Residual Analysis

Max Turgeon

SCI 2000–Introduction to Data Science

Lecture Objectives

- Recognize the relative importance of regression assumptions.
- Interpret residual plots to determine whether the assumptions are likely to be met.

Motivation

- In the previous lecture, we talked about how to fit a linear regression model in R, and how it relates to common statistical procedures (e.g. t-test and ANOVA).
- But we haven't talked about assumptions yet!
    - I'll introduce them in the next slides.
- **Residual analysis** allows us to assess whether the assumptions are met and whether we should change our model.
    - We will focus on a *graphical* approach. In other courses, you may see different approaches.

Multiple Linear Regression

- Recall: Y is an outcome variable, $X_1, \dots, X_p$ are covariates.
- The linear regression equation is

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

- Some authors also write the following equation:

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

- Here, ϵ is a random variable with mean 0 and variance σ^2.
  - You can use either equation; I prefer the first one.

Fitted values and residuals i

- After we have estimated the regression coefficients $\beta_0, \beta_1, \ldots, \beta_p$, we can compute **fitted values** and **residuals**.
- We will use the hat notation to indicate that a parameter has been estimated:
  - β_0 is the (population) parameter.
  - $\hat{\beta}_0$ is the estimate from linear regression.
- Now assume we have our estimates $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$. For a given observation in our dataset, we also have a set of covariate values $X_{i1}, \ldots, X_{ip}$.

Fitted values and residuals ii

- We get the i-th **fitted value** by plugging all these values in the regression equation:

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_{i1} + \cdots + \hat{\beta}_p X_{ip}.$$

- We get the i-th **residual** by taking the difference between the observed value Y_i and the fitted value $\hat{Y}_i$:

$$\hat{e}_i = Y_i - \hat{Y}_i.$$

The fitted values and residuals can help us understand the fit of our regression model.

Assumptions of Linear Regression

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

A powerful way of detecting violations of the assumptions is using **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.

Example i

- Dataset `ironslag` from the `DAAG` package contains 53 observations of iron measurements, obtained via two methods: `chemical` and `magnetic`.

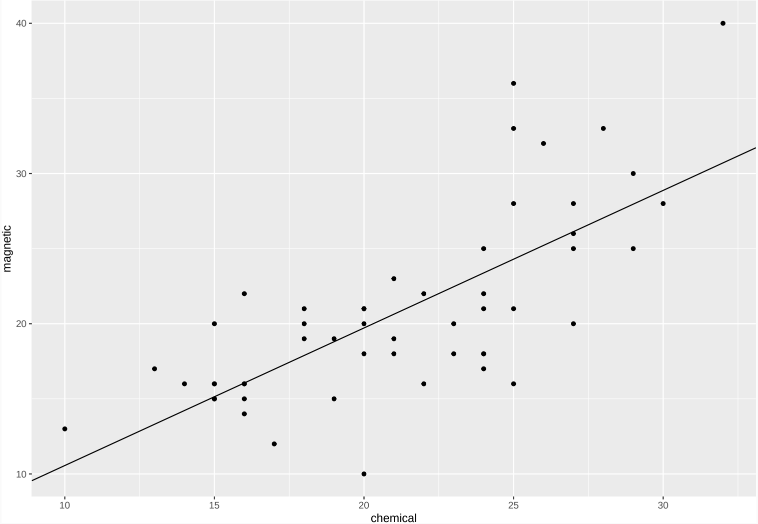
```r
library(DAAG)
library(tidyverse)

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

Example ii

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

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

Example iii



Example iv

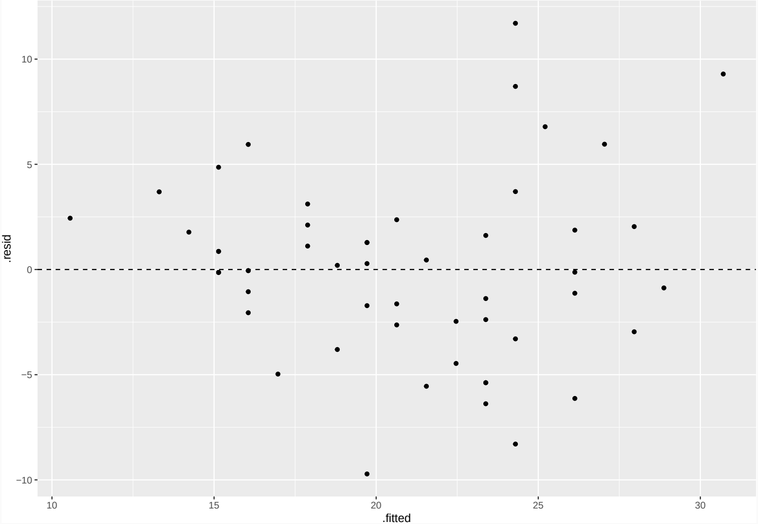
```r
library(broom)

