Handouts

EL1: Intro

Important aspects

- Data (where does is come from, what does it contain)
- Ethics and legal (how to handle sensitive data, what laws and regulations apply)
- Project management (how to plan and execute a data project, version control, reproducibility)

Structure

- Lectures on different data sources
- Exercises on data management and analysis
  - R with some additional tools (Git, GitHub, targets, data.table)
- A data project with written report and presentation
- Final exam
- Instruction web page (in addition to Canvas: https://sta220.github.io/documentation/)
- Litterature: accassible through GU library (O'Reilly Learning for Higher Education) or otherwise shared (no need to purchase books)

Different types of data

- Images
- Lab samples
- Unstructured medical records
- Sensor data
- EHRs (electronic health records)
- **Structured medical records (tabular data)**

Usages

- Research
- Quality control/improvement
- Administration/reporting
- News coverage
- Building prediction models and tools

Register data

- Administrative registers
- Health care registers
- Quality registers
- Administrative data
  - Billing codes

  - Claims data
  - Registries
  - Demographic data
- Clinical data
  - health care registers
  - quality registers
    - conditions (diabetes, cancer, etc)
    - procedures (total hip arthroplasty)
- Individual background data
  - socioeconomic data
  - education
  - income
  - occupation
  - family relations
  - migration status
  - Mortality data
    - date of death
    - cause of death
  - Hospital background data
    - hospital characteristics
    - staffing
    - resources
    - geographical area
    - level of specialization
    - private, public
- aggregated data
  - population data
  - neighborhood characteristics
  - pollution
  - crime rates

Inclusion criteria

- Defines the study population
- Defines the exclusion criteria
- Simple example: "Everyone who had a total hip arthroplasty in Sweden"
  - Includes: all ages, all hospitals, all reasons for the prothesis, all types of prosthesis
- Complicated example: Ovarian cancer

Coverage and completeness

- **Institutional coverage**: proportion of all eligible units/clinics that are connected to the registry

  - e.g., 90% of hospitals performing the procedure are connected
  - Should be known by the “register holder”
- **Case coverage**: proportion of patients who should have been reported from connected units that are actually included
  - e.g., 85% of eligible patients registered
  - The aim is to use 100 % but this is not always possible
- **Data completeness**: proportion of required data fields that are filled in for the registered patients
  - e.g., 95% of patients have smoking status recorded
  - e.g., 80% of patients have blood pressure data available

What is recorded?

- Some registers are mandated by law and regulations
- Quality registers often have a steering committee and register holder
- Reseasrh initiated databases according to specific protocols

Data linking

- Unique personal identifier
  - Not in every country!
- study specific id number
- HSA

Unique personal identifier

(Swedish: personnummer, reading: [1])

121212-1212 Tolvan Tolvansson

- 10 (or 12) digits
- date of birth-4 digits
- assigned at birth or immigration
- used in all health care contacts
- used for all administrative data
- sometimes reused after death
- sometimes changed (uncommon)
- sometimes inclusion criteria for register
- similair in the Nordic countries
  - Denmark: CPR number
  - Norway: Fødselsnummer
  - Finland: Henkilötunnus
  - Iceland: Kennitala

Combining data

- Similar registries in different areas/regions/countries

  - Different individuals but similar data
- Same definitions and variables?
- Same inclusion criteria?
- Don’t get fooled by similar names!
- Differences and similarities within the Nordic countries [2]

Working with health care data

A lot to do before the statistical analysis!

- Data management
  - large datasets
  - multiple datasets
  - different formats
  - missing data
  - data cleaning
  - data transformation
  - data wrangling
  - data munging
  - data governance
  - data engineering

R as a tool but …

- Large files often comes exported from SAS
- csv or text files
- API calls
- SQL databases

Our use of R

- data.table to handle large data sets efficiently
- targets to streamline a reproducible pipeline
- Git for version control
- GitHub for colaboration
- Quarto for reporting

Reading list

- Chapter 1 from [3]

EL2: Legal

Swedish law

- Fundamental laws (parliament but difficult to change)
- Ordinary laws (parliament)
  - New ones all the time
  - New laws to update existing regulations

- Published in Svensk författningsamling, SFS
  - The “big blue book” is only a smaller collection of important laws
- ordinances (“förordning”) from the goverment to implement laws
- regulations (“föreskrifter”) from authorities to implement laws and ordinances

