TALL

SOM FORTELLER

Introduction course

DAY 6 – 26TH JANUARY 2021

SUSIE JENTOFT & ASLAUG HURLEN FOSS

Statistisk sentralbyrå
Statistics Norway

Agenda

	Monday **25th January**	**Tuesday** **26th January**	**Wednesday** **27th January**
12:00	• General: indexing, vectors and lists	• Summary • Functions	• Building packages
12:45	Exercise 7	Exercise 9	Exercise 11
13:15	Break	Break	Break
13:30	• Sorting • Control with if and else • Loops	• RMarkdown • Dashboards	• Other useful packages and resources
14:15 – 15:00	Exercise 8	Exercise 10	Discussion and summary

Review of exercise 10

R packages

- Collection of code (functions and examples), documentation, data, tests
- Standard structure and requirements
- Easy to share

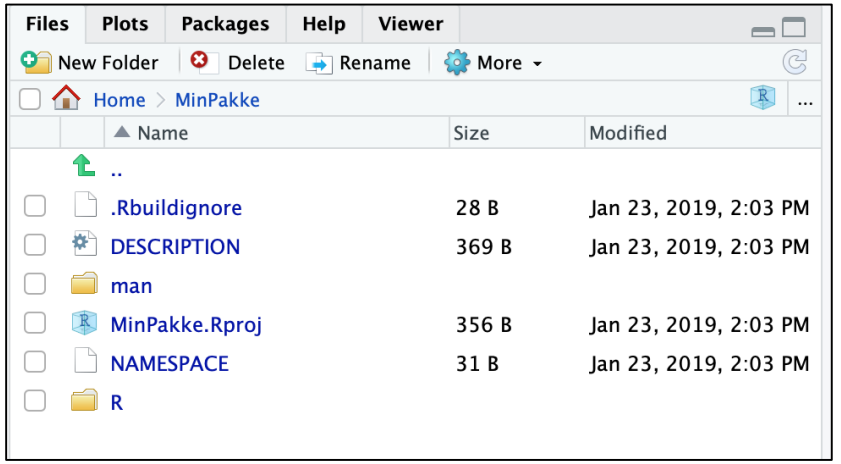
Why you should create your own packages

- Easy to share and move code
- Organised
- Built in data, parameters, tests
- Documentation for yourself and others

How?

- Using RStudio
  - File > New Project > New Directory > R Package
- Follow standard structure
  - Description
  - Namespace
  - Code and documentation

Structure

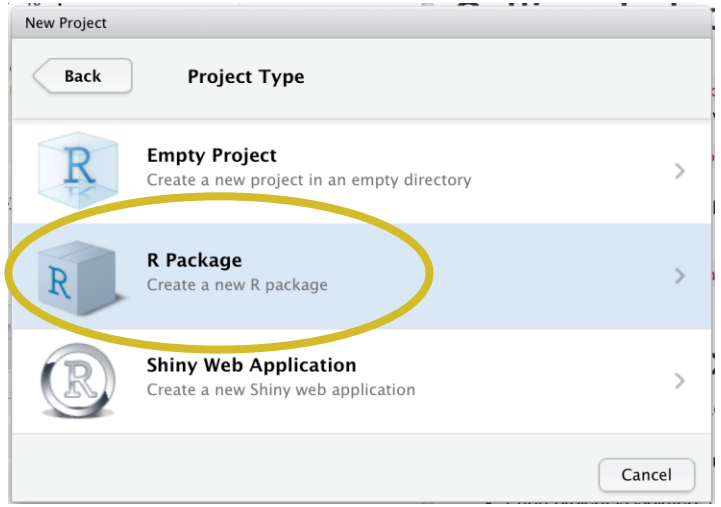


- Four components:

