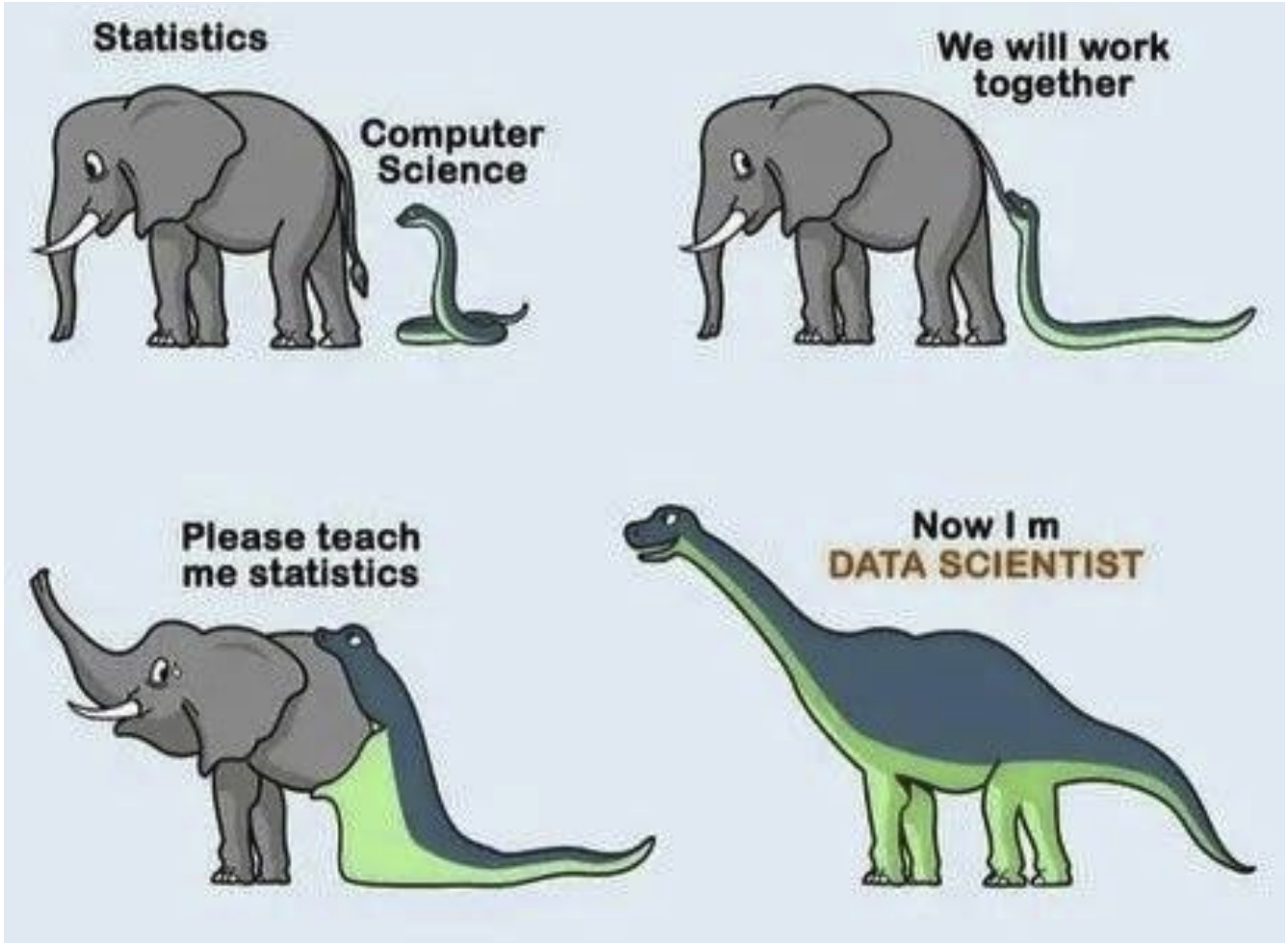
Introduction to Data Science

Intro

Prof. Dr. Ralf Lämmel & **Johannes Härtel**
(johanneshaertel@uni-koblenz.de)


Statistics
Computer Science
We will work together
Please teach me statistics
Now I m
DATA SCIENTIST

What did change, and why do we need Data Science?