Two main branches of law

- Civil and criminal law:
  - state what you can not do (everything else is “legal”)
  - Handled by ordinary courts
  - Ex: Brottsbalken kap 20 om tjänstefel m.m. (The Swedish Penal Code (Brottsbalken), Chapter 20 — Offences Relating to Public Office.)
- Public law: relationship between individuals and the state etc
  - state what the public authorities must do (everything else is “illegal”)
  - Handled by administrative courts
  - What we mostly care of here

Tryckfrihetsförordningen (TF)

- World’s oldest freedom of the press law (since 1766)
- Chapter 2: Public access to official documents
- Applies to public authorities and institutions
  - Including health care registers and medical records held by public authorities
- Everyone has the right to access official documents (TF 2.1)
- but there are exceptions (TF 2.2)
  - e.g., if disclosure would violate privacy or national security
  - if so, the goverment has the right to provide ordinary laws that restrict access (which they do …)

Offentlighets- och sekretesslagen (OSL)

- Law that regulates public access to official documents and secrecy
- Applies to public authorities and institutions
- Defines what information is considered secret and under what circumstances

Chap 21

Secrecy for private individuals’ personal circumstances **no mather the context**

- E.g., health data, economic circumstances, family relations

> OFS 21.1: Secrecy applies to information concerning an individual’s health or sexual life, such as information about illnesses, substance abuse, sexual orientation, gender reassignment, sexual offences, or other similar information, **if it can be assumed** that disclosure of the information would **cause significant harm** to the individual or to someone closely related to them.

Chap 24

Secrecy for the protection of individuals in research and statistics.

- Special research databases etc
- Some regulations for research ethics boards

Chapter 25

Secrecy for the protection of individuals in activities relating to **health and medical care** etc.

> OFS 25.1: Within the health and medical care services, secrecy applies to information concerning an individual's state of health or other personal circumstances, **unless it is clear** that the information may be disclosed **without causing harm** to the individual or to someone closely related to them. The same applies to other medical activities, such as forensic medical and forensic psychiatric examinations, insemination, in vitro fertilisation, abortion, sterilisation, circumcision, and measures to prevent communicable diseases.

- Exceptions exists,
  - for example to submit medical patient data to quality registers
  - to share data between public organisations for research purposes or statistics (OFS 25.11 p. 5).

Chapter 10

Provisions on disclosure overriding secrecy and provisions on exemptions from secrecy

> OFS 10.28: Secrecy does not prevent information from being disclosed to another authority where a duty to provide information follows from an act or an ordinance.

- This would apply to data sharing for research purposes when there is a legal basis for that

Patientdatalagen (PDL)

The Patient Data Act (PDL)

- regulates the processing of personal data within **health and medical care** in Sweden.
- Applies to **healthcare providers** (public and private).
- Main objectives:
  - Protect patient privacy
  - Ensure safe and effective healthcare
  - Enable **secondary use** of health data under strict conditions

Chapter 7 PDL

National and regionel quality registers

Opt-out for patients (every one is included by default until they opt out)

> PDL 7.4: Personal data in national and regional quality registers may be processed for the purpose of systematically and continuously developing and ensuring the quality of healthcare.

PDL 7.5: Personal data processed for the purposes set out in Section 4 may also be processed for the purposes of

- the production of **statistics**,
- estimating numbers for the planning of clinical research,
- **research within health and medical care**,
- disclosure to a party that will use the data for purposes referred to in Sections 1 and 3 or in Section 4, and
- …

Lag om hälsodataregister (SFS 1998:543)

This law regulates health data registers outside the health and medical care system. A new law is being proposed to replace this one.

§ 3: Personal data in a health data register may be processed for for the following purposes:

- the production of statistics,
- follow-up, evaluation and quality assurance of health and medical care, and
- research and epidemiological studies

Specific registers

Register (Swedish)	Register (English)	Governing act / ordinance
Folkbokföringen	Population Register	Population Registration Act (1991:481); Population Registration Ordinance (1991:749)
Totalbefolkningsregistret (RTB)	Total Population Register	Official Statistics Act (2001:99); Official Statistics Ordinance (2001:100)
Nationella patientregistret	National Patient Register	Health Data Act (1998:543); Ordinance on the National Patient Register (2001:707)
Cancerregistret	Swedish Cancer Register	Health Data Act (1998:543); Cancer Register Ordinance (2001:709)
Dödsorsaksregistret	Cause of Death Register	Health Data Act (1998:543); Cause of Death Register Ordinance (2001:709)
Läkemedelsregistret	Prescribed Drug Register	Act on the Prescribed Drug Register (2005:258); Ordinance (2005:363)
Medicinska födelseregistret	Medical Birth Register	Health Data Act (1998:543); Medical Birth Register Ordinance (2001:708)

