Multiple Linear Regression: Least squares, colinearity

Author: Nicholas G Reich, Jeff Goldsmith

This material is part of the **statsTeachR** *project*

Today's topics

- least squares for MLR: geometry, "hat matrix"
- collinearity and non-identifiability
- categorical predictors

Example: predicting respiratory disease severity ("lung" dataset)

Multiple linear regression model

- Observe data $(y_i, x_{i1}, \ldots, x_{ip})$ for subjects $1, \ldots, n$. Want to estimate $\beta_0, \beta_1, \ldots, \beta_p$ in the model

$$y_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip} + \epsilon_i; \; \epsilon_i \overset{iid}{\sim} (0, \sigma^2)$$

Assumptions

- Residuals have mean zero, constant variance, are independent
- Often assuming linearity
- Our primary interest will be $E(y|\mathbf{x})$
- Estimation using least squares

Déjà vu: Least squares

As in simple linear regression, we want to find the $\boldsymbol{\beta}$ that minimizes the residual sum of squares.

$$RSS(\boldsymbol{\beta}) = \sum_i \epsilon_i^2 = \epsilon^T \epsilon$$

After taking the derivative, setting equal to zero, we obtain:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

Not so Déjà vu: the "Hat matrix"

The Hat Matrix

The hat matrix transforms the observed **y** into the fitted values.

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = \mathbf{H}\mathbf{y}$$

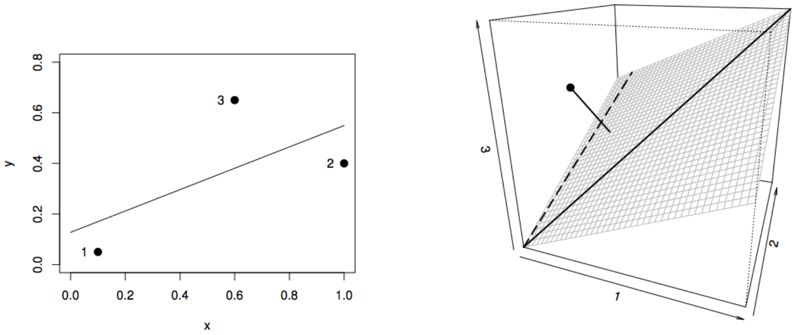
Some properties of the hat matrix:

- It is a projection matrix: $\mathbf{H}\mathbf{H} = \mathbf{H}$
- It is symmetric: $\mathbf{H}^T = \mathbf{H}$
- The residuals are $\hat{\epsilon} = (\mathbf{I} - \mathbf{H})\mathbf{y}$
- The inner product of $(\mathbf{I} - \mathbf{H})\mathbf{y}$ and $\mathbf{H}\mathbf{y}$ is zero (predicted values and residuals are uncorrelated).

Projection space interpretation

The hat matrix projects $\mathbf{y}$ onto the column space of $\mathbf{X}$.
Alternatively, minimizing the $RSS(\beta)$ is equivalent to minimizing the Euclidean distance between $\mathbf{y}$ and the column space of $\mathbf{X}$.

Geometry of least squares (in 3D)



$$\begin{bmatrix} .05 \\ .40 \\ .65 \end{bmatrix} = \begin{bmatrix} 1 & .1 \\ 1 & 1 \\ 1 & .6 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix}$$

* Image credits: Simon Wood, *Core Statistics*, Figure 7.1.

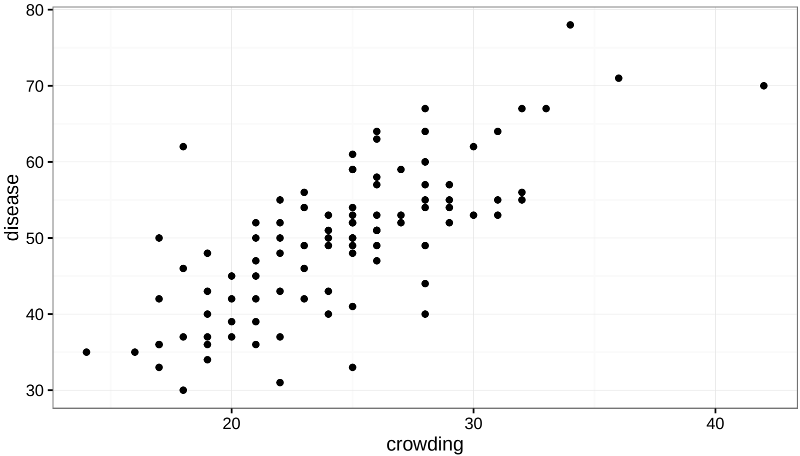
Lung Data Example

99 observations on patients who have sought treatment for the relief of respiratory disease symptoms.
The variables are:

- `disease` measure of disease severity (larger values indicates more serious condition).
- `education` highest grade completed
- `crowding` measure of crowding of living quarters (larger values indicate more crowding)
- `airqual` measure of air quality at place of residence (larger number indicates poorer quality)
- `nutrition` nutritional status (larger number indicates better nutrition)
- `smoking` smoking status (1 if smoker, 0 if non-smoker)

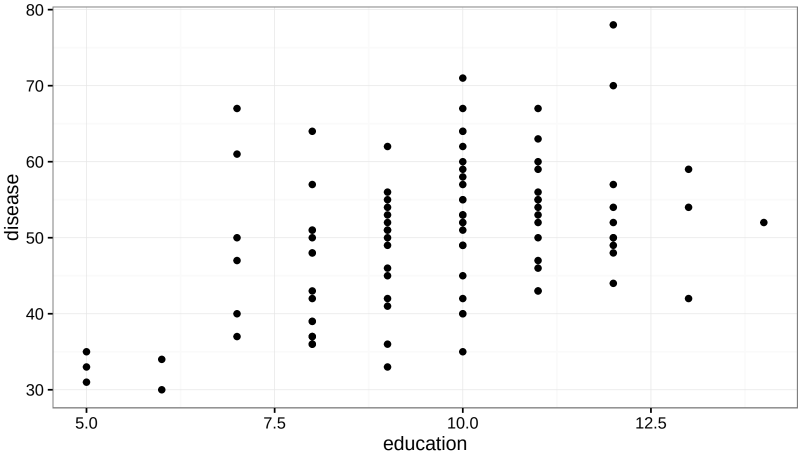
Lung Data Example

```
qplot(crowding, disease, data=dat)
```



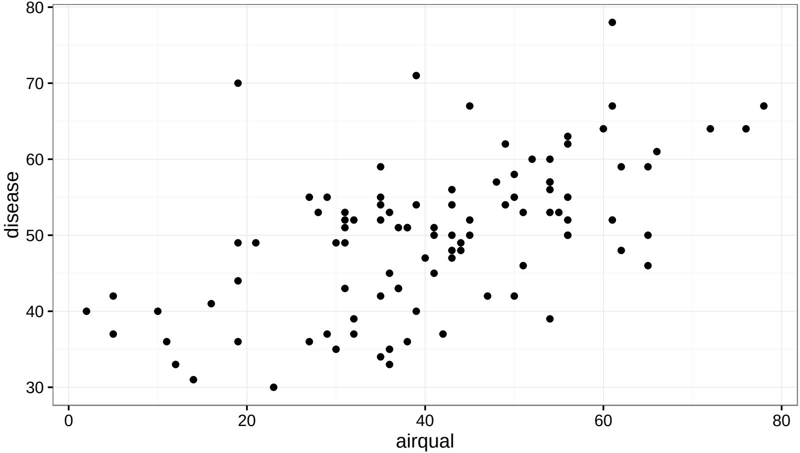
Lung Data Example

```
qplot(education, disease, data=dat)
```



Lung Data Example

```
qplot(airqual, disease, data=dat)
```



Lung Data Example

```
mlr1 <- lm(disease ~ crowding + education + airqual,
           data=dat, x=TRUE, y=TRUE)
coef(mlr1)

## (Intercept)    crowding   education     airqual
##  -7.7505215   1.3127837   1.4376563   0.2880687

X = mlr1$x
y = mlr1$y
(beta_hat = solve(t(X)%*%X) %*% t(X) %*% y )

##                   [,1]
## (Intercept) -7.7505215
## crowding     1.3127837
## education    1.4376563
## airqual      0.2880687
```

Least squares estimates: identifiability

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$$

A condition on $\left(\mathbf{X}^T\mathbf{X}\right)$: must be invertible

- If $\left(\mathbf{X}^T\mathbf{X}\right)$ is singular, there are infinitely many least squares solutions, making $\hat{\boldsymbol{\beta}}$ non-identifiable (can't choose between different solutions)
- In practice, true **non-identifiability** (there really are infinite solutions) is rare.
- More common, and perhaps more dangerous, is **collinearity**.