Data science

From Wikipedia, the free encyclopedia

Not to be confused with information science.

Data science is an interdisciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from noisy, structured and unstructured data,[1][2] and apply knowledge and actionable insights from data across a broad range of application domains. Data science is related to data mining, machine learning and big data.

Data science is a "concept to unify statistics, data analysis, informatics, and their related methods" in order to "understand and analyze actual phenomena" with data.[3] It uses techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, information science, and domain knowledge. However, data science is different from computer science and information science. Turing Award winner Jim Gray imagined data science as a "fourth paradigm" of science (empirical, theoretical, computational, and now data-driven) and asserted that "everything about science is changing because of the impact of information technology" and the data deluge.[4][5]

Questions and Answers

Questions

- Descriptive
  - How often does X happen?
  - What are the properties of X?

Boring

- Comparative
  - How does X compare to Y?

Boring

- Relational
  - What is the relation between X and Y?

More ML (e.g., classification/clustering)

- **Causal**
  - **Does X cause Y?**

This is serious stuff!

Source: Slides last year (Lecture 3)

Answers

Classical Statistics

Data is produced.

Data is collected.

Experimental

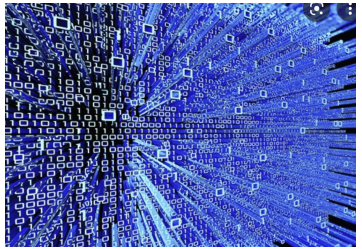
Observational

Data Science

Increasing use of Information Technology & (Digital) Data

1000

2021



*The first methodical approaches to experiments in the modern sense are done by an Arab mathematician and scholar **Ibn al-Haytham** around c. 1000.*

Experiment

Answering Questions the “Experimental Way”

Andrei Varanovich
avaranovich

Andrei was a PhD student in Softlang group, inventing a new software language called “MegaL”.

To prove the relevance of his new language, he conducted an **experiment** to answer: “**What-if** someone uses this new language?”.

BTW, we start with questions related to software, because we are experts on that field.

*The last year’s lectures called this **intrusively**.*

Experiment: Ambitions and Immediate Problems

- Ambitions
  - What-if we use the language, does it improve productivity or quality in software development?
    - How much code do we need to solve a problem?
    - How long does solving a problem take?
    - How often do defects occur when solving a problem?
- Immediate Problems
  - The language has never been used in the wild by real users (until this point in time).
  - Only a research prototype of the language was available.
  - The language was rather exotic: Users need previous instructions to use/understand the language properly.
  - **There is no data.**

Experiment: Teaching the new language to students

Experiment: Splitting the audience at random

Experiment: Executing a task with/without the new language

Experiment: Examine the data

In total, the sample size n=21.

Categorical Feature (Nominal)

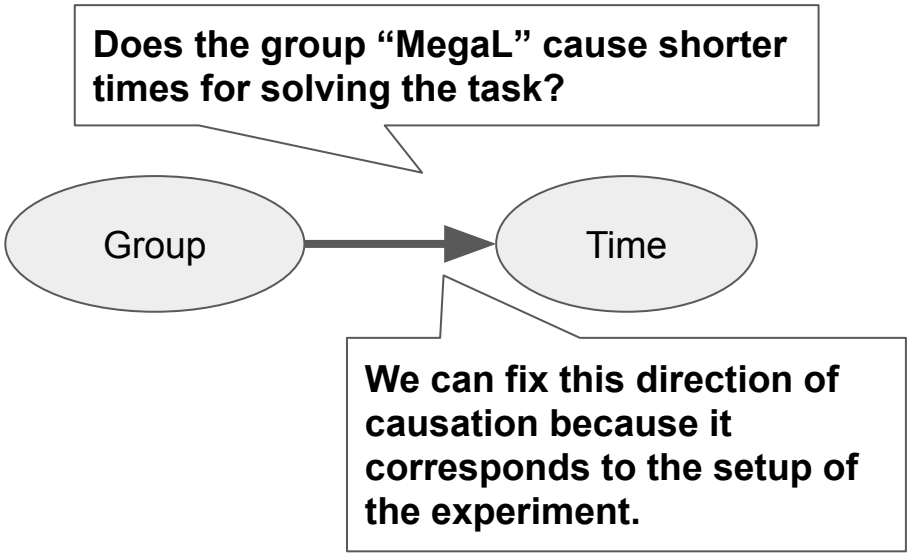
Numerical Feature (Ratio)