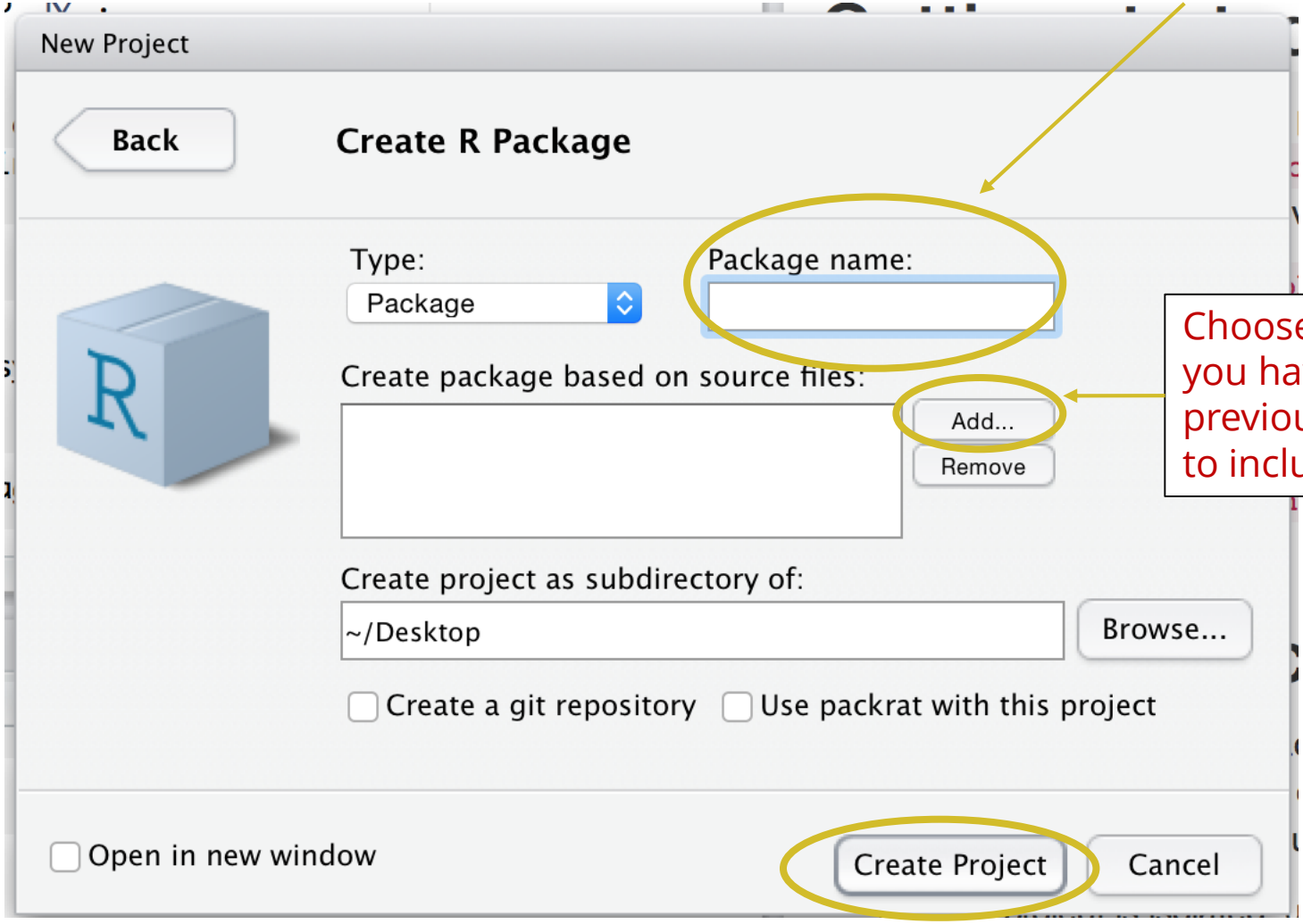
1. `R/` Folder with R code
2. `man/` Folder with function descrpitions
3. `DESCRIPTION` File which describes the package
4. `NAMESPACE` Specifies how the package communicates with other packages

Additionally, an Rstudio prosjekt fil, `pkgname.Rproj` and `.buildignore` files are created