Register (Swedish)	Register (English)	Governing act / ordinance
Tandhälsoregistret	Dental Health Register	Health Data Act (1998:543); Dental Health Register Ordinance (2008:194)

Arkivdatalagen (ADL)

- Regulates the management of public records and archives
- Applies to public authorities and institutions
- Differnt authorities then have different rules for how long data must be kept
    - For example healthcare data is often required to be kept for at least 10 years

GDPR

- Regulation (EU) 2016/679 (GDPR)
- Enforced since May 25, 2018
- Applies to all EU member states
- Regulates the processing of personal data
- Aims to protect the privacy and rights of individuals
- Sets out rules for data controllers and processors

Special categories of personal data

1. personal data revealing racial or ethnic origin,
2. political opinions,
3. religious or philosophical beliefs,
4. trade union membership,
5. genetic data,
6. biometric data for the purpose of uniquely identifying a natural person,
7. **data concerning health**,
8. data concerning a natural person's sex life or sexual orientation.

Legal grounds for processing personal data:

1. the data subject has given **consent** to the processing of his or her personal data for one or more specific purposes;

2. processing is necessary for the performance of a contract to which the data subject is party or in order to take steps at the request of the data subject prior to entering into a contract;
3. processing is necessary for compliance with a legal obligation to which the controller is subject;
4. processing is necessary in order to protect the vital interests of the data subject or of another natural person;
5. processing is necessary for the performance of a task carried out in the **public interest** or in the exercise of official authority vested in the controller;
6. processing is necessary for the purposes of the legitimate interests pursued by the controller or by a third party, except where such interests are overridden by the interests or funda-

mental rights and freedoms of the data subject which require protection of personal data, in particular where the data subject is a child.

7. processing is necessary for compliance with a legal obligation to which the controller is subject under Union or Member State law requiring the processing of personal data for a specific purpose.

Data Controller (PUA) under GDPR

- The entity that **determines the purposes and means** of the processing of personal data
- Bears the **primary legal responsibility**
- Responsible for:
  - Lawful basis
  - Compliance with GDPR principles
  - Transparency and information to data subjects
  - Appropriate technical and organisational measures
- Typical examples:
  - Public authorities
  - Universities
  - Regions and municipalities

Data Processor (PUB) under GDPR

- Processes personal data **on behalf of the controller**
- Acts **only on documented instructions** from the controller
- May **not** determine purposes of processing
- Has direct responsibilities for:
  - Security of processing (Article 32)
  - Confidentiality
- Must be governed by a **data processing agreement**

Controller–Processor Relationship

- A formal **Data Processing Agreement (DPA)** is required
- The agreement must specify:
  - Subject matter and duration
  - Nature and purpose of processing
  - Types of personal data
  - Categories of data subjects
  - Security measures
- The controller remains responsible even when processing is outsourced

Example: Sahlgrenska

- If a researcher work at the Sahlgrenska hospital, VGR might be the data controller (PUA)
- If he/she asks for statistical consulting from Sahlgrenska Academy, GU might be the data processor (PUB)

Technical Safeguards

Examples of technical safeguards include:

- Pseudonymisation
- Encryption of personal data
- Access controls and authentication
- Logging and monitoring of access
- Secure storage and transmission

Organisational Safeguards

Examples of organisational safeguards include:

- Defined roles and responsibilities
- Internal policies and procedures
- Staff training and confidentiality obligations
- Data protection by design and by default
- Incident and breach response procedures
- Documentation and accountability measures

Data Minimisation and Purpose Limitation

- Only data that are **necessary** are processed
- Data are processed **only for specified purposes**
- Access is limited to authorised personnel
- Retention periods are defined and respected

Pseudonymisation and Anonymisation

- **Pseudonymisation** reduces risks while allowing reuse
- Identifiers are kept separately and protected
- Anonymisation removes data from GDPR scope (if irreversible)

Safeguards for Special Categories of Data

When processing sensitive data (e.g. health data), additional safeguards are required:

- Strict access control
- Enhanced security measures
- Ethical approval where applicable
- Clear separation of identifiers and content data

GDPR and Swedish law

- GDPR is directly applicable in Sweden
- There are refereces to GDPR in Swedish laws such as PDL and OSL
- Swedish laws may provide additional regulations and requirements beyond GDPR
- Data protection authorities in Sweden: Integritetsskyddsmyndigheten (IMY)
- Should be easy to collaborate across EU borders due to GDPR, but more difficult with non-EU countries

Etikprövningslagen (EPL)