# Augment adds fitted values and residuals
# to the original data
names(augment(fit))
```

```
## [1] ”magnetic” ”chemical” ”.fitted” ”.resid”
”.hat”
## [6] ”.sigma” ”.cooksd” ”.std.resid”
```

Example v

```
# Fitted against residuals
augment(fit) %>%
  ggplot(aes(x = .fitted,
             y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0,
             linetype = ”dashed”)
```

Example vi



Example vii

- The residual plot shows evidence of heteroscedasticity and violation of additivity/linearity.
- **Conclusion**: Some assumptions of the linear model are likely violated.

Exercise

Use the dataset `mammals` from the package `MASS`. Create a new variable `log_body` by using a log transformation on the body size measurement. Fit a linear model of `brain` by `log_body`. Investigate whether the assumptions hold.

Solution i

- Dataset contains body and brain size measurements for 62 mammals.

```r
library(MASS)
library(tidyverse)

dataset <- mutate(mammals,
                  log_body = log(body))

# Fit model
fit <- lm(brain ~ log_body, data = dataset)
```

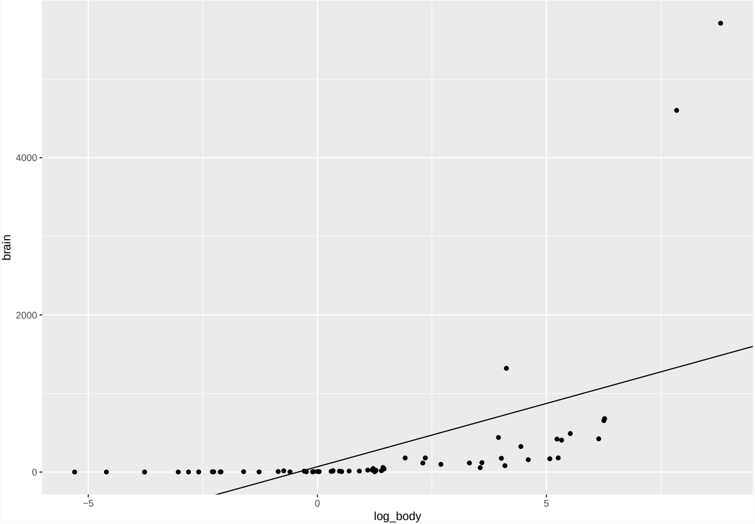
Solution ii

```
confint(fit)
```

```
##                    2.5 %   97.5 %
## (Intercept) -150.57636 286.1659
## log_body      96.27998 225.7135
```