File > New Project > New Directory > R Package



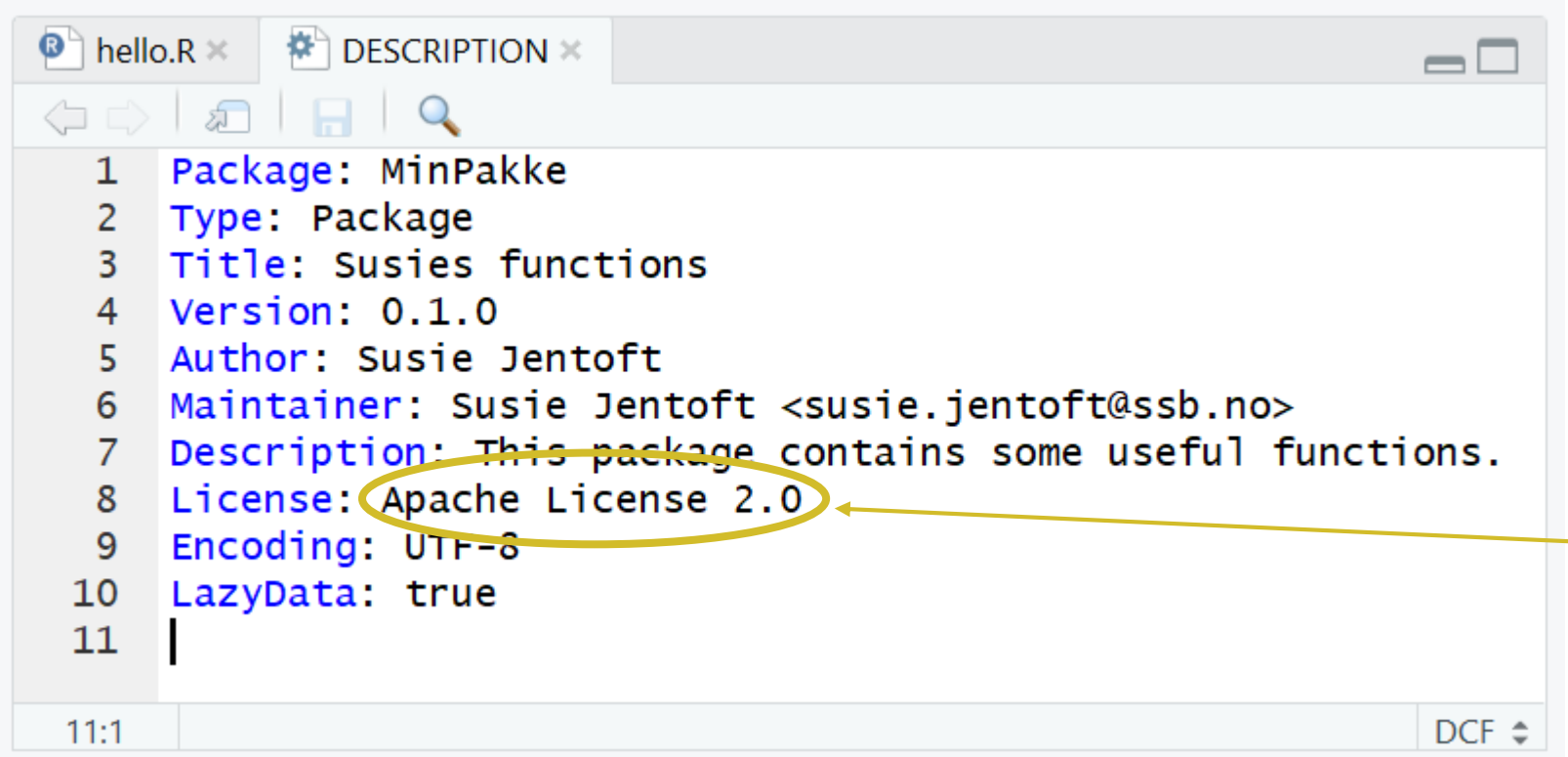
Name of package



Choose Add if you have previous code to include.