- Regulates ethical review of research involving humans (including their data!)
  - Had received some critisism and might be revised
- Applies to research projects conducted in Sweden
- Requires ethical review and approval by an ethics review board
- Aims to protect the rights, safety, and well-being of research participants
- Based on the Declaration of Helsinki and other international ethical guidelines
- One application for each new research project
  - Ammendments for changes in already approved projects
- Application fees applies
- Swedish Ethical Review Authority

International laws

- Note that other countries have different laws and regulations
- In USA, for example, HIPAA regulates the use and disclosure of protected health information (PHI)
  - Different states have different laws as well
- When collaborating internationally, compliance with all relevant laws is required

Reading

Public access and secrecy

EL3: Version control

History

> It is widely acknowledged that the most fundamental developments in statistics in the past 60 years are driven by information technology (IT). We should not underestimate the importance of pen and paper as a form of IT but it is since people start using computers to do statistical analysis that we really changed the role statistics plays in our research as well as normal life.

> Although: "Let's not kid ourselves: the most widely used piece of software for statistics is Excel."" /Brian Ripley (2002)

Short overview

General-Purpose Programming Languages

Early statistical computing relied heavily on:

- **FORTRAN**
  - Dominant language for numerical and statistical computation
  - Statistical methods implemented as libraries and subroutines

  - Still used as subroutines in modern statistical software!
- **ALGOL / ALGOL 60**
  - Used mainly in academic environments
- **PL/I**
  - Used in some government and industrial contexts

📌 These languages required substantial programming expertise.

Early Statistical Packages

Several dedicated statistical systems emerged:

- **SPSS** (1968)
  - Originally batch-oriented
  - Widely used in social sciences
  - Originaly "Statistical Package for the Social Sciences"
- **BMDP** (1960s)
  - Developed at UCLA
  - Common in medical statistics
  - Bio-Medical Data Package
- **GENSTAT** (1968)
  - Focused on agricultural statistics
- **MINITAB** (1972)
  - Designed for teaching and education
  - Still popular in quality control

SAS: A Transitional System

- **SAS** (early 1970s)
- Developed for agricultural and biostatistical analysis
- Script-based, but largely batch-oriented
- Became a standard in:
  - Government agencies
  - Large organisations
- Known for strong data management capabilities
- Still widely used in pharmaceutical industry and clinical trials

📌 SAS predates S but influenced later statistical workflows.

Limitations of Pre-S Systems

Common limitations included:

- Batch processing rather than interactivity
- Separation of data management and analysis
- Limited graphics capabilities
- High barriers to exploratory data analysis

S

- S takes form at Bell Laboratories (interactive statistical computing).
- John Chambers leads the effort.
- **1976:** first working version of S runs on GCOS
- **1979:** S2 is ported to UNIX; UNIX becomes the primary platform
- **1980:** S is first distributed outside Bell Labs
- **1981:** source versions are made available
- **1984:** key S books published (often called the "Brown Book" era)

The New S Language

- **1988:** "New S" is released (major language redesign)
- **1988: S-PLUS** is first released as a commercial implementation of S
- **1991:** *Statistical Models in S* ("White Book") popularizes formula notation (the ~ operator), data frames, and modeling workflows

R

- **1993:** first versions of **R** are published (Auckland; Ross Ihaka & Robert Gentleman)
- **1995:** R becomes open source (GPL)
- **1997:** the R Core group forms; **CRAN** is founded (Kurt Hornik)
- **2000: R 1.0.0** is released 2000-02-29

RStudio brings an IDE to the R community

- **2009:** RStudio (the company) is founded
- **2011:** RStudio IDE is introduced as an open-source IDE for R (desktop + server)

Microsoft (Revolution Analytics)

- **Jan 2015:** Microsoft announces it will acquire **Revolution Analytics**
- Microsoft promotes enterprise R offerings (e.g., Microsoft R Open / R Server)
- **2016:** SQL Server 2016 introduces **R Services** (in-database R)
- **2017:** Microsoft expands the stack under "Machine Learning Server" branding
- **June 2021:** Microsoft announces retirement of **Microsoft Machine Learning Server**
- **July 1, 2023:** Microsoft era over

RStudio becomes Posit

- **July 27, 2022:** RStudio rebrands as **Posit**
- **July 28, 2022: Quarto** is announced as a next-generation scientific and technical publishing system (multi-language, multi-engine)

Modern IDE

Positron

- New generation IDE for data science
- From Posit PBC
- Free for individual use

- Based on Code OSS (open source version of VS Code from Microsoft)
- For both R and Python (Julia?)