```
# Plot fitted linear trend
ggplot(dataset, aes(x = log_body,
                    y = brain)) +
  geom_point() +
  geom_abline(intercept = coef(fit)[1],
              slope = coef(fit)[2])
```

Solution iii



Solution iv

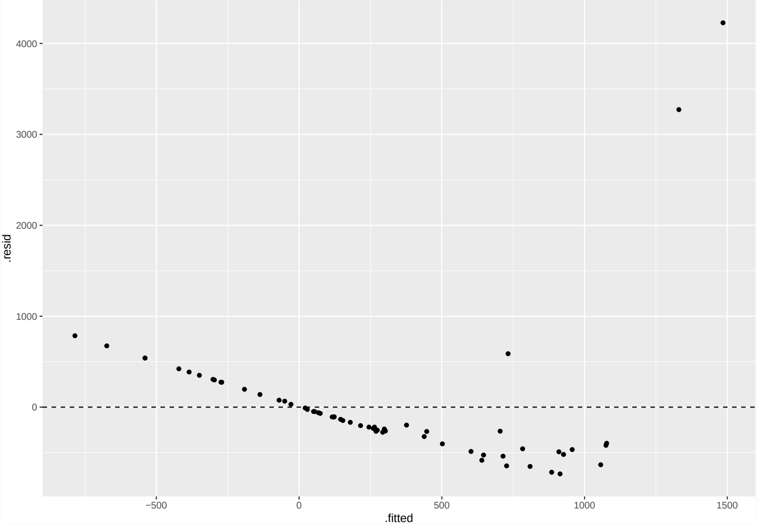
```
names(augment(fit))
```

```
## [1] ”.rownames” ”brain” ”log_body” ”.fitted”
”.resid”
## [6] ”.hat” ”.sigma” ”.cooksd” ”.std.resid”
```

Solution v

```
# Fitted against residuals
augment(fit) %>%
  ggplot(aes(x = .fitted,
             y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0,
             linetype = ”dashed”)
```

Solution vi



Solution vii

- There is clearly something wrong with our model...

Transforming variables i

- In the previous example, the relationship between `log_body` and `brain` started almost flat and then quickly jump up.
    - This looked like an exponential relationship...
- If we log-transform the outcome, the relationship should look more linear.

```
dataset <- mutate(mammals,
                  log_body = log(body),
                  log_brain = log(brain))

# Fit model
fit2 <- lm(log_brain ~ log_body, data = dataset)
```

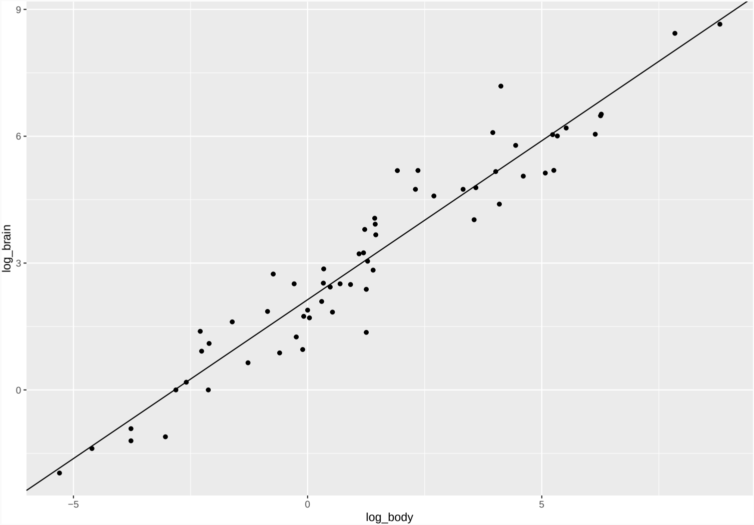
Transforming variables ii

