Resampling Applications

Max Turgeon

STAT 3150–Statistical Computing

Lecture Objectives

- Review multiple linear regression and residual analysis.
- Recognize the relative importance of regression assumptions.
- Understand the difference between resampling cases vs residuals.

Motivation

- In the last module, we discussed how bootstrap can be used to approximate the sampling distribution.
    - The general idea is based on replacing the true CDF by the empirical CDF (i.e. sampling with replacement).
- If we make assumptions about the data-generating mechanism, we can sometimes improve on the general bootstrap.
- We will explore this idea using **linear regression**.

Recall: Linear model

- Y is an outcome variable, $X_1, \dots, X_p$ are covariates.

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon.$$

- Here, ϵ is a random variable with mean 0 and variance σ^2, so we can also write

$$E(Y \mid X_1, \dots, X_p) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p.$$

- In matrix notation, we have

$$E(Y, \mid \mathbf{X}) = \beta^T \mathbf{X},$$

where

$$\beta = (\beta_0, \beta_1, \dots, \beta_p),$$
$$\mathbf{X} = (1, X_1, \dots, X_p).$$

Least-Squares Estimation

- Let $Y_1, \ldots, Y_n$ be a random sample of size n, and let $\mathbf{X}_1, \ldots, \mathbf{X}_n$ be the corresponding sample of covariates.
- We will write $\mathbb{Y}$ for the vector whose i-th element is Y_i, and $\mathbb{X}$ for the matrix whose i-th row is $\mathbf{X}_i$.
- The Least-Squares estimate $\hat{\beta}$ is given by

$$\hat{\beta} = (\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\mathbb{Y}.$$

Assumptions

Gelman, Hill and Vehtari (2020) list the assumptions of linear regression **in decreasing order of importance**:

1. Validity (with respect to the research question).
2. Representativeness (of the data with respect to the population).
3. Additivity and linearity.
4. Independence of errors.
5. Equal variance of errors.
6. Normality of errors.

Additivity and linearity

- Main mathematical assumption:

$$E(Y \mid X_1, \ldots, X_p) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p.$$

- Or in English:
    - Changes in the conditional mean of Y should be additive and linear.
- **Note**: Conditional mean = on average
    - Life is probably nonlinear and non-additive…
    - But it can still be a good approximation of the average

Diagnostic plots

1. For **simple** linear regression (i.e. only one covariate), plot outcome against covariate.
2. Plot outcome against fitted values.
3. Plot residuals against fitted values and/or covariates.

Note: It is not recommended to plot outcome against residuals.

First example i

- Data contains 53 observations of iron measurements, obtained via two methods: `chemical` and `magnetic`.

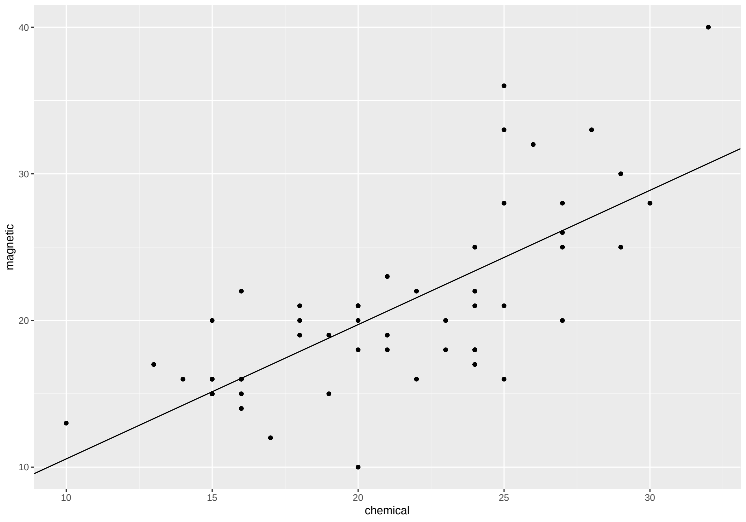
```
library(DAAG)
library(ggplot2)

# Fit model
fit1 <- lm(magnetic ~ chemical, data=ironslag)
coef(fit1)
```