Causes of non-identifiability

- Can happen if $\mathbf{X}$ is not of full rank, i.e. the columns of $\mathbf{X}$ are linearly dependent (for example, including weight in Kg and lb as predictors)
- Can happen if there are fewer data points than terms in $\mathbf{X}$: $n < p$ (having 100 predictors and only 50 observations)
- Generally, the $p \times p$ matrix $\left(\mathbf{X}^T\mathbf{X}\right)$ is invertible if and only if it has rank p.

Infinite solutions

Suppose I fit a model $y_i = \beta_0 + \beta_1 x_{i1} + \epsilon_i$.

- I have estimates $\hat{\beta}_0 = 1, \hat{\beta}_1 = 2$
- I put in a new variable $x_2 = x_1$
- My new model is $y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \epsilon_i$
- Possible least squares estimates that are equivalent to my first model:
  - $\hat{\beta}_0 = 1, \hat{\beta}_1 = 2, \hat{\beta}_2 = 0$
  - $\hat{\beta}_0 = 1, \hat{\beta}_1 = 0, \hat{\beta}_2 = 2$
  - $\hat{\beta}_0 = 1, \hat{\beta}_1 = 1002, \hat{\beta}_2 = -1000$
  - $\ldots$

Non-identifiability example: lung data

```
mlr3 <- lm(disease ~ airqual, data=dat)
coef(mlr3)

## (Intercept)     airqual
##  35.4444812   0.3537389

dat$x2 <- dat$airqual/100
mlr4 <- lm(disease ~ airqual + x2, data=dat, x=TRUE)
coef(mlr4)

## (Intercept)     airqual          x2
##  35.4444812   0.3537389          NA

X = mlr4$x
solve( t(X) %*% X)

## Error in solve.default(t(X) %*% X): system is computationally
singular:  reciprocal condition number = 3.57906e-20
```

Non-identifiablity: causes and solutions

- Often due to data coding errors (variable duplication, scale changes)
- Pretty easy to detect and resolve
- Can be addressed using *penalties* (might come up much later)
- A bigger problem is near-unidentifiability (collinearity)

Diagnosing collinearity

- Arises when variables are highly correlated, but not exact duplicates
- Commonly arises in data (perfect correlation is usually there by mistake)
- Might exist between several variables, i.e. a linear combination of several variables exists in the data
- A variety of tools exist (correlation analyses, multiple R^2, eigen decompositions)

Effects of collinearity

Suppose I fit a model $y_i = \beta_0 + \beta_1 x_{i1} + \epsilon_i$.

- I have estimates $\hat{\beta}_0 = 1, \hat{\beta}_1 = 2$
- I put in a new variable $x_2 = x_1 + \mathit{error}$, where *error* is pretty small
- My new model is $y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \epsilon_i$
- Possible least squares estimates that are nearly equivalent to my first model:
  - $\hat{\beta}_0 = 1, \hat{\beta}_1 = 2, \hat{\beta}_2 = 0$
  - $\hat{\beta}_0 = 1, \hat{\beta}_1 = 0, \hat{\beta}_2 = 2$
  - $\hat{\beta}_0 = 1, \hat{\beta}_1 = 1002, \hat{\beta}_2 = -1000$
  - $\ldots$
- A unique solution exists, but it is hard to find

Effects of collinearity

- Collinearity results in a “flat” RSS
- Makes identifying a unique solution difficult
- Dramatically inflates the variance of LSEs

Collinearity example: lung data

```
dat$crowd2 <- dat$crowding + rnorm(nrow(dat), sd=.1)
mlr5 <- lm(disease ~ crowding, data=dat)
summary(mlr5)$coef

##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 12.991536  3.4750250  3.738544 3.130355e-04
## crowding     1.508806  0.1393709 10.825836 2.231686e-18

mlr6 <- lm(disease ~ crowding + crowd2, data=dat)
summary(mlr6)$coef

##              Estimate Std. Error    t value     Pr(>|t|)
## (Intercept) 13.278967   3.510938  3.7821704 0.0002702813
## crowding     5.570542   6.036769  0.9227688 0.3584411012
## crowd2      -4.073836   6.053130 -0.6730131 0.5025558447
```