```r
confint(fit2)
```

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

```r
# Plot fitted linear trend
ggplot(dataset, aes(x = log_body,
                    y = log_brain)) +
  geom_point() +
  geom_abline(intercept = coef(fit2)[1],
              slope = coef(fit2)[2])
```

Transforming variables iii



Transforming variables iv

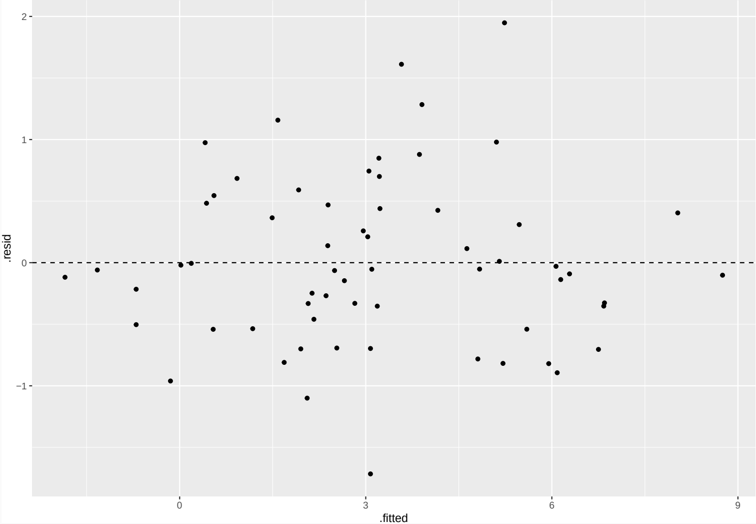
```
names(augment(fit2))
```

```
## [1] ”.rownames” ”log_brain” ”log_body”
”.fitted” ”.resid”
## [6] ”.hat” ”.sigma” ”.cooksd” ”.std.resid”
```

Transforming variables v

```
# Fitted against residuals
augment(fit2) %>%
  ggplot(aes(x = .fitted,
             y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0,
             linetype = ”dashed”)
```

Transforming variables vi



Transforming variables vii

- The residual plot shows little evidence of heteroscedasticity or any model violation.
- **Conclusion**: The assumptions of the linear model likely hold.

Lifecycle of a regression model

1. Model building (i.e. choosing the variables in your model)
2. Model fitting
3. **Understanding the fit** (e.g. residual analysis)
4. Criticism

Important: This is typically an iterative process.

Equal variance of errors i

- Equal variance (aka homoscedasticity) is actually a fairly unimportant assumption.
    - If the goal of the model is prediction, accounting for unequal variance will improve accuracy.
- Unequal variance (aka heteroscedasticity) does not affect the validity of the confidence intervals.
- However, accounting for unequal variance can lead to more efficient inference (i.e. lower variance, narrower CIs).

Equal variance of errors ii

- **When is it not met?** Unequal variance could simply be a feature of the data, and it is common to have the variance depend on covariates (e.g. higher income patients have more variability in their diet).
- **How to fix this?** Weighted linear regression (beyond the scope of this course) or Eicker–Huber–White standard errors (see below).
    - These can also help address dependent errors.

Example i

- Let’s go back to our first example:

```r
library(DAAG)
library(tidyverse)

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

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

Example ii

- The Eicker–Huber–White standard errors replace the usual standard errors used to construct the confidence intervals.
    - But it doesn’t affect the estimates themselves!

```r
library(lmtest)
library(sandwich)
coefci(fit, vcov. = vcovHC(fit))
```

```
##                   2.5 %   97.5 %
## (Intercept) -3.6068737 6.412069
## chemical     0.6546812 1.176859
```

Exercise

Compute robust confidence intervals for the regression model of `log_brain` vs `log_body`. Compare with the usual confidence intervals.

Solution i

```
dataset <- mutate(mammals,
                  log_body = log(body),
                  log_brain = log(brain))

# Fit model
fit2 <- lm(log_brain ~ log_body, data = dataset)
```

```
confint(fit2)
```

Solution ii

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

```
coefci(fit2, vcov. = vcovHC(fit2))
```

```
##                 2.5 %   97.5 %
## (Intercept) 1.9542558 2.315322
## log_body    0.7062298 0.797142
```

Summary

- Residual analysis allows us to evaluate the fit of our model.
    - How well does the model explain our dataset?
- The most important statistical assumption is **additivity and linearity**, i.e. that the regression equation holds.
- If it doesn't seem to hold, it means we need to change the regression model.
    - Transform variables.
    - Add more covariates.
- Equal variance is not as important.
- Non-normality of the errors is rarely a problem.