Quick tour

> ! Positron assistant (mentioned in the video)
>
> This feature will most likely be disabled in any secure working environment. Such environments often have strict rules about data privacy and security, which may conflict with the assistant's functionality. Health data in SENSITIVE and SECURE environments must not be shared with external services, including AI assistants, to comply with data protection regulations and institutional policies.
>
> It is recommended to not rely on such tools during the course (even if all our data is synthetic). If you start to rely on such tools, you might get difficulties the day you work with real data (might lead to prosecution for "brott mot tystnadsplikten" which is not only public, but actually civil law ("Brottsbalken") with prision sentence as a possibility). Society put an extreme emphasis on protecting health data, and rightfully so!

Version control

Before Version Control Systems

1950s–1970s: Early software development relied on:

- Manual file naming:
  - `analysis_final.f`
  - `analysis_final_v2.f`
- Physical media:
  - Punch cards
  - Magnetic tape
- Centralised mainframes

📌 No automated tracking of changes.

Floppy discs, Mail, and Shared Directories

1970s–1980s: Common practices included:

- Copying files to:
  - Floppy disks
  - Magnetic tapes
- Sending media by **postal mail**
- Sharing files via:
  - Network drives
  - FTP servers

📌 Version control was social, not technical.

Early Version Control Systems

1980s: First-generation tools focused on single files:

- **SCCS** (Source Code Control System, 1972)
- **RCS** (Revision Control System, 1982)

Characteristics:

- Versioning per file
- Linear history
- Central storage

Centralised Version Control

1990s: Project-level systems emerge:

- **CVS** (Concurrent Versions System)
- **Subversion (SVN)**

Key features:

- Central repository
- Multiple users
- Check-in / check-out model

📌 Still required constant access to the central server.

Limitations of Centralised Systems

Common problems:

- Single point of failure
- Poor support for branching and merging
- Difficult offline work
- Slow operations on large repositories

These limitations became critical for large projects.

Git

- **Git** is created by Linus Torvalds in 2005
- Original motivation:
  - Support Linux kernel development
  - Replace proprietary tools

Design principles:

- Distributed architecture
- Fast local operations
- Strong support for branching and merging

Distributed Version Control with Git

Key ideas in Git:

- Every clone is a full repository
- Local commits without network access
- Cheap and fast branches
- Cryptographic integrity (hash-based)

📌 Collaboration becomes more flexible and robust.

Hosting Platforms

Platforms built around Git:

- **GitHub** (2008)
- **Bitbucket** (2008)
- **GitLab** (2011)
- Gitea - open source alternative to GitHub (get the same functionallity locally or on a server)

They add:

- Pull requests / merge requests
- Issue tracking
- Code review
- CI/CD integration

Version Control Today

Modern usage includes:

- Code
- Documentation
- Data analysis (scripts, notebooks)
- Configuration and infrastructure

Git is integrated into:

- IDEs (VS Code, Positron)
- CI/CD pipelines
- Cloud platforms

Version Control Beyond Code

Today, version control supports:

- Reproducible research
- Collaborative writing
- Data science workflows
- Teaching and learning

📌 Version control is now a core professional skill.

WARNING!

- Not everything should be shared!
- Scripts and documentation yes!
- But **Health data is sensitive!**
  - Do not share it!
  - Avoid unintentional sharing!
  - Private repositories are still shared with hosting provider!
- Avoid explicit file paths and sensitive info in scripts!
- Can give information about data location and internal structure!
- No hard-coded passwords or API keys!

Git basics

(After installing the Git software)

- Collect all files related to a project in a folder
- Initialize a git repository in that folder
- Make changes to files
- Stage changes for commit
- Commit changes with a message
- Possibly push commits to remote repository

```
cd path/to/your/project
git init
git status
## make changes to files
git add filename1 filename2
git commit -m "Descriptive message about changes"
git remote add origin
```

Video tutorials

The video below is a good start to understand the basic concepts of Git and GitHub (and there are others to be found on YouTube). Note that there are a lot of videos on Git combined with Rstudio. They might still be relevant even though we are using Positron here.

Inspirational video

Watch this video even though some parts might be overwhelming. It gives a good overview of the current state (2025), even though many things will be too advanced for this course (it is not specifically aimed for statisticians or R users).

! .gitignore

The `.gitignore` file is very important in settings with health data! Pay close attention to this section of the video!

Git in Positron

- Remember that Positron is build on Code OSS (which shares a lot of features with VS Code).
- Branching and merging is possible but we will not cover that here.

