Please scan this code to register your presence on Ilias
for contact-tracing purposes. Your Ilias identity will be used.
– Only do this if you are *physically present*! –

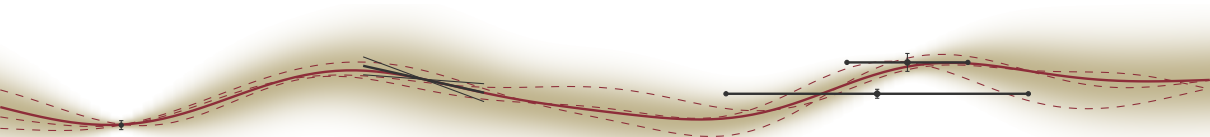
Data Literacy
Lecture 05
Testing Hypotheses

Philipp Hennig

1 December 2020

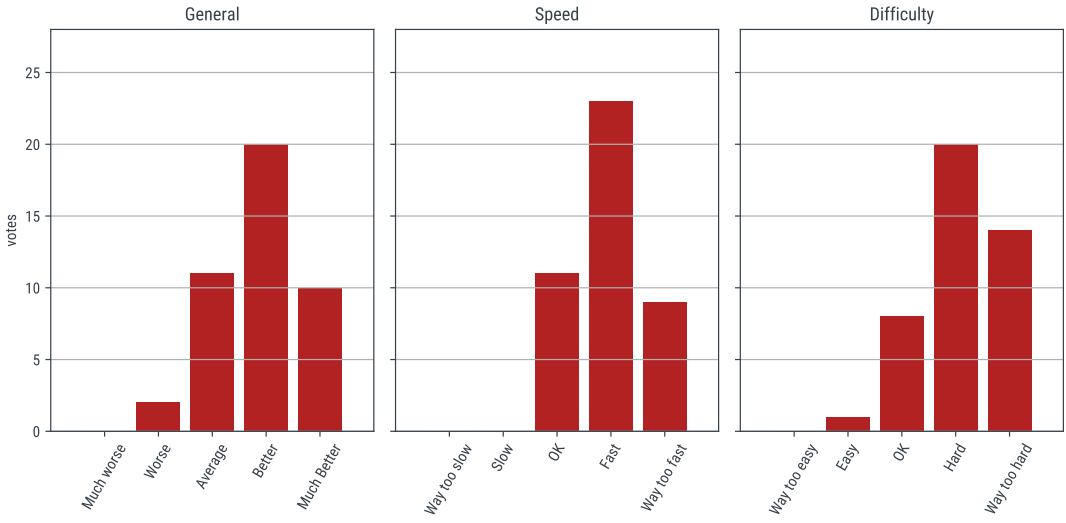
Eberhard Karls
Universität
Tübingen

Faculty of Science
Department of Computer Science
Chair for the Methods of Machine Learning



Feedback

quantitative



Detailed Feedback

your answers

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Things you *didn't* like

- Please use examples with numbers (or a use case), not abstract variables!
- Statistics that are presumed to be known. (e.g. what is the Hessian?) It's quite hard to follow without a statistics background. Maybe provide materials for catching up on statistics? That would be great.
- Laplace approximation
- Math statistics

Things you liked

- Examples! (5x)
- Code! (8x)
- How you adapted the speed of the lecture after the last feedback and took the time to explain some parts again or in more detail. Good intuition for what might be hard to understand.
- Connection Bayesian and frequentist approach
- Fisher information - props to you for teaching this wonderful concept

Things you didn't understand

- The python code
- Everything pertaining to the math.
- How the score is different from MLE or how it relates to one another.
- The mode of the likelihood is the mode of the posterior

Please register here if you want to participate in the course evaluation.

From Estimate to Hypothesis

Turning a number into a decision

Consider data $\boldsymbol{x} = [x_1, \ldots, x_n]$ drawn *identically and independently distributed (iid)* from $p(\boldsymbol{x} \mid \theta)$.

- *Estimates* $\hat{\theta}$ reduce the likelihood to a single number
- *Confidences* C_α extend this to regions of high values under the likelihood (actually: to regions confining a high mass under the posterior)
- a *Hypothesis* is a statement about a binary variable. For example:
  - "The vaccines work."
  - "Human emissions are affecting the climate."

