# Plan for the day

# Introduction to R

Silje Synnøve Lyder Hermansen

11 februar 2026

# Plan for the day

# Plan for the day

- lecture: uncertainty and interpretation of linear models
  - substantive interest: the size of the effect
  - statistical significance: sources of variation/uncertainty
- chatGPT/chatTutor: how to use AI/LLMs in this class

# Introduction

# Today's example

**What is the effect of electoral systems on parliamentarians resource allocation?**

- Members of the European Parliament (MEPs) sit together in one institution, but run for election under different rules
- expectation: more local investment among MEPs in candidate-centered systems (compared to party-centered systems), because of their need for a personal brand
- variables:
  - y: number of constituency-level assistants employed
  - x : candidate vs. party-centered systems

# Two views on linear regression

# Two views on linear regression

*Linear regression summarizes how the average values of a numerical outcome variable vary over subpopulations defined by linear functions of predictors. (Gelman and Hill, 2007, ch 3)*

- **comparison of means:** descriptive approach to regression; makes sense for categorical predictors
- **relationship between variables:** their correlation; more causal, makes sense for numerical predictors

# Regression as a comparison of means

```
df %>%
  group_by(OpenList) %>%
  reframe("mean_y" = mean(LocalAssistants)) %>%
  ungroup %>%
  mutate(diff = mean_y - lag(mean_y))
```

```
## # A tibble: 2 x 3
##   OpenList mean_y   diff
##      <int>  <dbl>  <dbl>
## 1        0   2.47 NA
## 2        1   3.42  0.949
```

- MEPs from *party-centered* systems employ on average 2.47 local assistants
- MEPs from *candidate-centered* systems employ on average 3.42 local assistants.
- The difference is 0.95

# Relationship between variables

```
mod <- lm(LocalAssistants ~ OpenList,
          df)

summary(mod)
```

```
##
## Call:
## lm(formula = LocalAssistants ~ OpenList, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -3.42  -2.42  -0.47   1.53  36.08
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.468      0.161   15.35  < 2e-16 ***
## OpenList       0.949      0.234    4.05  5.7e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.2 on 737 degrees of freedom
## Multiple R-squared:  0.0218, Adjusted R-squared:  0.0204
## F-statistic: 16.4 on 1 and 737 DF,  p-value: 5.68e-05
```

# Relationship between variables

```r
summary(mod)
```

```
##
## Call:
## lm(formula = LocalAssistants ~ OpenList, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -3.42  -2.42  -0.47   1.53  36.08
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.468      0.161   15.35  < 2e-16 ***
## OpenList       0.949      0.234    4.05  5.7e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.2 on 737 degrees of freedom
## Multiple R-squared:  0.0218, Adjusted R-squared:  0.0204
## F-statistic: 16.4 on 1 and 737 DF,  p-value: 5.68e-05
```

- MEPs from *party-centered* systems employ on average 2.47 local assistants
- The difference is 0.95.
- MEPs from *candidate-centered* systems employ on average $2.47 + 0.95 = 3.42$ local assistants.

# Interpretation

# Linear predictor

```r
mod2 <- lm(LocalAssistants ~ OpenList + LaborCost,
          df)

summary(mod2)
```

```
##
## Call:
## lm(formula = LocalAssistants ~ OpenList + LaborCost, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -4.49  -1.94  -0.41   1.08  35.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.1266     0.2861   14.42  < 2e-16 ***
## OpenList      0.8288     0.2278    3.64  0.00029 ***
## LaborCost    -0.0702     0.0102   -6.91    1e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.1 on 736 degrees of freedom
## Multiple R-squared:  0.0814, Adjusted R-squared:  0.0789
## F-statistic: 32.6 on 2 and 736 DF,  p-value: 2.69e-14
```

# Stages of interpretaion

- **hypothesis testing:** direction and signficance
- **marginal effect:** the relative increase in your predictor wo/accounting for the value of other preditors.
- **prediction**: fill in the equation for all predictors and calculate the predicted effect
- **first difference**: fill in the equation for two *scenarios* and calculate the difference in y
- **effect plot**: fill in the equation for all scenarios relevant to your predictor