First example ii

```
## (Intercept)    chemical
##   1.4025974   0.9157699
```

```r
# Plot fitted linear trend
ggplot(ironslag, aes(chemical, magnetic)) +
  geom_point() +
  geom_abline(intercept = coef(fit1)[1],
              slope = coef(fit1)[2])
```

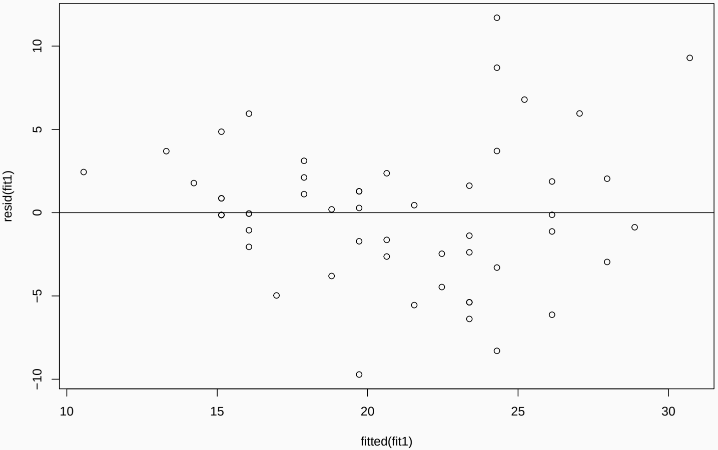
First example iii



First example iv

```r
# Fitted against residuals
plot(fitted(fit1), resid(fit1))
abline(h = 0)
```

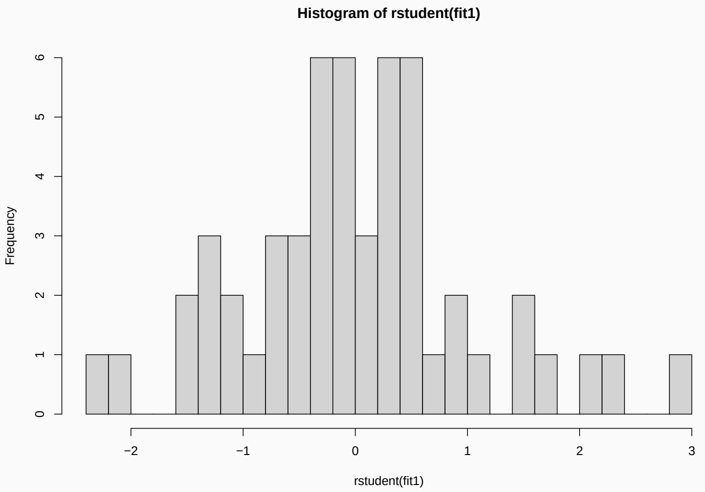
First example v



First example vi

```r
# Histogram of student residuals
hist(rstudent(fit1), 20)
```

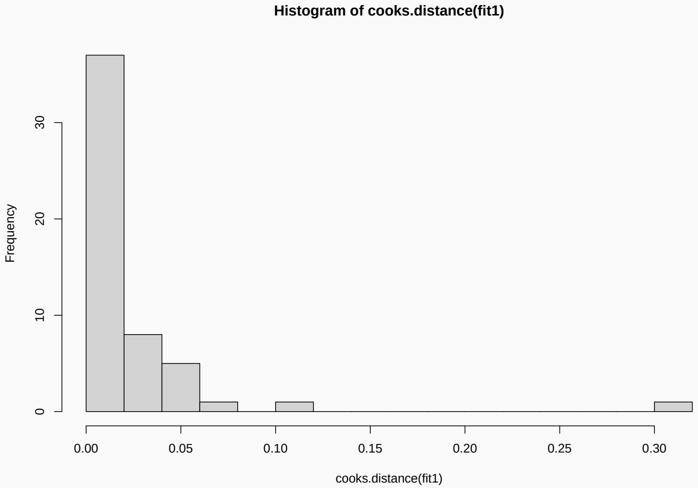
First example vii



First example viii

```r
# Histogram of Cook’s distance
hist(cooks.distance(fit1), 20)
```

First example ix



First example x

- The residual plot shows evidence of heteroscedasticity.
    - **Conclusion**: Some assumptions of the linear model are likely violated.
- There is also evidence of potential outliers and influential observations.

