = 50, p = 5, Residual SE = 3.80267, R-Squared = 0.34
```

Model after scaling response by 1000:

```
lm3 <- lm(I(sr*1000) ~ pop15 + pop75 + dpi + ddpi, data = savings)
sumary(lm3)
```

```
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept) 28566.08654  7354.51611  3.8842 0.0003338
## pop15        -461.19315   144.64222 -3.1885 0.0026030
## pop75       -1691.49768  1083.59893 -1.5610 0.1255298
## dpi            -0.33690     0.93111 -0.3618 0.7191732
## ddpi          409.69493   196.19713  2.0882 0.0424711
##
## n = 50, p = 5, Residual SE = 3802.66865, R-Squared = 0.34
```

Scaling

A very thorough approach to scaling is to convert all variables to standard units (mean 0 and variance 1) using the `scale` function.

- The fitted line will have an intercept of 0.
- Advantages:
  - All the predictors are on a comparable scale, making comparisons simpler.
  - The coefficients can be viewed as a kind of partial correlation (the values are always between -1 and 1).
  - We avoid numerical problems that arise when predictors are on very different scales.
- Disadvantages:
  - The regression coefficients represent the effect of a one standard deviation increase in the predictor on the response in standard deviations.
  - This is not usually easy to interpret.

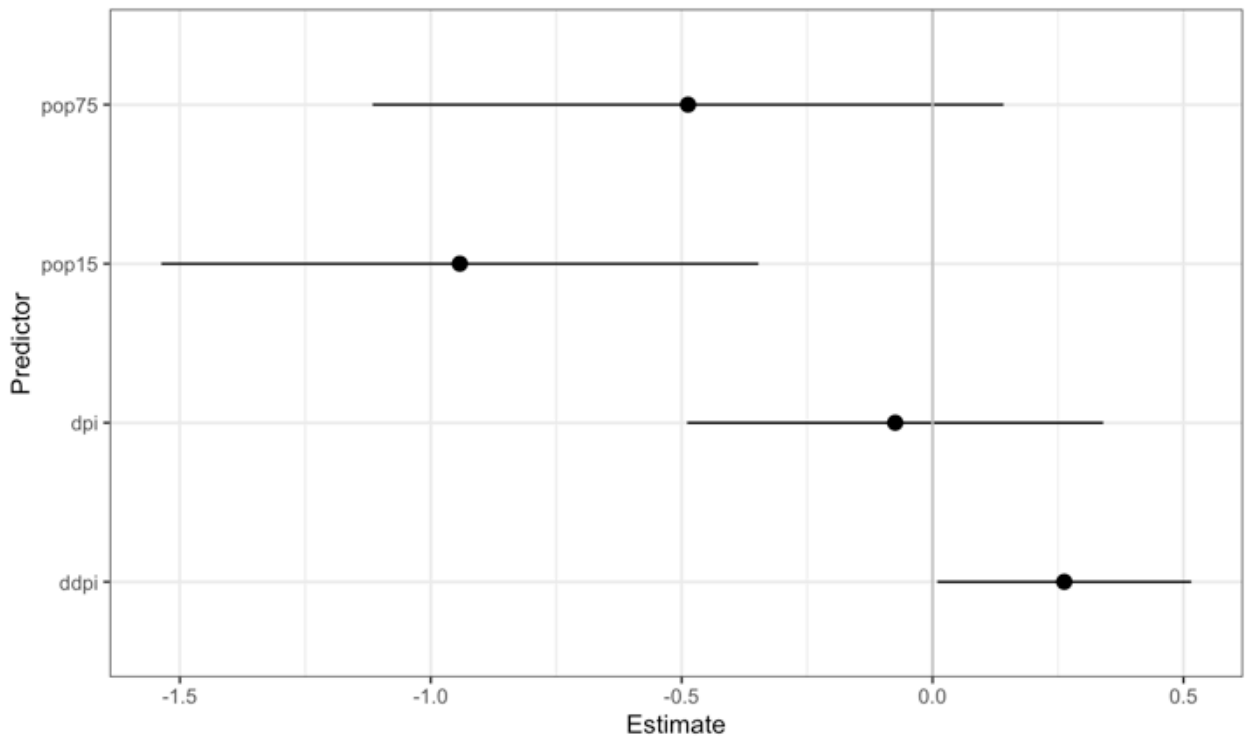
Savings example