$\Rightarrow$ *as we move to GLMs, the importance of stages 3-6 becomes important*

# Hypothesis testing

# Hypothesis testing

```
summary(mod2)
```

```
##
## Call:
## lm(formula = LocalAssistants ~ OpenList + LaborCost, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -4.49  -1.94  -0.41   1.08  35.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.1266     0.2861   14.42  < 2e-16 ***
## OpenList      0.8288     0.2278    3.64  0.00029 ***
## LaborCost    -0.0702     0.0102   -6.91    1e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.1 on 736 degrees of freedom
## Multiple R-squared:  0.0814, Adjusted R-squared:  0.0789
## F-statistic: 32.6 on 2 and 736 DF,  p-value: 2.69e-14
```

- MEPs from candidate-centered systems have on average more local assistants on their payroll

# Marginal effect

# Marginal effect

**The relative increase in your predictor wo/accounting for the value of other predictors.**

```
summary(mod2)
```

```
##
## Call:
## lm(formula = LocalAssistants ~ OpenList + LaborCost, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -4.49  -1.94  -0.41   1.08  35.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.1266     0.2861   14.42  < 2e-16 ***
## OpenList      0.8288     0.2278    3.64  0.00029 ***
## LaborCost    -0.0702     0.0102   -6.91    1e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.1 on 736 degrees of freedom
## Multiple R-squared:  0.0814, Adjusted R-squared:  0.0789
## F-statistic: 32.6 on 2 and 736 DF,  p-value: 2.69e-14
```

- when labor cost increases with 1 unit (here 1000 euros), the average number of assistants decreases by 0.07
- 10.000 euro increase (increase by 10) corresponds to a

# Prediction

# Prediction

**We make predictions by filling in the equation**

$$Y_i = \alpha + \beta X_i$$

$$Y_i = 4.13 + 0.95 \times X_i$$

**data (observed)**

- variables: X and Y
- observations: i is a counter for the observations, refers to the $i^{th}$ observation. $i \ldots N$

**parameters (estimated)**

- $\alpha$ intercept, the value of Y when X == 0
- $\beta$ slope, the increase in Y when X increases by one unit

# Creating scenarios

**You create a scenario when you fill in values in all the predictors (x).**

$$Y_i = \alpha + \beta X_i$$

$$5.08 = 4.13 + 0.95 \times 1$$

In R:

```
x = 1

# or

scenario <- data.frame(OpenList = 1)
```

# First difference

**You create two scenarios and calculate the difference in y**

$Y_i = \alpha + \beta X_i$

*scenario 1:* $4.13 = 4.13 + 0.95 \times 0$ *scenario 2:* $5.08 = 4.13 + 0.95 \times 1$

In R:

```
x = c(0, 1)

# or

scenario <- data.frame(OpenList = c(0, 1))
```

$\Rightarrow$ *The first difference is 0.95.*

# Effect plot

# Prediction

**You create a bunch of scenarios covering the entire range of the variable**

In R:

```
#Scenario
scenario <- data.frame(OpenList = c(0),
                       LaborCost = min(df$LaborCost): max(df$LaborCost))

scenario[1:3,]
```

```
##   OpenList LaborCost
## 1        0       3.8
## 2        0       4.8
## 3        0       5.8
```

```
#Predict
scenario <- scenario %>% mutate(preds = predict(mod2, newdata = scenario))
scenario$preds[1:3]
```