Second example i

- Data contains body and brain size measurements for 62 mammals.
- We will fit a linear model of the *log* brain size vs the *log* body size

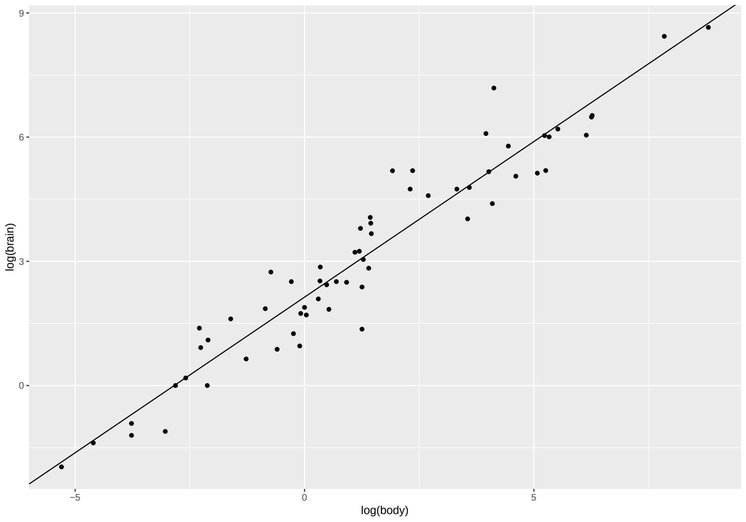
```r
library(MASS)
library(ggplot2)

# Fit model
fit2 <- lm(log(brain) ~ log(body), data = mammals)
coef(fit2)
```

Second example ii

```
## (Intercept)   log(body)
##   2.1347887   0.7516859
```

```
# Plot fitted linear trend
ggplot(mammals, aes(log(body), log(brain))) +
  geom_point() +
  geom_abline(intercept = coef(fit2)[1],
              slope = coef(fit2)[2])
```

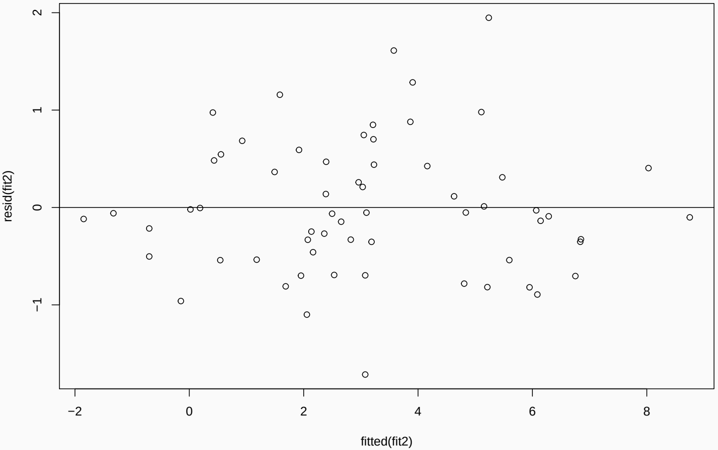
Second example iii



Second example iv

```r
# Fitted against residuals
plot(fitted(fit2), resid(fit2))
abline(h = 0)
```