Even though a hypothesis is either true or false, with finite data, we usually do not know *with infinite certainty*. So we will assign a *probability* to its truth value. Statistical **testing** is the process of turning likelihoods over continuous variables into binary hypotheses, thereby computing this probability.

Why is it hard to test hypotheses?

Simple Example

`Gaussian Tests.ipynb`

Key problem: There are often many (even infinitely many) ways to "explain" an observation. If we pick one specific one, it's nearly always wrong.

Idea: Life is asymmetric. In science, it is often possible to define the "expected" / "boring" explanation. **Hypothesis testing** starts with a default theory, the **null hypothesis** and asks whether the data "provide sufficient evidence to *reject* the null hypothesis". That is, whether the data is so improbable to have been generated from the hypothesis that we have to find a better explanation.

General process: Consider data $\boldsymbol{x} = [x_1, \ldots, x_n]$ drawn from $p(\boldsymbol{x} \mid \theta)$. As the null hypothesis, we assume that the true value θ comes from some domain of possible explanations, $\theta \in H_0$. Assign a prior probability measure $p(\theta \mid H_0)$ over H_0 (a uniform one is not always possible or a good idea). Compute the *evidence*

$$p(\boldsymbol{x} \mid H_0) = \int p(\boldsymbol{x} \mid \theta) p(\theta) \, d\theta.$$

Reject the null if the evidence is "very low" (we have to define this further).

Why is this even a problem?

Evidences are likelihoods, not posteriors

$$p(\boldsymbol{x} \mid H_0) = \int p(\boldsymbol{x} \mid \theta, H_0) p(\theta \mid H_0)\, d\theta.$$

- The evidence is just a likelihood $p(x \mid H_0) \in [0, 1]$. It can be arbitrarily low.
- Ideally, we would like a (posterior) *probability* $p(H_0 \mid \boldsymbol{x})$. This requires a prior over *all* hypotheses

$$p(H_0 \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid H_0) p(H_0)}{\sum_i p(\boldsymbol{x} \mid H_i) p(H_i)}$$

- But we may worry that we can not enumerate all hypotheses (nor assign priors to them).

"Solution:"

- Just consider the null hypothesis and see if it can be rejected, without suggesting an alternate hypothesis
- Increase the set of events under consideration to include "the event we observed and all 'more extreme' events" (to get sets of finite measure)

Why is this even a problem?

Evidences are likelihoods, not posteriors

"Solution"

- Just consider the null hypothesis and see if it can be rejected, without suggesting an alternate hypothesis
- Increase the set of events under consideration to include "the event we observed and all 'more extreme' events" (to get sets of finite measure)

This causes two problems/challenges/dangers:

- Need to define "more extreme"
- Rejecting the null does not imply accepting the one. This is extremely difficult to communicate.
- (Rejecting the null does not even mean the null is wrong! It's just unlikely!)

- Testing is a philosophically fraught concept: Because we are worried we can not list *all* hypotheses (to do Bayesian inference), instead we find just one (the null), show that it is likely wrong, then use this as an argument in favor of another (arbitrary) hypothesis.
- There is really only one good use for tests: For scientists to decide whether observations are "sufficiently surprising" ("significant") to warrant further study. If a hypothesis is unlikely (the test rejects the hypothesis), this just means "our explanation for the data is probably wrong". But it does *not* tell us what the *correct* explanation may be!

Formal Treatment

Definition: Consider a random variable $X \in \mathbb{X}$ drawn iid. from $p(x \mid \theta)$ and a **rejection region** $R \subset \mathbb{X}$. A **statistical test** for the null hypothesis $H_0 : \theta \in \Theta_0$ is a decision rule that **rejects** the hypothesis if $X \in R$ and does not reject (retains) the hypothesis if $X \notin R$. This usually involves a **test statistic** $T : \mathbb{X} \to \mathbb{R}$ and a **critical value** $c \in \mathbb{R}$, then setting

$$R = \{x : \tau(x) > c\}.$$

The **power function** is $\beta(\theta) = \mathbb{P}_{p(x|\theta \in \Theta_0)}(x \in R)$,

and the **size** of the test is $\alpha = \sup_{\theta \in \Theta_0} \beta(\theta)$.

Suppose that for every $\alpha \in (0, 1)$ we have a size α test with rejection region R_α. Then the p-value is the smallest level at which we decide to reject H_0:

$$p\text{-value} = \inf\{\alpha : T(X^n) \in R_\alpha\}$$

Intuitively: A small p-value means that even a test with very small size (with very small rejection probability, with very low power) would reject this hypothesis. Thus, the hypothesis must be very bad.