	time	group
1	30.1927	MegaL
2	38.4469	
3	41.5104	
4	47.7083	MegaL
5	44.5447	None
6	50.1438	None
7	41.6217	None
8	38.8866	MegaL
9	36.8266	MegaL
10	42.9855	None
11	29.0809	None
12	45.9439	None
13	37.8999	MegaL
	48.0623	None
	37.8966	None
16	45.7603	None
17	37.6444	None
18	47.7172	MegaL
19	43.3382	None
20	38.0811	MegaL
21	43.7954	None

The mean of time for group MegaL is 39.6 and for None it is 42.1
The standard deviation 6.2 and 5.2 respectively.

(caution, simulated data, no real data, everything is made up)

Experiment: Process Model



Experiment: Supporting our intuition by a test

Welch Two Sample t-test

data: time by group

t = -0.94034, df = 10.424, p-value = 0.3683

We can not reject the null hypothesis (if p>0.05).

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

 -8.666981 3.502671

sample estimates:

mean in group MegaL mean in group None

        39.61606   42.19822

(caution, simulated data, no real data, everything is made up)

Experiment: Summary

- Negative:
  - Using students as subject is problematic when generalization to programmers is intended.
  - Using a single task (around 50 min) does not correspond to real software development.
  - This does not scale (e.g., to 10,000 Students).
  - Some things might be hard to manipulate.
  - **We do not examine a phenomenon in its natural context.**
- Positive:
  - We can answer questions on causation (what if using/not-using the language) since we intercept the process of programming by the experiment. **Randomization and the fixed process model protects against confounding variables.**

Simulating the Experiment

Simulating the previous data (in R)

- Understanding which statistical test or model applies to real data is complicated.
- We typically miss a way to test a “statistical test or model”, because we just don’t know the correct output for real input data. Often we assume that we did well if the results match our expectations.
- In such situations, simulating experiments/observational studies (and variations), by producing synthetic data, helps in testing if a method (model or statistic test) works as expected.

Simulating the previous data (in R)

```r
n <- 21 # Sample size (number of students)

# Assign random group to students.
group <- sample(c("MegaL", "None"), n, replace = T)

# Simulate possible values for using MegaL (g1) and not using MegaL (g2).
g1 <- rnorm(n, 41, 5) # 'rnorm' is a random number generator producing a normal distributed variable (with parameters mean and std).
g2 <- rnorm(n, 41, 5)

# Students either use or not use Megal (not both).
time <- ifelse(group == "MegaL", g1, g2)

# Compose a data frame.
data <- data.frame(time,group)
```

We simulated the data of the two groups with the same mean and std. Hence, we expect no difference to appear when applying a test or model to the data.

Observational

Answering Questions the “Observational Way”

I (PhD student, Softlang Koblenz, see picture) am examining **established** programming languages.

To prove the dangers and benefits of certain languages, we argue based on **existing observational data** from, e.g., GitHub, Bitbucket or Maven.