Short official introduction from Microsoft:

More detailed introductions. Watch both! The first one is based on a Windows version of VS code and the second on Mac but the concepts are the same:

> ℹ Overwhelmed?
>
> This video includes some parts which might be overwhelming if you are new to Git and GitHub. Don't worry! You don't need to understand everything right away. Just try to follow along with the basic concepts and steps. You will get more comfortable with practice.

Statistics projects

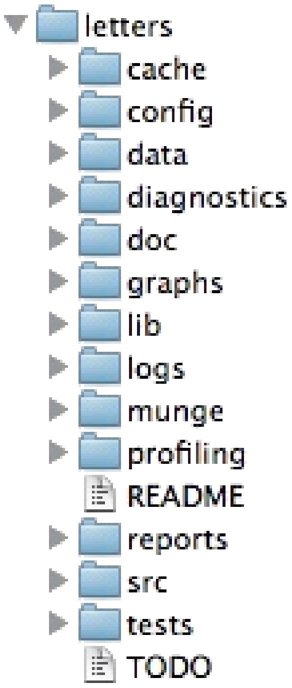
Not a single R script

- Real projects are more complex than a single R script!
- Multiple scripts
- Data files
- Documentation
- Reports
- Version control
- Reproducible workflows

Project structure

Common file structures

- Help you organize your thoughts
- Help others to collaborate
- simplifies paths used in your code



```
letters
  cache
  config
  data
  diagnostics
  doc
  graphs
  lib
  logs
  munge
  profiling
  README
  reports
  src
  tests
  TODO
```

Example structure

```
/.../my_project/
  ├── README.md        - project documentation
  ├── TODO             - what should be done next?
  ├── .git             - handled by git (hidden folder)
  ├── .gitignore       - used by git but your responsibility!
  ├── data/            - your data files (not under version control!)
      ├── cancer.csv
      └── patients.qs
  ├── R/               - your saved R functions
      ├── function1.R
      └── function2.R
  ├── reports/
  └── _targets.R       - targets pipeline script
```

`README.md`

- Document the purpose of the project
- What is it about?
- What is the aim?
- Who to contact for questions?
- In what circumstances was it created?

Markdown format (simple text with some possible formatting)

`data` folder

- Store your data files here as they are when you get them
- Avoid any modifications to the raw files!
- It is very easy to forget what you do if it can not be traced by code
- Do NOT include this folder in version control!
- Git is not good at handling large files
- Sensitive data should not be shared!
- Add `data/*` to your `.gitignore` file
- In realistic projects, data might come in varying formats
  - csv, txt, xlsx, sas7bdat, sav, dta, etc
  - some files might be very big (gigabytes not uncommon)

`R` folder

- Store your R functions here
- Document their purpose inline!
- Helps you to reuse code
- Easier to read main scripts
- Easier to test and debug code
- Easier to share code between projects

`reports` folder

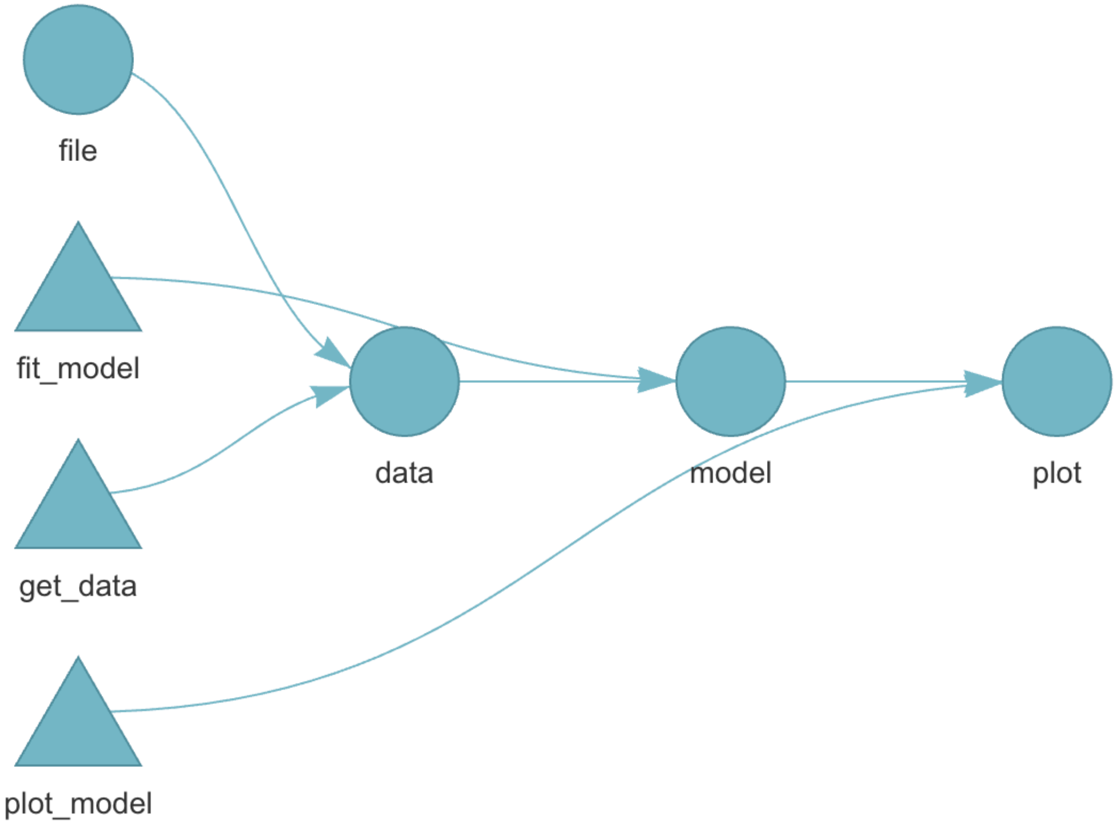
- Store your reports here
- Quarto documents
- Documemnt your analysis
- Include figures and tables
- Share with external collaborators