Example:

Beta-binomial tests

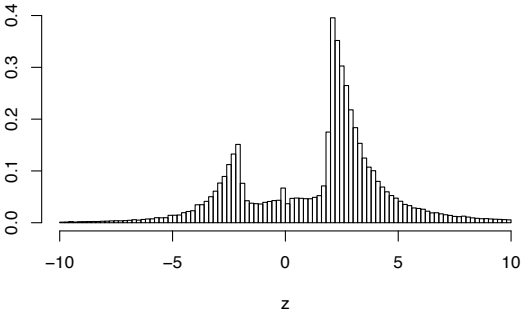
`BioNTech_BetaBinomial.ipynb`

Discussion

informal definition of *p*-values

source: Zwet & Cator 22/09/2020 https://arxiv.org/abs/2009.09440

Informally: The *p*-value describes how probable observations equal or "more extreme than" X^n are under the null hypothesis.
In many scientific communities, values $p < 0.05$ are required to describe a result as "significant". This means that if the null is actually correct, we would "accidentally reject" it about every 20th time.



Distribution of more than one million *z*-scores from Medline (1976–2019)
($|z| > 2$ implies $p < 0.05$)

Multiple Testing – Hunting for Significance

An example

Data from August 2020 (recent data in your homework)

```
p = scipy.stats.betabinom.cdf(Won_2020, Matches_2020, Won + 1, Matches - Won + 1)
```

TeamName	Matches	Won	Draw	Lost	Matches 2020	Won 2020	Lost 2020	Draw 2020	p
1. FC Köln	238	67	63	108	7	0	4	3	0.100445
1. FC Union Berlin	34	12	5	17	7	3	1	3	0.763838
1. FSV Mainz 05	340	118	81	141	7	0	6	1	0.051756
Bayer Leverkusen	340	171	73	96	7	4	0	3	0.766256
Borussia Dortmund	340	203	72	65	7	5	2	0	0.843269
Borussia Mönchengladbach	340	152	75	113	7	3	2	2	0.612902
Eintracht Frankfurt	306	105	75	126	7	2	1	4	0.545959
FC Augsburg	306	91	84	131	7	3	3	1	0.873150
FC Bayern	340	255	47	38	7	6	1	0	0.865623
FC Schalke 04	340	140	81	119	7	0	4	3	**0.025269**
Hertha BSC	272	88	69	115	7	2	4	1	0.590058
RB Leipzig	136	72	36	28	7	5	1	1	0.910808
SC Freiburg	306	96	85	125	7	1	3	3	0.301545
TSG 1899 Hoffenheim	340	120	100	120	7	2	4	1	0.523972
VfB Stuttgart	272	87	57	128	7	2	1	4	0.598366
VfL Wolfsburg	340	127	92	121	7	2	0	5	0.477795
Werder Bremen	340	105	93	142	7	2	1	4	0.623958

The Bonferroni Correction

A conservative bound on multiple tests

Definition (Bonferroni Correction)

Given m p-values $p_1, \ldots, p_m$, reject the null hypothesis H_{oi} if

$$p_i \leq \frac{\alpha}{m}$$

Carlo E. Bonferroni
1892–1960

Nomenclature

characterising tests

source: https://en.wikipedia.org/wiki/Receiver_operating_characteristic

Consider a test for the null hypothesis H_0 and the alternative hypothesis H_1

type-I error ("false positive") occurs if the test **rejects** H_0 even though it is true

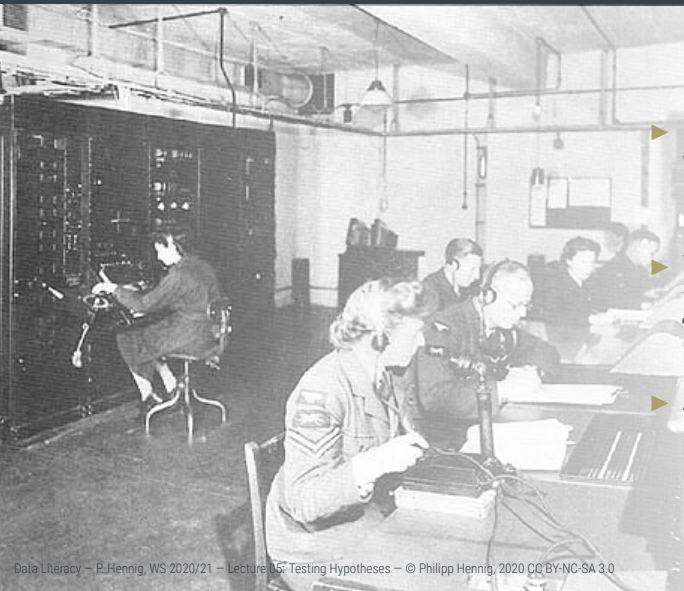
type-II error ("false negative") occurs if the test **does not reject** H_0 even though it is false