These are just example, they might be replaced by other digital sources of data (e.g., Facebook or Twitter)

The last year’s lectures called this ***non-intrusively.***

Observational: Ambitions and Immediate Problems

- The ambitions are the same as before.
- Immediate Problems:
  - **The data is there but …**
  - Does it make sense to compare languages directly, regardless of task and user?
  - Different languages may have different use groups.
  - Different languages may be used to solve different tasks.
  - …

Observational: Collecting existing data

Observational: Examining the data

Categorical Feature (Nominal)

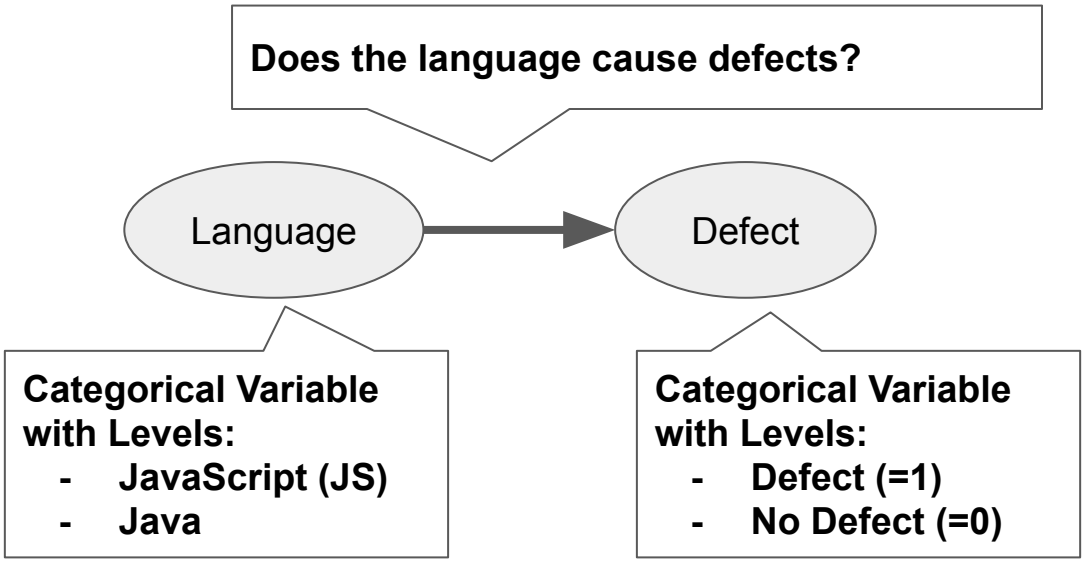
Categorical Feature (Nominal)

These observations might be commits.

	defect	language
	0	Java
2	0	JS
3	0	Java
4	0	Java
5	0	JS
6	0	Java
7	1	Java
8	0	JS
9	1	JS
10	0	Java
11	0	Java
12	0	Java
13	1	Java
14	1	JS
15	0	Java
16	0	JS
17	1	JS
18	0	JS
19	1	Java
20	0	JS
21	0	Java

(caution, simulated data, no real data, everything is made up)

Observational: Process Model



(caution, simulated data, no real data, everything is made up)

Observational: Creating a statistic model (a logistic regression)