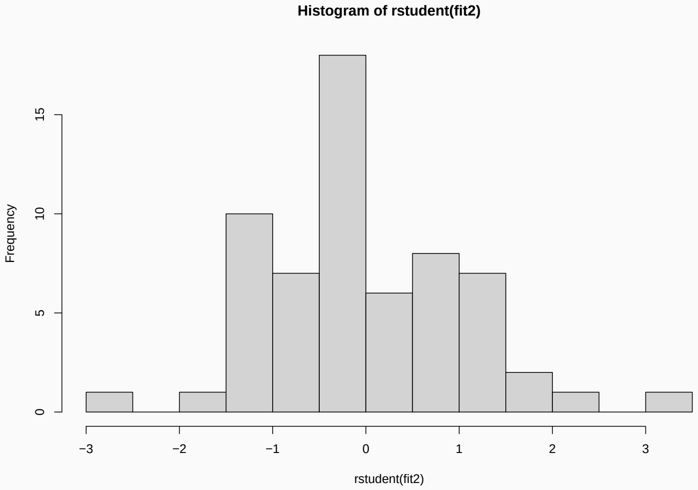
Second example v



Second example vi

```r
# Histogram of student residuals
hist(rstudent(fit2), 20)
```

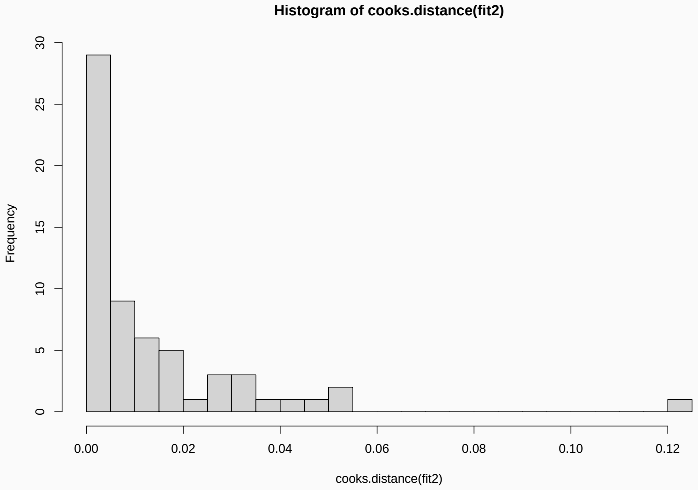
Second example vii



Second example viii

```r
# Histogram of Cook's distance
hist(cooks.distance(fit2), 20)
```

Second example ix



Second example x

- The residual plot *does not* show evidence of heteroscedasticity or any model violation.
    - **Conclusion**: The assumptions of the linear model likely hold.
- There is some evidence of potential outliers and influential observations.

Bootstrap and Linear regression i

- When the error term is normally distributed, we know the distribution of $\hat{\beta}$:

$$\hat{\beta} \sim N\left(\beta, \sigma^2(\mathbb{X}^T\mathbb{X})^{-1}\right).$$

  - This can be used to compute p-values and confidence intervals.
- But when we *don't know* the distribution, or if we *don't want* to assume it follows a normal distribution, we can use bootstrap to make valid inference.

Bootstrap and Linear regression ii

- As we will see, there are two different ways to use bootstrap:
  - **Resample cases**;
  - **Resample residuals**.
- The main difference is how many assumptions we want to retain:
  - To resample residuals, we need to assume additivity, linearity, and homoscedasticity.
- In both cases, we still need to assume **independence of the errors**.

Resampling cases

- This is the simplest form of bootstrap for linear regression.
    - It should also be familiar.
- For this form of bootstrap to be valid, we only need to assume the errors are independent.
- In fact, it can be shown that when resampling cases, the bootstrap estimate of the standard error is approximately equal to the Huber-White robust standard error.

Algorithm (Cases)

1. Sample with replacement from $(Y_1, \mathbf{X}_1), \ldots, (Y_n, \mathbf{X}_n)$.
2. Refit the linear model using the bootstrap sample and obtain bootstrap estimates $\hat{\beta}^{(b)}$.