```
scsavings <- data.frame(scale(savings))
lm4 <- lm(sr ~ pop15 + pop75 + dpi + ddpi, data = scsavings)
sumary(lm4)
```

```
##                Estimate  Std. Error t value Pr(>|t|)
## (Intercept)  4.0116e-16  1.2003e-01  0.0000 1.000000
## pop15       -9.4204e-01  2.9545e-01 -3.1885 0.002603
## pop75       -4.8731e-01  3.1218e-01 -1.5610 0.125530
## dpi         -7.4508e-02  2.0592e-01 -0.3618 0.719173
## ddpi         2.6243e-01  1.2567e-01  2.0882 0.042471
##
## n = 50, p = 5, Residual SE = 0.84873, R-Squared = 0.34
```

Plot of estimates

```
edf <- data.frame(coef(lm4),confint(lm4))[-1,]
names(edf) <- c('Estimate','lb','ub')
library(ggplot2)
p <- ggplot(aes(y=Estimate,ymin=lb,ymax=ub,x=row.names(edf)),data=edf) + geom_pointrange()
p + coord_flip() + xlab("Predictor") + geom_hline(yintercept=0, col=gray(0.75)) + theme_bw()
```



Scaling Binary Variables

Scaling might be done differently when there are binary regressors.

- A binary variable that takes the values 0/1 with probability half will have a standard deviation of 0.5.
- Moving the binary variable from 0 to 1 means moving 2 SDs.

This suggests scaling the other continuous regressors by 2 SDs rather than 1 so that interpretations are on a common scale (1 unit increase = 2 SD increase)

Savings example

Recall that the data clusters based on the pop15 predictor.

- We divide pop15 at 35%, so that the younger countries are coded as zero and the older countries as one.

```
savings$age <- ifelse(savings$pop15 > 35, 0, 1)
savings$dpis <- (savings$dpi-mean(savings$dpi))/(2*sd(savings$dpi))
savings$ddpis <- (savings$ddpi - mean(savings$ddpi))/(2*sd(savings$ddpi))
sumary(lm(sr ~ age + dpis + ddpis, savings))
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.8176     1.0106  6.7464 2.19e-08
## age           5.2841     1.5849  3.3341 0.001697
## dpis         -1.5642     1.6093 -0.9720 0.336127
## ddpis         2.4681     1.1082  2.2272 0.030866
##
## n = 50, p = 4, Residual SE = 3.79990, R-Squared = 0.32
```

Interpretation

The predicted savings rate is about 5.3% higher for countries with a younger population.

The same change of two standard deviations in ddpi means a difference of one in the new scale of ddpis.

Recall: ddpi is the percent growth rate of dpi. A typical country with a growth rate of dpi two standard deviation more than another typical country has a savings rate 2.47% higher.

Another way to achieve a similar effect is to use a −1/+1 coding rather than 0/1 so that the standard scaling can be used on the continuous predictors.

Collinearity

When the columns of X are linearly dependent, then $X^T X$ is singular and there is no unique least squares estimate of β.

- The columns of X are said to be exactly collinear in this case.
- This causes serious problems with estimation and interpretation.

Even when the columns of X are not perfectly dependent, we still have problems.

What it does

Collinearity leads to imprecise estimates of β.

- The signs of the coefficients can be the opposite of what intuition about the effect of the predictors might suggest.
- The standard errors become inflated so it may be difficult to detect significant regression coefficients.
- The fit becomes very sensitive to measurement errors.
    - Small changes in y can lead to large changes in $\hat{\beta}$.

Detect Collinearity

Pairwise correlation

Examine the pairwise correlation matrix of the regressors and look for large pairwise correlations.

- Large is a bit subjective, but the larger the correlation among regressors, the more likely it is that you have a collinearity problem.

Coefficient of determination among regressors

Let R_j^2 denote the coefficient of determination when regressing x_j on all other regressors.

- Repeat for all regressors.

R_j^2 close to one indicates a collinearity problem.

The offending linear combination can be discovered by examining the regression coefficients from each of these fits (which ones are significant?).

Difficulties

Collinearity makes some of the parameters difficult to estimate precisely.

Define S_j^2 to be the sample variance of regressor j. We can show that

$$var(\hat{\beta}_j) = \sigma^2 \frac{1}{1 - R_j^2} \frac{1}{(n-1)S_j^2}.$$

Two facts

- If x_j does not vary much, then the variance of $\hat{\beta}_j$ will be large (since S_j^2 will be small).