Effect on defects

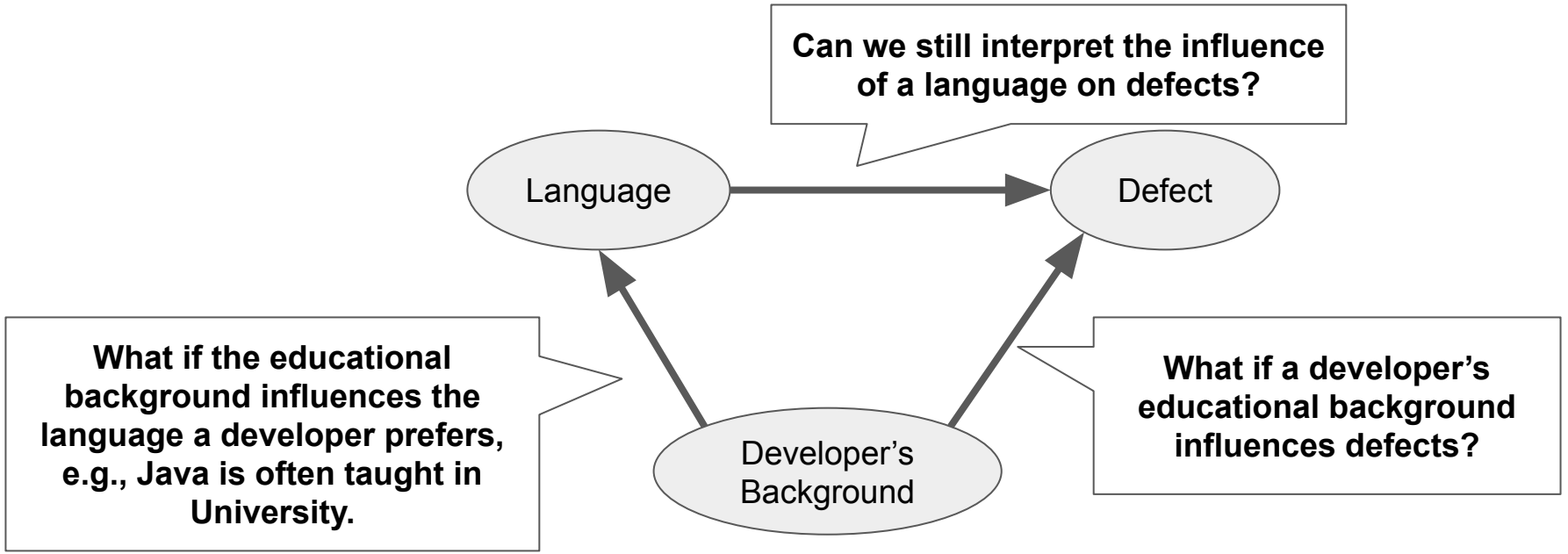
```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -2.8016     0.1766 -15.862  < 2e-16 ***
languageJS    0.6642     0.2200   3.019  0.00253 **
```

Variable if using JS

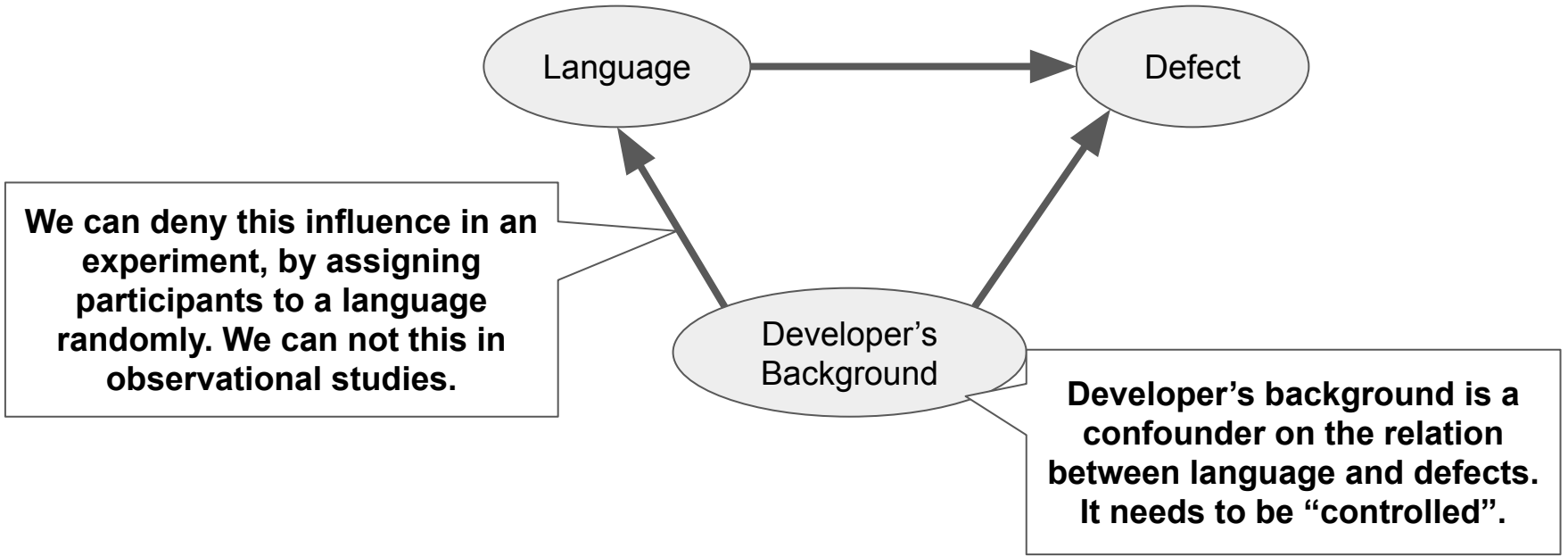
Positive effect of language JS on defects
(stop using JS instantly!!!).