Some take away messages

- Collinearity can (and does) happen, so be careful
- Often contributes to the problem of variable selection, which we'll touch on later

Categorical predictors

- Assume X is a categorical / nominal / factor variable with k levels
- With only one categorical X, we have classic one-way ANOVA design
- Can't use a single predictor with levels $1, 2, \ldots, K$ – this has the wrong interpretation
- Need to create *indicator* or *dummy* variables

Indicator variables

- Let x be a categorical variable with k levels (e.g. with $k = 3$ "red", "green", "blue").
- Choose one group as the baseline (e.g. "red")
- Create $(k-1)$ binary terms to include in the model:

$$\begin{aligned} x_{1,i} &= \mathbb{1}(x_i = \text{"green"}) \\ x_{2,i} &= \mathbb{1}(x_i = \text{"blue"}) \end{aligned}$$

- For a model with no additional predictors, pose the model

$$y_i = \beta_0 + \beta_1 x_{1,i} + \ldots + \beta_{k-1} x_{k-1,i} + \epsilon_i$$

  and estimate parameters using least squares
- Note distinction between *predictors* and *terms*

Categorical predictor design matrix

Which of the following is a "correct" design matrix for a categorical predictor with 3 levels?

$$\mathbf{X}_1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 1 \end{bmatrix} \quad \text{or} \quad \mathbf{X}_2 = \begin{bmatrix} 1 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ \vdots & \vdots & \vdots \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ \vdots & \vdots & \vdots \\ 1 & 0 & 1 \end{bmatrix} \quad \text{or} \quad \mathbf{X}_3 = \begin{bmatrix} 1 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 1 \end{bmatrix}$$

ANOVA model interpretation

Using the model $y_i = \beta_0 + \beta_1 x_{1,i} + \ldots + \beta_{k-1} x_{k-1,i} + \epsilon_i$, interpret

$\beta_0 =$

$\beta_1 =$

Equivalent model

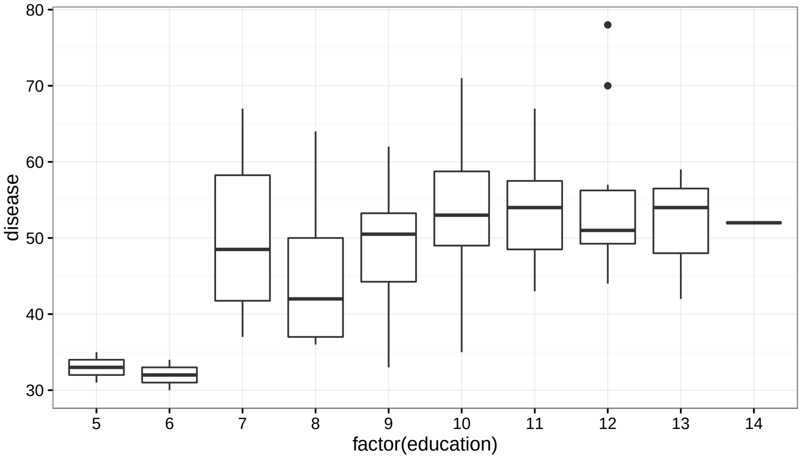
Define the model $y_i = \beta_1 x_{i1} + \ldots + \beta_k x_{i,k} + \epsilon_i$ where there are indicators for each possible group

$\beta_1 =$

$\beta_2 =$

Categorical predictor example: lung data

```
qplot(factor(education), disease, geom="boxplot", data=dat)
```