		True condition			
	Total population	Condition positive	Condition negative	Prevalence = $\frac{\sum \text{Condition positive}}{\sum \text{Total population}}$	Accuracy (ACC) = $\frac{\sum \text{True positive} + \sum \text{True negative}}{\sum \text{Total population}}$
Predicted condition	Predicted condition positive	**True positive**	**False positive**, Type I error	Positive predictive value (PPV), Precision = $\frac{\sum \text{True positive}}{\sum \text{Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\sum \text{False positive}}{\sum \text{Predicted condition positive}}$
	Predicted condition negative	**False negative**, Type II error	**True negative**	False omission rate (FOR) = $\frac{\sum \text{False negative}}{\sum \text{Predicted condition negative}}$	Negative predictive value (NPV) = $\frac{\sum \text{True negative}}{\sum \text{Predicted condition negative}}$
	True positive rate (TPR), Recall, Sensitivity, probability of detection, Power = $\frac{\sum \text{True positive}}{\sum \text{Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm = $\frac{\sum \text{False positive}}{\sum \text{Condition negative}}$	Positive likelihood ratio (LR+) = $\frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) = $\frac{\text{LR+}}{\text{LR−}}$	F_1 score = $2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$
	False negative rate (FNR), Miss rate = $\frac{\sum \text{False negative}}{\sum \text{Condition positive}}$	Specificity (SPC), Selectivity, True negative rate (TNR) = $\frac{\sum \text{True negative}}{\sum \text{Condition negative}}$	Negative likelihood ratio (LR−) = $\frac{\text{FNR}}{\text{TNR}}$		

More about tests and all these characteristics in *Math for Machine Learning* (Prof. Pons-Moll)

Designing tests is tricky

A/B Testing – how many experiments do you need?



Ventnor Chain Home Radar Station, Isle of Wight, 1941

- Say you're working for a company that runs a website. Currently, 75% of all customers landing on the page leave without clicking anywhere.
- You've done a re-design and would like to A/B test the page. Your boss cautiously asks: "How many customers should we trial the page on?"
- Actually, we can afford to show it to 100 people. What kind of improvement would you be able to detect?

DEMO

`BioNTech_BetaBiomial.ipynb`

Summary

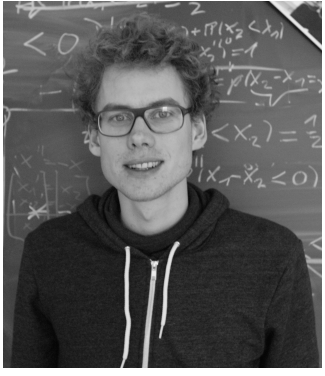
- tests measure whether a *null* hypothesis is likely false
- they help ensure we do not surprised by something that isn't surprising
- but rejecting the null does not imply the one is true
- if we run *multiple* tests, we have to account by decreasing the test's power
- analysing tests can help design experiments

Please provide feedback:

Methods of Machine Learning Research Seminar
Wednesday, 24 November, 15:00 (st)
Dr. Hans Kersting (ENS/INRIA)

Forward and Inverse ODE Filters

The field of probabilistic numerics recasts numerical approximations as statistical inference, by interpreting the solution to a numerical problem as a parameter in a probabilistic model. For the numerical problem of solving ODEs, one of the main probabilistic numerical methods are ODE filters and smoothers. These methods utilize linear-time algorithms from signal processing (Kalman filters etc.) to solve ODEs. In this talk, we will introduce these methods and summarize some recent developments in this field. Also, we will describe how ODE filters can speed up inverse problem solvers.



details at https://talks.tue.ai/talks/talk/id=15