(caution, simulated data, no real data, everything is made up)

Observational: Revising the Process Model



Observational: Revising the Process Model



Observational: Examining the new data

Java and University are quite often going together (there is a correlation, as we expected).

	defect	language	background
1	0	Java	University
2	1	JS	Regular
3	0	Java	University
4	0	Java	University
5	0	JS	Regular
6	0	JS	University
7	0	JS	University
8	0	Java	University
9	0	Java	Regular
10	0	JS	Regular
11	0	Java	University
12	0	Java	University
13	0	Java	University
14	0	JS	University
15	0	Java	University
16	0	Java	Regular
17	0	JS	Regular
18	0	JS	Regular
19	1	JS	Regular
20	0	Java	University
21	0	Java	University

(caution, simulated data, no real data, everything is made up)

Observational: Creating a revised statistic model

Formulating this as a statistical model, with "background" as control variable.

```
Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)          -1.5188     0.2196  -6.916 4.66e-12 ***
languageJS           -0.3767     0.2509  -1.501    0.133
backgroundUniversity -2.2176     0.3361  -6.598 4.17e-11 ***
```

We see university is associated with fewer defects. JS turns out to be the better language.

(caution, simulated data, no real data, everything is made up)

Observational: Summary

- Negative:
  - **It is harder to answer questions on causation** (what if using/not-using the language), since we cannot intercept the process, and we do not always know the process in detail.
  - It is hard to credibly deny confounding variables.
- Positive:
  - **This does scale.** We can use it to study huge digital phenomenons.
  - **We can examine a phenomenon in its natural context.**

Simulating the Observational Study

Simulating the previous data (in R)

```r
n <- 1210 # Sample size.

# Random university or regular developers.
background <- sample(c("University", "Regular"), n, replace = T)

# Produce language used (University developers tend to use Java).
university <- sample(c("Java", "JS"), n, replace = T, prob = c(0.8, 0.2))
regular <- sample(c("Java", "JS"), n, replace = T, prob = c(0.2, 0.8))

language <- ifelse(background == "University", university, regular)

# Produce the probability of defects by adding up different constants depending on background and language.
prob <- inv logit(-2 +
           ifelse(language == "Java", 0.3, 0.0) + # Java causes more defect than JS.
           ifelse(background == "University", -2.5, 0.0)) # University developer prevent defects due education.

Simulate possible defects by a binomial distribution with 1 trial taking prob.
defect <- rbinom(n, 1, prob)

# Compose data frame.
data <- data.frame(defect, language, background)
```

Simulating the previous data (in R)

```r
# Inverted logistic function needed to produce defects.
inv_logit <- function(x) {
 p <- 1 / (1 + exp(-x))
 p <- ifelse(x == Inf, 1, p)
 p
}
```

A library function that is needed for the previous code.

Other examples

Experiment

BACKGROUND

Although several therapeutic agents have been evaluated for the treatment of coronavirus disease 2019 (Covid-19), no antiviral agents have yet been shown to be efficacious.

METHODS

We conducted a double-blind, randomized, placebo-controlled trial of intravenous remdesivir in adults who were hospitalized with Covid-19 and had evidence of lower respiratory tract infection. Patients were randomly assigned to receive either remdesivir (200 mg loading dose on day 1, followed by 100 mg daily for up to 9 additional days) or placebo for up to 10 days. The primary outcome was the time to recovery, defined by either discharge from the hospital or hospitalization for infection-control purposes only.