Categorical predictor example: lung data

$$dis_i = \beta_0 + \beta_1 educ_{6,i} + \beta_2 educ_{7,i} + \cdots + \beta_9 educ_{14,i}$$

```
mlr7 <- lm(disease ~ factor(education), data=dat)
summary(mlr7)$coef

##                     Estimate Std. Error    t value
## (Intercept)         33.00000   4.912705  6.7172765
## factor(education)6  -1.00000   7.767669 -0.1287387
## factor(education)7  17.33333   6.016811  2.8808175
## factor(education)8  11.17647   5.328577  2.0974588
## factor(education)9  15.50000   5.353496  2.8953040
## factor(education)10 20.38462   5.188395  3.9288865
## factor(education)11 20.53333   5.381599  3.8154707
## factor(education)12 22.20000   5.601346  3.9633332
## factor(education)13 18.66667   6.947614  2.6867735
## factor(education)14 19.00000   9.825411  1.9337614
##                         Pr(>|t|)
## (Intercept)         1.689481e-09
## factor(education)6  8.978549e-01
## factor(education)7  4.969406e-03
## factor(education)8  3.878868e-02
```

Categorical predictor releveling

$$dis_i = \beta_0 + \beta_1 educ_{5,i} + \beta_2 educ_{6,i} + \beta_1 educ_{7,i} + \beta_2 educ_{9,i} + \cdots + \beta_{14} educ_{14,i}$$

```
dat$educ_new <- relevel(factor(dat$education), ref="8")
mlr8 <- lm(disease ~ educ_new, data=dat)
summary(mlr8)$coef

##                Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)  44.176471   2.063749 21.4059318 7.303151e-37
## educ_new5   -11.176471   5.328577 -2.0974588 3.878868e-02
## educ_new6   -12.176471   6.360902 -1.9142680 5.879890e-02
## educ_new7     6.156863   4.040594  1.5237520 1.311162e-01
## educ_new9     4.323529   2.963834  1.4587624 1.481508e-01
## educ_new10    9.208145   2.654021  3.4695065 8.059293e-04
## educ_new11    9.356863   3.014298  3.1041594 2.558604e-03
## educ_new12   11.023529   3.391086  3.2507375 1.625933e-03
## educ_new13    7.490196   5.328577  1.4056653 1.633049e-01
## educ_new14    7.823529   8.755746  0.8935309 3.739828e-01
```

Categorical predictor: no baseline group

$$dis_i = \beta_1 educ_{5,i} + \beta_2 educ_{6,i} + \cdots + \beta_{14} educ_{14,i}$$

```
mlr9 <- lm(disease ~ factor(education) - 1, data=dat)
summary(mlr9)$coef

##                      Estimate Std. Error   t value
## factor(education)5   33.00000   4.912705  6.717277
## factor(education)6   32.00000   6.016811  5.318432
## factor(education)7   50.33333   3.473807 14.489386
## factor(education)8   44.17647   2.063749 21.405932
## factor(education)9   48.50000   2.127264 22.799241
## factor(education)10  53.38462   1.668763 31.990531
## factor(education)11  53.53333   2.197029 24.366243
## factor(education)12  55.20000   2.690800 20.514349
## factor(education)13  51.66667   4.912705 10.516948
## factor(education)14  52.00000   8.509055  6.111137
##                          Pr(>|t|)
## factor(education)5   1.689481e-09
## factor(education)6   7.715960e-07
## factor(education)7   3.845787e-25
## factor(education)8   7.303151e-37
```

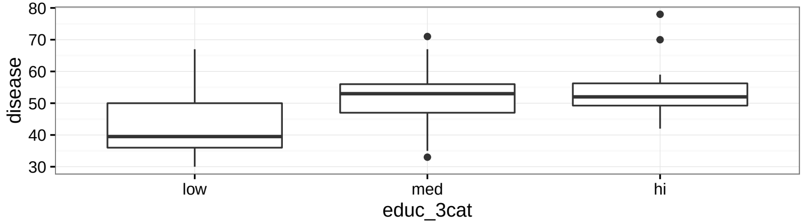
Creating categories using cut()

$$dis_i = \beta_1 educ_{low,i} + \beta_2 educ_{med,i} + \cdots + \beta_{14} educ_{hi,i}$$

```
dat$educ_3cat <- cut(dat$education, breaks=3,
                     labels=c("low", "med", "hi"))
mlr10 <- lm(disease ~ educ_3cat - 1, data=dat)
coef(mlr10)

## educ_3catlow educ_3catmed  educ_3cathi
##     43.42857     52.05263     54.21429

qplot(educ_3cat, disease, geom="boxplot", data=dat)
```



Today's big ideas

- least squares geometry, "hat matrix"
- dangers of collinearity and non-identifiability
- categorical predictors