```
x1<-runif(100,0.4,0.5)
x2<-runif(100,0,1)
y1<-2*x1 + rnorm(100, 0, 0.5)
y2<-2*x2 + rnorm(100, 0, 0.5)
summary(lm(y1~x1))$coefficients
```

```
##               Estimate Std. Error    t value  Pr(>|t|)
## (Intercept) 0.05701166  0.8064542 0.07069423 0.9437852
## x1          1.73470475  1.7828313 0.97300555 0.3329461
```

```
summary(lm(y2~x2))$coefficients
```

```
##                Estimate Std. Error     t value     Pr(>|t|)
## (Intercept) -0.002995889  0.0968211 -0.03094252 9.753783e-01
## x2           1.977811897  0.1700102 11.63348713 3.624086e-20
```

Two facts

- We can maximize S_j^2 by spreading X as much as possible.
  - Placing half of the points at the minimum practical value and half at the maximum maximizes this.

  - This design assumes linearity and makes it impossible to check for curvature.
  - Generally, we distribute values a bit more than this.

- We can use this fact to choose experimental designs that minimize the variance of the estimated regression coefficients.
  - Orthogonality implies that $R_j^2 = 0$, which minimizes the variance.

Variance Inflation Factor

If R_j^2 is close to 1, then the variance inflation factor

$$VIF_j = \frac{1}{1 - R_j^2}$$

will be large.

VIF_j more than 5 or 10 indicates a potential problem with collinearity for regressor x_j.

VIF

The VIF is the standard diagnostic for assessing collinearity.

The VIF is not appropriate for assessing collinearity for sets of related regressors like dummy-variable regressors or polynomial regressors.

The generalized VIF should be used in those cases.

For the model $Y = \beta_0 + X_c\beta_c + X_r\beta_r + \epsilon$,

$$GVIF_c = \frac{\det(R_c)\det(R_r)}{\det(R)}$$

where R_c, R_r, and R represent correlation matrix for X_c, X_r and X.

- The vif function in the car package automatically computes the generalized VIF for related regressors.

Condition number

Examine the eigenvalues of $X^T X$ (usually after scaling the predictors so they have a standard deviation of 1). Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$ be the eigenvalues of the p regressors ordered from largest to smallest.

- When the condition number $\kappa = \sqrt{\lambda_1 / \lambda_p} \geq 30$ then there is a potential problem with collinearity.
- The other condition indices are $ are worth examining, because they may indicate a problem with more than one linear combination of the regressors.

Variance Decomposition Proportions

Variance decomposition proportions can be examined to determine the regressors that are leading to large condition indices.

- This information is provided by the colldiag function in the perturb package.
- A variable is involved in the linear dependency if the sum of its proportions over the rows with large condition indices is more than 0.5.

Belsley (1991)

Belsley (1991) recommends that when using condition indices to assess collinearity that:

- The intercept be included in your X matrix
- The columns of X should NOT be centered.
- The columns of X should be scaled (i.e., the standard deviation of each column should be constant).

Belsley, D.A. Computer Science in Economics and Management (1991) 4: 33. https://doi.org/10.1007/BF00426854

Driving Example:

Car drivers like to adjust the seat position for their own comfort. Car designers would find it helpful to know where different drivers will position the seat depending on their size and age. Researchers at the HuMoSim laboratory at the University of Michigan collected data on 38 drivers. They measured age in years, weight in pounds, height with shoes and without shoes in cm, seated height arm length, thigh length, lower leg length and `hipcenter` the horizontal distance of the midpoint of the hips from a fixed location in the car in mm.

Fit a model with all predictors