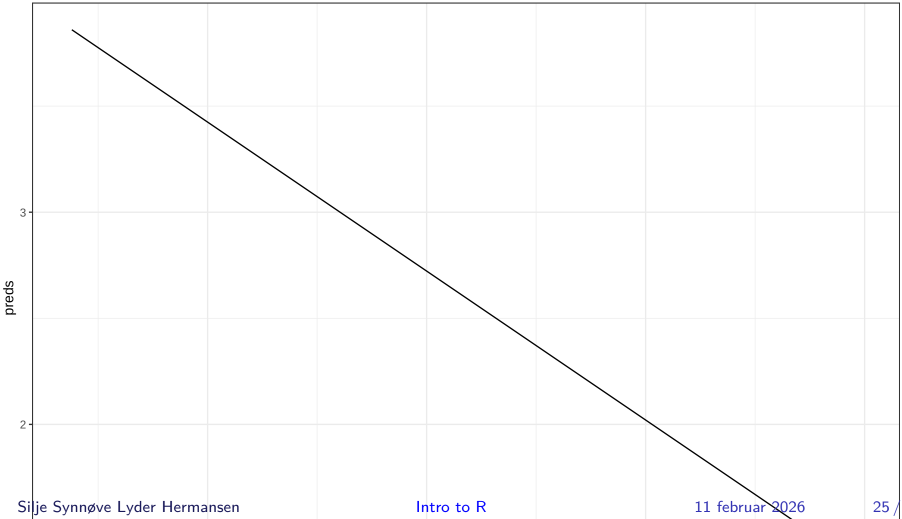
```
##   1   2   3
## 3.9 3.8 3.7
```

- The first difference is 0.95

⇒ *The first difference can be calculated for any two scenarios of your choice!*

# Plot

```
scenario %>%
ggplot +
  geom_line(aes(x = LaborCost,
                y = preds))
```



# Two sources of variation in the data

# Two sources of variation in the data

**But are these effects statistically significant?**

- **Fundamental uncertainty:** The natural randomness in outcomes, even if the true parameters were known (Captured by residual variance).
- **Estimation uncertainty:** How precisely are the coefficients estimated? (Captured by the variance-covariance matrix)

$\Rightarrow$ *the uncertainty of your predictions depend on both*

# Fundamental uncertainty

# Fundamental uncertainty

$$Y_i = \alpha + \beta X1_i + \beta X2_i + \sigma^2$$

**data (observed)**

- variables: X and Y
- observations: i is a counter for the observations, refers to the $i^{th}$ observation. $i...N$

**parameters (estimated)**

- $\alpha$ intercept, the value of Y when X == 0
- $\beta$ slope, the increase in Y when X increases by one unit
- $\sigma^2$ variance in the error term; $\sqrt{\sigma^2}$ = standard deviation

# Let's rewrite

$$Y \sim g(\theta, \sigma^2)$$
$$\theta = \alpha + \beta X_i + \sigma^2$$

- $\theta$: the average value of y
- $g()$: the link function

**The normal model**

$$Y_i \sim N(\mu_i, \sigma^2)$$
$$\mu_i = \alpha + \beta X_i + \sigma^2$$

- $\mu$: mean predicted value
- $N()$: the normal distribution

# What are the residuals?

**We are always wrong in our predictions, but how wrong are we (in-sample)?**

```
df <- df %>% mutate(
  #Predict in sample
  preds = predict(mod2, newdata = .),
  #Calculate the difference between expected and observed
  residuals = LocalAssistants - preds
  )
```

# How to describe the residuals?

**We describe the residuals by their spread (standard deviation/residual standard error)**

```
mean(df$residuals)
```

```
## [1] -9.8e-15
```

- mean: with an unbiased estimator, their average is 0

```
sd(df$residuals)
```

```
## [1] 3.1
```

- standard deviation: but their spread can be more or less high
- here, the average distance from their mean is is a staff size of 3.08 local assistants.

$\Rightarrow$ *residual standard error*

# Where is it reported?

```r
summary(mod2)
```

```
##
## Call:
## lm(formula = LocalAssistants ~ OpenList + LaborCost, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -4.49  -1.94  -0.41   1.08  35.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.1266     0.2861   14.42  < 2e-16 ***
## OpenList      0.8288     0.2278    3.64  0.00029 ***
## LaborCost    -0.0702     0.0102   -6.91    1e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.1 on 736 degrees of freedom
## Multiple R-squared:  0.0814, Adjusted R-squared:  0.0789
## F-statistic: 32.6 on 2 and 736 DF,  p-value: 2.69e-14
```

```r
summary(mod2)$sigma
```

```
## [1] 3.1
```