`_targets.R` file

Use the `targets` package to create a reproducible data analysis pipeline

Pipeline

- Define steps in your analysis as targets
- Define dependencies between targets
- Automatically track changes and rerun only necessary parts



Why?

- You do not want to rerun everything all the time!
- You want to keep track of what you have done
- You want to be able to reproduce your results later
- You want to share your workflow with others

Reading

- A bit old but still relevant: https://happygitwithr.com/
- Targets overview
- Target manual

EL4: Analysis pipelines

Not a single R script

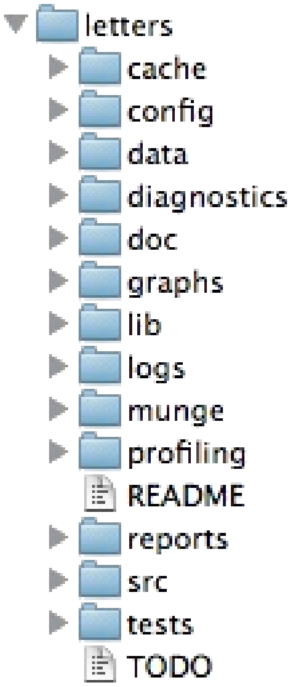
- Real projects are more complex than a single R script!
- Multiple scripts
- Data files
- Documentation
- Reports

- Version control
- Reproducible workflows

Project structure

Common file structures

- Help you organize your thoughts
- Help others to collaborate
- simplifies paths used in your code



letters
cache
config
data
diagnostics
doc
graphs
lib
logs
munge
profiling
README
reports
src
tests
TODO

Example structure

```
/.../my_project/
  ├── README.md        - project documentation
  ├── TODO             - what should be done next?
  ├── .git             - handled by git (hidden folder)
  ├── .gitignore       - used by git but your responsibility!
  ├── data/            - your data files (not under version control!)
      ├── cancer.csv
      └── patients.qs
  ├── R/               - your saved R functions
      ├── function1.R
      └── function2.R
  ├── ...
  ├── ...
  └── _targets.R       - targets pipeline script (WHAT??? :-))
```

`README.md`

- Document the purpose of the project
- What is it about?
- What is the aim?
- Who to contact for questions?
- In what circumstances was it created?

Markdown format (simple text with some possible formatting)

Markdown

TODO: intro and give examples + link etc

`data` folder

- Store your data files here as they are when you get them
- Avoid any modifications to the raw files!
- It is very easy to forget what you do if it can not be traced by code
- Do NOT include this folder in version control!
- Git is not good at handling large files
- Sensitive data should not be shared!
- Add `data/*` to your `.gitignore` file
- In realistic projects, data might come in varying formats
  - csv, txt, xlsx, sas7bdat, sav, dta, etc (we will cover some of these later)
  - some files might be very big (gigabytes not uncommon)

`.gitignore` fiile

TODO

`R` folder

- Store your R functions here
- You have learned about R functions in the earlier R course
- We will cover some more of that later
- Document their purpose inline!
- Helps you to reuse code
- Easier to read main scripts if functions are defined elsewhere
- Easier to test and debug code
- we eill not cover testing but some about debugging later
- Easier to share code between projects