First example cont'd i

```r
n <- nrow(ironslag)
boot_beta1 <- replicate(5000, {
  indices <- sample(n, n, replace = TRUE)
  fit_boot <- lm(magnetic ~ chemical,
                 data = ironslag[indices, ])
  coef(fit_boot)
})
```

```r
str(boot_beta1)
```

First example cont'd ii

```
## num [1:2, 1:5000] 0.788 0.95 1.999 0.901 2.574
...
## - attr(*, "dimnames")=List of 2
## ..$ : chr [1:2] "(Intercept)" "chemical"
## ..$ : NULL
```

```
se_int <- sd(boot_beta1[1,])
se_slope <- sd(boot_beta1[2,])

cbind("Lower" = coef(fit1) - 1.96*c(se_int, se_slope),
      "Upper" = coef(fit1) + 1.96*c(se_int, se_slope))
```

First example cont'd iii

```
##                   Lower    Upper
## (Intercept) -3.1964989 6.001694
## chemical     0.6759036 1.155636
```

```r
# Compare to MLE theory
confint(fit1)
```

```
##                  2.5 %   97.5 %
## (Intercept) -3.7856893 6.590884
## chemical     0.6768355 1.154704
```

First example cont'd iv

- Our confidence interval for the intercept is a bit smaller, but it still includes 0.
- On the other hand, the confidence interval for `chemical` is comparable to the one from MLE theory.

Resampling residuals i

- As mentioned above, this approach requires more assumptions than resampling cases:
    - Additivity and linearity;
    - Homoscedasticity.
- But the trade-off is that we get smaller confidence intervals than if we resample cases.

Resampling residuals ii

Algorithm (Residuals)
First, compute residuals E_i and fitted values $\hat{Y}_i = \hat{\beta}^T \mathbf{X}_i$ for each observation $i = 1, \ldots, n$.

1. Sample with replacement from the residuals and obtain a bootstrap sample $E_1^{(b)}, \ldots, E_n^{(b)}$.
2. Add the bootstrapped residuals to the fitted values: $Y_i^{(b)} = \hat{Y}_i + E_i^{(b)}$.
3. Using these new outcomes $Y_i^{(b)}$ and the original covariates $\mathbf{X}_i$, fit a linear regression model and obtain bootstrap estimates $\hat{\beta}^{(b)}$.

Second example cont'd i

```r
# Compute residuals
resids <- resid(fit2)

n <- length(resids)
boot_beta2 <- replicate(5000, {
  indices <- sample(n, n, replace = TRUE)
  logbrain_boot <- fitted(fit2) + resids[indices]
  fit_boot <- lm(logbrain_boot ~ log(mammals$body))
  coef(fit_boot)
})
```

```
str(boot_beta2)
```

```
## num [1:2, 1:5000] 2.138 0.781 2.057 0.749
2.296 ...
## - attr(*, "dimnames")=List of 2
## ..$ : chr [1:2] "(Intercept)"
"log(mammals$body)"
## ..$ : NULL
```

Second example cont'd iii

```
se_int <- sd(boot_beta2[1,])
se_slope <- sd(boot_beta2[2,])

cbind("Lower" = coef(fit2) - 1.96*c(se_int, se_slope),
      "Upper" = coef(fit2) + 1.96*c(se_int, se_slope))
```

```
##                 Lower     Upper
## (Intercept) 1.950476 2.3191012
## log(body)   0.697057 0.8063148
```

Second example cont'd iv

```r
# Compare to MLE theory
confint(fit2)
```

```
##                 2.5 %    97.5 %
## (Intercept) 1.9426733 2.3269041
## log(body)   0.6947503 0.8086215
```

- This time, we can see that we get essentially the same result in both cases.
    - The bootstrap confidence intervals are slightly smaller.

Second example cont'd v

- **Note**: Other types of residuals can be used for the bootstrap, e.g. to mitigate the effect of outliers.
    - But don't use standardized residuals! You want the residuals to retain approximately the same variance as in the original data.

Final remarks i

- We looked at two different ways to perform bootstrap in the context of linear regression.
    - Resample the **cases** or the **residuals**.
- Resampling the cases is valid more generally than resampling the residuals.
- But resampling the residuals can lead to smaller, more accurate confidence intervals.
- Deciding which approach to use is a question of how much you trust the model.

Final remarks ii

- **Importantly**, neither approach is valid when the errors are *correlated*.
    - E.g. clustered data, repeated measurements, time series.
    - Bootstrap can be adapted to these methods, but this is beyond the scope of STAT 3150.