$\Rightarrow$ *residual standard error is 3.08*

# Conclusion: fundamental error

- important for predictions and model statistics
- not really for the uncertainty of our estimation of our effect

# Estimation uncertainty

# Estimation uncertainty

- most research is about the *effect of x* on y
- so, we're interested in the uncertainty of $\beta$

# The central limit theorem and sampling

**A fiction: the assumptions underpinning the uncertainty of the parameters**

- assumption that data is a sample from a population
- we *could* sample many times
- we calculate the same parameter (e.g. mean, differences in means...) in each sample
- they will vary, but will follow a *normal distribution*

$\Rightarrow$ *each parameter is a distribution with a mean and a standard deviation*

# Standard errors

```
summary(mod2)
```

```
##
## Call:
## lm(formula = LocalAssistants ~ OpenList + LaborCost, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -4.49  -1.94  -0.41   1.08  35.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.1266     0.2861   14.42  < 2e-16 ***
## OpenList      0.8288     0.2278    3.64  0.00029 ***
## LaborCost    -0.0702     0.0102   -6.91    1e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.1 on 736 degrees of freedom
## Multiple R-squared:  0.0814, Adjusted R-squared:  0.0789
## F-statistic: 32.6 on 2 and 736 DF,  p-value: 2.69e-14
```

- mean: average of all the differences in means between the two groups of MEPs: 0.95
- spread: the standard deviation of this distribution is 0.23

$\Rightarrow$ *a standard error is the standard deviation of a hypothetical distribution (parameters)*

# Colinearities

# Colinearities

## Regression parameters may be correlated

```
mat <-vcov(mod2)
mat
```

```
##             (Intercept) OpenList LaborCost
## (Intercept)      0.0819 -0.02846  -0.00244
## OpenList        -0.0285  0.05191   0.00018
## LaborCost       -0.0024  0.00018   0.00010
```

- reported in the *variance-covariance matrix*
- diagonal: the variance of the parameter.
  - variance in effect of electoral system: $\sigma^2 = 0.05$
  - standard error in effect of electoral system: $\sqrt{\sigma^2} = 0.23$
- off-diagonal: the covariance of the parameters
  - low correlation between labor cost and electoral system

# Estimate

**King et al. (2000) make two points**

- find interesting scenarios when you interpret
- estimate the uncertainty for the scenarios including
  - standard error (diagonal)
  - covariance (off-diagonal)

$\Rightarrow$ *the correlation between variables may mean higher or lower uncertainty than only using the standard error*

# Simulation

**They do this using simulation**

- set scenario for all predictors
- draw from the distribution of parameters
- make prediction
- repeat many times
- extract the information and report
    - mean
    - median
    - mode
    - standard deviation
    - plot the distribution!

# Our class

**We will see two ways of doing this in R**

- `ggeffects` package: simulates scenarios for us and can be plotted seamlessly $\rightarrow$ *effect plots, coefplots and point predictions*
- `MASS` package: the "manual" simulation from a multivariate normal distribution using the variance-covariance matrix. $\rightarrow$ *entire vector of simulations; for other plots/purposes*

# Study technique

# For this class

- learn by doing!
  - all readings include R examples; code along!
  - my R notebooks
  - then play around with the concepts; also with your own data/former exams
- dialogue with AI (ChatGPT, Claude)

# What to ask and not to ask chat for?

## R codes

- dont ask for complex codes
  - requires quirey competence on your end
  - you don't learn
- ask it to annotate your scripts
  - explain what each line means
  - dissect all code chunks you find and ask

# What to ask and not to ask chat for?

**Statistics**

- don't ask for a summary of the reading
  - it's not necessarily what we will focus on
  - you don't learn
- ask for definitions
  - ask it to define key concepts you don't understand while you read
  - rephrase definitions and ask it this is a good understanding
- match with your readings
  - upload the PDF and ask specific questions
  - ask for examples, possibly with R codes
- interpretation
  - copy-paste your model output and ask for an explainer
  - use descriptive statistics to find interesting scenarios, ask it to help you find a plain English intuitive sentence