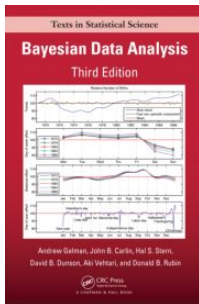


Bayesian data analysis (Aalto fall 2022)

- ▶ Book: Gelman, Carlin, Stern, Dunson, Vehtari & Rubin: Bayesian Data Analysis, Third Edition. (online pdf available)
- ▶ The course website has more detailed information than these slides

https://avehtari.github.io/BDA_course_Aalto/Aalto2022.html

- ▶ Timetable: see the course website
- ▶ TAs: Anna Riha, Elena Shaw, Kunal Ghosh, Andrew Johnson, Noa Kallioinen, David Kohns, Leevi Lindgren, Yann McLatchie, Teemu Sailyoja, Niko Siccha



Bayesian data analysis

Pre-requisites

- ▶ Basic terms of probability theory
 - ▶ probability, probability density, distribution
 - ▶ sum, product rule, and Bayes' rule
 - ▶ expectation, mean, variance, median
- ▶ Some algebra and calculus
- ▶ Basic visualisation techniques (R or Python)
 - ▶ histogram, density plot, scatter plot

These will be tested with the first assignment round

Bayesian data analysis

Pre-requisites

- ▶ What to do if the course seems to be too difficult
 - ▶ refresh your memory on pre-requisites (see the course web site for some links)
 - ▶ ask for help
 - ▶ consider reading Regression and Other Stories
<https://avehtari.github.io/ROS-Examples/>
 - ▶ consider reading Statistical rethinking + watching videos
<https://xcelab.net/rm/statistical-rethinking/>

Bayesian data analysis

Different learning styles

- ▶ Reading
- ▶ Listening lectures
- ▶ Solving problems
 - ▶ mathematical derivations
 - ▶ programming

Bayesian data analysis

Assessment

- ▶ Exercises 2/3, and project work and presentation 1/3
 - ▶ Minimum of 50% of points must be obtained from both the project work and the exercises.

Bayesian data analysis

- ▶ Lectures describe basics and give broader overview (recorded and made available)
 - ▶ written material has all the details and self-study is possible
- ▶ Supporting material and assignments in https://avehtari.github.io/BDA_course_Aalto/Aalto2022.html
 - ▶ reading instructions and chapter notes
 - ▶ demos (very useful for assignments)
 - ▶ slides (not very useful without the lectures)
 - ▶ video clips
 - ▶ links to additional material
- ▶ R demos https://avehtari.github.io/BDA_course_Aalto/demos.html#BDA_R_demos
- ▶ (Python demos https://avehtari.github.io/BDA_course_Aalto/demos.html#BDA_Python_demos)
- ▶ Aalto Zulip chat instance (link in MyCourses)

Bayesian data analysis

Assignments

- ▶ Weekly assignments (some have two weeks time)
 - ▶ R (Python) simulation exercises
 - ▶ Stan probabilistic programming exercises (via R (Python))
- ▶ Related R (Python) demos available (see the course web site)
- ▶ TAs available: see Oodi for exercise sessions
- ▶ Exercise deadlines on Sunday (see detailed info in the course web page)
 - ▶ we recommend to submit before Friday 3pm as TAs are not available during the weekend
 - ▶ we allow the late submission on Sunday as some students are working on weekdays
- ▶ After the exercise deadline, the grading period Monday–Tuesday
- ▶ Students grade 3 other exercises using peergrade.io

Bayesian data analysis

R vs Python

- ▶ We strongly recommend using R in the course as there are more packages for Stan and statistical analysis in general in R
- ▶ If you are already fluent in Python, but not in R, then using Python may be easier, but it can still be more useful to learn also R

Bayesian data analysis

Assignments

- ▶ Assignments are given on PeerGrade (also available in the course website)
- ▶ Assignments are returned and graded on Peergrade

Assignments

peergrade.io

- ▶ Peergrading used in BDA course since 2016
- ▶ Each student grades 3 exercises (randomly distributed)
- ▶ Detailed grading instructions – rubric (available also on the course website)
- ▶ Also text feedback
- ▶ Possible to flag inappropriate grading (please, be polite!)
- ▶ TAs check flagged gradings
- ▶ Possible to give thumb up for great feedback
 - ▶ those who give good feedback will get bonus points
- ▶ See more at https://avehtari.github.io/BDA_course_Aalto/assignments.html

Assignments

peergrade.io

- ▶ Combined score: 75% submission performance, 25% feedback performance

Assignments

peergrade.io

- ▶ Combined score: 75% submission performance, 25% feedback performance
- ▶ Hand-in score:
 - ▶ averaging the scores from peers
 - ▶ after flagging, teacher may overrule the score
 - ▶ different exercises have different weights

See details at

<http://help.peergrade.io/interfaces-and-features/grading-and-scores/the-hand-in-score>

Assignments

peergrade.io

- ▶ Combined score: 75% submission performance, 25% feedback performance
- ▶ Hand-in score:
 - ▶ averaging the scores from peers
 - ▶ after flagging, teacher may overrule the score
 - ▶ different exercises have different weights

See details at

<http://help.peergrade.io/interfaces-and-features/grading-and-scores/the-hand-in-score>

- ▶ Feedback score:
 - ▶ When students receive a review, they are asked to react to it using a scale ranging from “Not useful at all” to “Extremely useful”.
 - ▶ These ratings each correspond to a score between 0% and 100%.
 - ▶ The feedback score is the average of the reaction scores.
 - ▶ “Somewhat useful. Could be more elaborate.” is the baseline reaction.

Peergrade.io

Registration

- ▶ Go to BDA MyCourses page
- ▶ Click Peergrade and login with Aalto account

Assignments

Plagiarism and empty reports

- ▶ It's ok to discuss assignments with others
- ▶ It's ok to use code from the demos (good to mention the source)
- ▶ Don't copy reports from others or from internet
- ▶ Don't submit empty, almost empty or nonsense report
 - ▶ these will be problematic for other students
 - ▶ if you see such, you can mark it as problematic and get another one for grading

Project work

- ▶ Project work in groups of 1–3
 - ▶ combines all the pieces learned in one project work
 - ▶ R or Python notebook report
 - ▶ project report peer graded
 - ▶ oral presentation graded by me and TAs
- ▶ More about projects later

Zulip chat

bda2022.zulip.cs.aalto.fi

RStudio, Quarto, R markdown

- ▶ RStudio is a great IDE for R
- ▶ Quarto is a new markdown language for making reports mixing text, code, equations, tables, etc
 - ▶ *Quarto is the next iteration of R Markdown, and allows you can create dynamic content with Python, R, Julia, and Observable, author documents as plain text markdown or Jupyter notebooks, and output to multiple format types.*
- ▶ RStudio has also visual editor for Quarto (and R markdown) making it easy for new users
- ▶ RStudio is also installed in Aalto JupyterHub