```
lm1 <- lm(hipcenter ~ ., data = seatpos)
sumary(lm1)
```

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 436.432128 166.571619  2.6201  0.01384
## Age           0.775716   0.570329  1.3601  0.18427
## Weight        0.026313   0.330970  0.0795  0.93718
## HtShoes      -2.692408   9.753035 -0.2761  0.78446
## Ht            0.601345  10.129874  0.0594  0.95307
## Seated        0.533752   3.761894  0.1419  0.88815
## Arm          -1.328069   3.900197 -0.3405  0.73592
## Thigh        -1.143119   2.660024 -0.4297  0.67056
## Leg          -6.439046   4.713860 -1.3660  0.18245
##
## n = 38, p = 9, Residual SE = 37.72029, R-Squared = 0.69
```

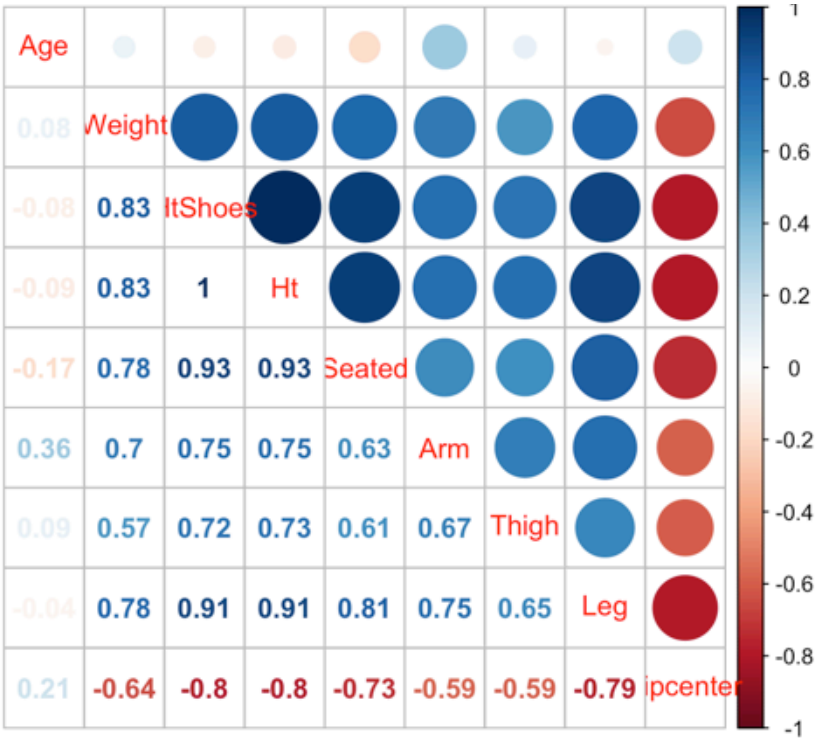
What we get?

Notice that the R^2 value is large (the model seems to fit the data fairly closely) but none of the individual predictors are significant!

This is a sign of a problem with collinearity.

Pairwize correlations

```
corrplot::corrplot.mixed(cor(seatpos))
```



There are several large pairwise correlations between predictors and between predictors and the response.

VIF

```
vif(lm1)
```

```
##        Age     Weight    HtShoes         Ht     Seated        Arm      Thigh
##   1.997931   3.647030 307.429378 333.137832   8.951054   4.496368   2.762886
##        Leg
##   6.694291
```

There is a lot of variance inflation.

- We can interpret $\sqrt{307.4}=17.5$ as meaning that the standard error for height with shoes is 17.5 times larger than it would have been without collinearity.
    - This interpretation is not completely perfect since this is observational data and we cannot make orthogonal predictors.