`reports`

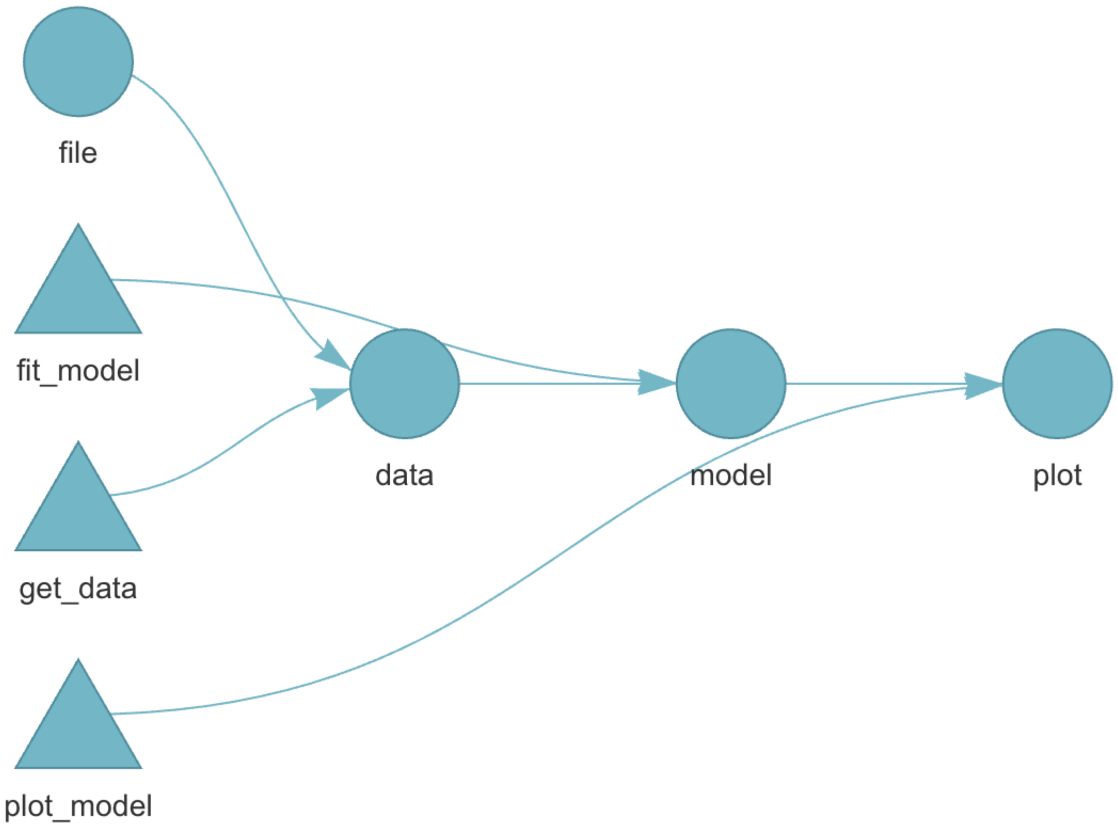
- Separate reports from analysis/data management but integrate them with the analysis pipeline
- Quarto documents
- Documemnt your analysis and thoughts/decisions along the way
- Include figures and tables
- To share with external collaborators

`_targets.R` file

Use the `targets` package to create a reproducible data analysis pipeline

Pipeline

- Define steps in your analysis as targets
- Define dependencies between targets
- Automatically track changes and rerun only necessary parts



Why?

- You do not want to rerun everything all the time!
- You want to keep track of what you have done
- You want to be able to reproduce your results later
- You want to share your workflow with others

Reading

- Targets overview
- Target manual

Bibliography

[1] J. F. Ludvigsson, P. Otterblad-Olausson, B. U. Pettersson, and A. Ekbom, "The Swedish personal identity number: Possibilities and pitfalls in healthcare and medical research," *European Journal of Epidemiology*, vol. 24, no. 11, pp. 659–667, 2009, doi: 10.1007/s10654-009-9350-y.

[2] K. Laugesen *et al.*, "Nordic Health Registry-Based Research: A Review of Health Care Systems and Key Registries," *Clinical Epidemiology*, pp. 533–554, Jul. 2021, doi: 10.2147/CLEP.S314959.

[3] A. Nguyen, *Hands-on healthcare data: taming the complexity of real-world data*, First edition. Beijing Boston Farnham Sebastopol Tokyo: O'Reilly, 2022.