Source: *Beigel, John H., et al. "Remdesivir for the treatment of Covid-19—preliminary report." New England Journal of Medicine (2020).*

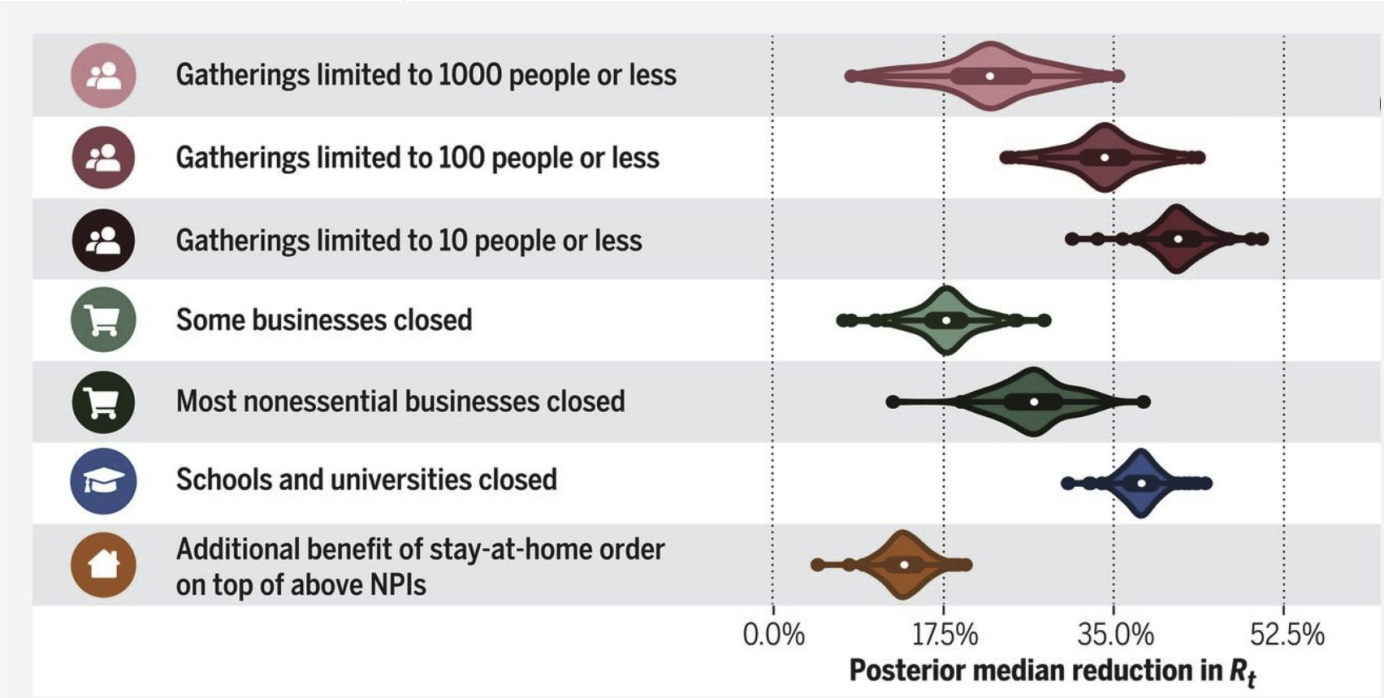
Experiment

RESULTS

A total of 1062 patients underwent randomization (with 541 assigned to remdesivir and 521 to placebo). Those who received remdesivir had a median recovery time of 10 days (95% confidence interval [CI], 9 to 11), as compared with 15 days (95% CI, 13 to 18) among those who received placebo (rate ratio for recovery, 1.29; 95% CI, 1.12 to 1.49; P<0.001, by a log-rank test). In an analysis that used a proportional-odds model with an

Source: *Beigel, John H., et al. "Remdesivir for the treatment of Covid-19—preliminary report." New England Journal of Medicine (2020).*

Observational Study



Source: *Brauner, Jan M., et al. "Inferring the effectiveness of government interventions against COVID-19." Science 371.6531 (2021).*

Summary

- Difference between experimental and observational studies
- Hypothesis tests
- Statistic models and exploring alternative models
- Process models
- Confounding variables

We will meet all these topics again in the remainder of this course.