Condition indices

intercept	Age	Weight	HtShoes	Ht	Seated	Arm	Thigh	Leg	cond.index
0	0.001	0	0	0	0	0	0	0	1
0	0.485	0.002	0	0	0	0	0	0	7.833
0.007	0.002	0.349	0	0	0	0	0.001	0	17.196
0.051	0.022	0.084	0	0	0.005	0.044	0.464	0	43.518
0.03	0.045	0.188	0	0	0.001	0.402	0.287	0.071	55.578
0.092	0.259	0.12	0	0	0.001	0.515	0.002	0.514	79.522
0.804	0.11	0.244	0.002	0.002	0.13	0.004	0.045	0.227	116.37
0.016	0.001	0.003	0.016	0.015	0.862	0.027	0.12	0.185	213.599
0	0.075	0.011	0.981	0.983	0.001	0.008	0.08	0.002	1153.483

Several condition indices are large.

- There are problems with more than one linear combination of predictors.

Add more noice

If we add a little bit of measurement error to the response, we get a large change in the estimated regression coefficients.

```
lm2 <- lm(hipcenter + 10 * rnorm(38) ~ ., data = seatpos)
sumary(lm2)
```

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 444.353146 163.103050  2.7244   0.0108
## Age           0.680906   0.558453  1.2193   0.2326
## Weight        0.085923   0.324078  0.2651   0.7928
## HtShoes      -1.876784   9.549945 -0.1965   0.8456
## Ht           -2.330553   9.918937 -0.2350   0.8159
## Seated        3.150185   3.683559  0.8552   0.3995
## Arm           0.509640   3.818982  0.1334   0.8948
## Thigh        -0.430835   2.604633 -0.1654   0.8698
## Leg          -5.797951   4.615702 -1.2561   0.2191
##
## n = 38, p = 9, Residual SE = 36.93483, R-Squared = 0.71
```

Compare

	Model1	Model2
(Intercept)	436.432	444.353
Age	0.776	0.681
Weight	0.026	0.086
HtShoes	-2.692	-1.877
Ht	0.601	-2.331
Seated	0.534	3.150
Arm	-1.328	0.510
Thigh	-1.143	-0.431
Leg	-6.439	-5.798

The R^2 and standard error are very similar to the previous fit, but the coefficients have changed dramatically!

- The coefficients are quite sensitive to collinearity.

Solution

- Amputating regressors collinear with other regressors.
  - Too many regressors are trying to do the same job, so we should remove some of them.
- Centering the regressors (subtracting their mean)
- Scaling the regressors (dividing by their standard deviation).
- Standardizing regressors.
- Combining the collinear regressors into a single regressor.

Drawbacks

- Removing a regressor from the model that has a non-zero coefficient will result in a biased fitted model.

*The intercept column of X becomes orthogonal to the other regressors if the other regressors are centered. - In that case, the interpretation of the intercept becomes that it is the mean response when the regressors are at their sample mean values.

- Centering a predictor BEFORE using it to construct polynomial terms can help mitigate problems with collinearity among the polynomial terms, but will not remove all problems.

Summary

- Identify regressors with high pairwise correlation. Only keep one of the regressors.
- Remove regressors with large variance inflation factors since they have a strong linear relationship with others regressors.
- Look at the variance decomposition proportions and, for rows with large condition indices, identify the regressors that have a total proportion of 0.5 or more when added together across rows.

Example

```
print_colldiag(lm1)
```

intercept	Age	Weight	HtShoes	Ht	Seated	Arm	Thigh	Leg	cond.index
0	0.001	0	0	0	0	0	0	0	1
0	0.485	0.002	0	0	0	0	0	0	7.833
0.007	0.002	0.349	0	0	0	0	0.001	0	17.196
0.051	0.022	0.084	0	0	0.005	0.044	0.464	0	43.518
0.03	0.045	0.188	0	0	0.001	0.402	0.287	0.071	55.578
0.092	0.259	0.12	0	0	0.001	0.515	0.002	0.514	79.522
0.804	0.11	0.244	0.002	0.002	0.13	0.004	0.045	0.227	116.37
0.016	0.001	0.003	0.016	0.015	0.862	0.027	0.12	0.185	213.599
0	0.075	0.011	0.981	0.983	0.001	0.008	0.08	0.002	1153.483

Notice that `Ht` and `HtShoes` have very large variance decomposition proportions for the largest condition index.

Iteratively remove regressors and recompute condition indices until the problem is fixed.

Remove `HtShoes`

```
lm2 = update(lm1, .~.-HtShoes)
print_colldiag(lm2)
```

intercept	Age	Weight	Ht	Seated	Arm	Thigh	Leg	cond.index
0	0.001	0	0	0	0	0	0	1
0	0.518	0.003	0	0	0	0	0	7.446
0.008	0.004	0.348	0	0	0	0.002	0	16.319
0.059	0.02	0.088	0	0.006	0.04	0.479	0	41.24
0.031	0.048	0.19	0	0.002	0.402	0.302	0.071	52.344
0.072	0.262	0.106	0.001	0	0.511	0.003	0.552	75.19
0.83	0.129	0.244	0.041	0.258	0.005	0.031	0.143	116.865
0	0.018	0.022	0.957	0.733	0.042	0.183	0.234	245.363

Remove `Seated`

```
lm3 = update(lm2, .~.-Seated)
print_colldiag(lm3)
```

intercept	Age	Weight	Ht	Arm	Thigh	Leg	cond.index
0	0.002	0	0	0	0	0	1
0	0.524	0.004	0	0	0.001	0	7.048
0.012	0.007	0.354	0	0	0.004	0	15.58
0.113	0.008	0.076	0.002	0.021	0.572	0.004	40.364
0.074	0.046	0.23	0	0.433	0.252	0.062	49.348
0.106	0.27	0.118	0.002	0.51	0.004	0.565	70.297
0.695	0.143	0.219	0.995	0.035	0.167	0.368	150.424

Remove **Arm**

```
lm4 = update(lm3, .~.-Arm)
print_colldiag(lm4)
```

intercept	Age	Weight	Ht	Thigh	Leg	cond.index
0	0.003	0	0	0	0	1
0	0.812	0.004	0	0.001	0	6.532
0.013	0.01	0.349	0	0.005	0	14.412
0.093	0.001	0.042	0.002	0.718	0.006	37.643
0.219	0.059	0.38	0.001	0.067	0.5	55.459
0.676	0.113	0.224	0.997	0.21	0.494	136.988

Remove `Leg`

```
lm5 = update(lm4, .~.-Leg)
print_colldiag(lm5)
```

intercept	Age	Weight	Ht	Thigh	cond.index
0	0.005	0.001	0	0	1
0	0.822	0.007	0	0.001	6.086
0.015	0.013	0.34	0.001	0.006	13.167
0.126	0	0.068	0.004	0.679	34.499
0.859	0.16	0.585	0.994	0.314	95.56

Remove **Weight**

```
lm6 = update(lm5, .~.-Weight)
print_colldiag(lm6)
```

intercept	Age	Ht	Thigh	cond.index
0	0.009	0	0	1
0.002	0.919	0.001	0.003	5.636
0.389	0	0	0.445	28.236
0.608	0.072	0.998	0.552	56.69

```
lm7 = update(lm6, .~.-Thigh)
print_colldiag(lm7)
```

intercept	Age	Ht	cond.index
0	0.017	0	1
0.005	0.955	0.007	5.121
0.994	0.028	0.993	37.433

```
sumary(lm7)
```

```
##               Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 526.95889   92.24788  5.7124 1.848e-06
## Age           0.52106    0.38625  1.3490     0.186
```

Conclusion

If all the variables must be kept in the model, an alternative regression procedure such as ridge regression may be more appropriate.

The effect of collinearity on prediction depends on where the prediction is to be made.

- The greater the distance is from the observed data, the more unstable the prediction.
- Distance needs to be considered in a Mahalanobis (accounting for the correlation between predictors) rather than a Euclidean sense.

Conclusion

Note: You really should assess collinearity right after exploratory data analysis and before variable selection.

If your regressors are collinear, then all the subsequent inference is suspect and none of the diagnostics require you to fit a model first.

It is better to remove collinear variables first, then proceed with analysis.