Statistisk sentralbyrå
Statistics Norway

Description file



For external publishing

Roxygen

- Generates documentation
- `install.packages(roxygen2)`
- Build > Configure Build Tools... > Build Tools

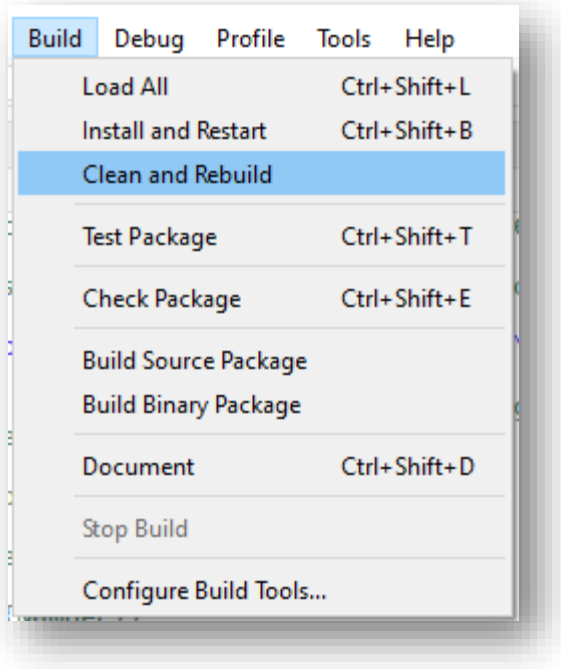
Documentation of functions

- Roxygen2

```
#' Norwegian sanchwich preferance
#'
#' @param paalegg String for the spread which is liked
#' @param broed String for bread type preferance
#' @return A string giving the bread and spread preferance
#' @examples
#' smoerbroed("jam")
smoerbroed <- function(paalegg, broed = "brød"){
   melding <- paste("Jeg liker", broed, "med", paalegg)
   melding
}
```

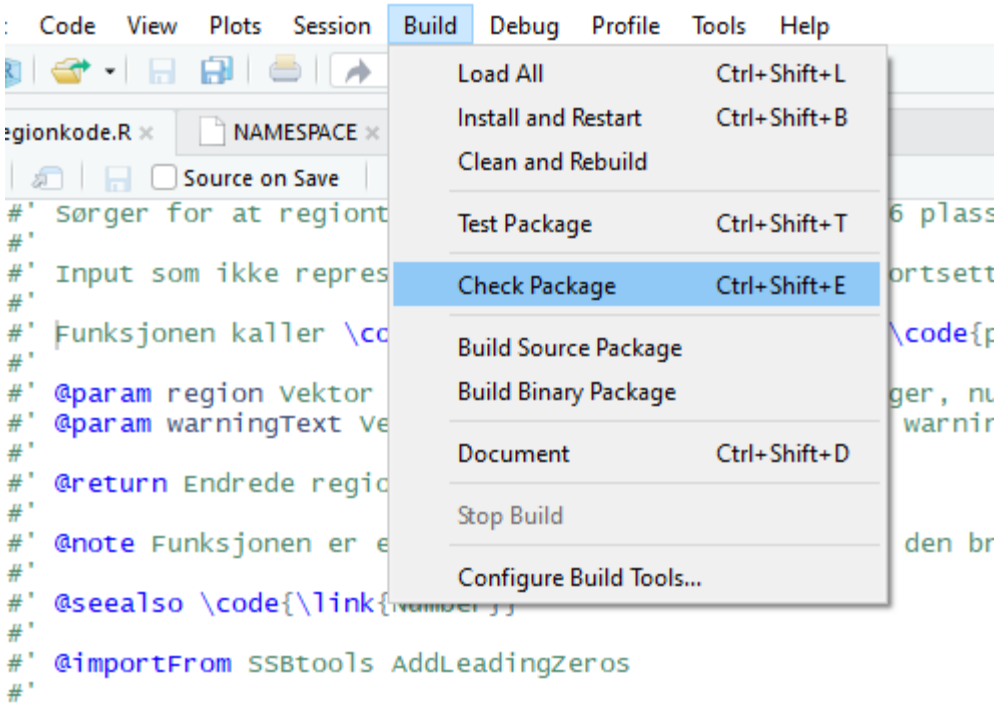
Building and sharing packages

- Clean and Rebuild to build package for yourself
  - Updates code and documentaion
- Share package
  - Build Source Package
  - Build Binary Package (for windows-installasjon)



Check package

- Check code and documentation, structure with Check Package
  - Useful even if you are not publishing
  - Checks:
    - Function parameters are documented
    - Examples work
    - Functions from other pacakges are imported
    - ...
- Errors don't necessarily mean the packages doesn't work
  - Some strict CRAN requirements

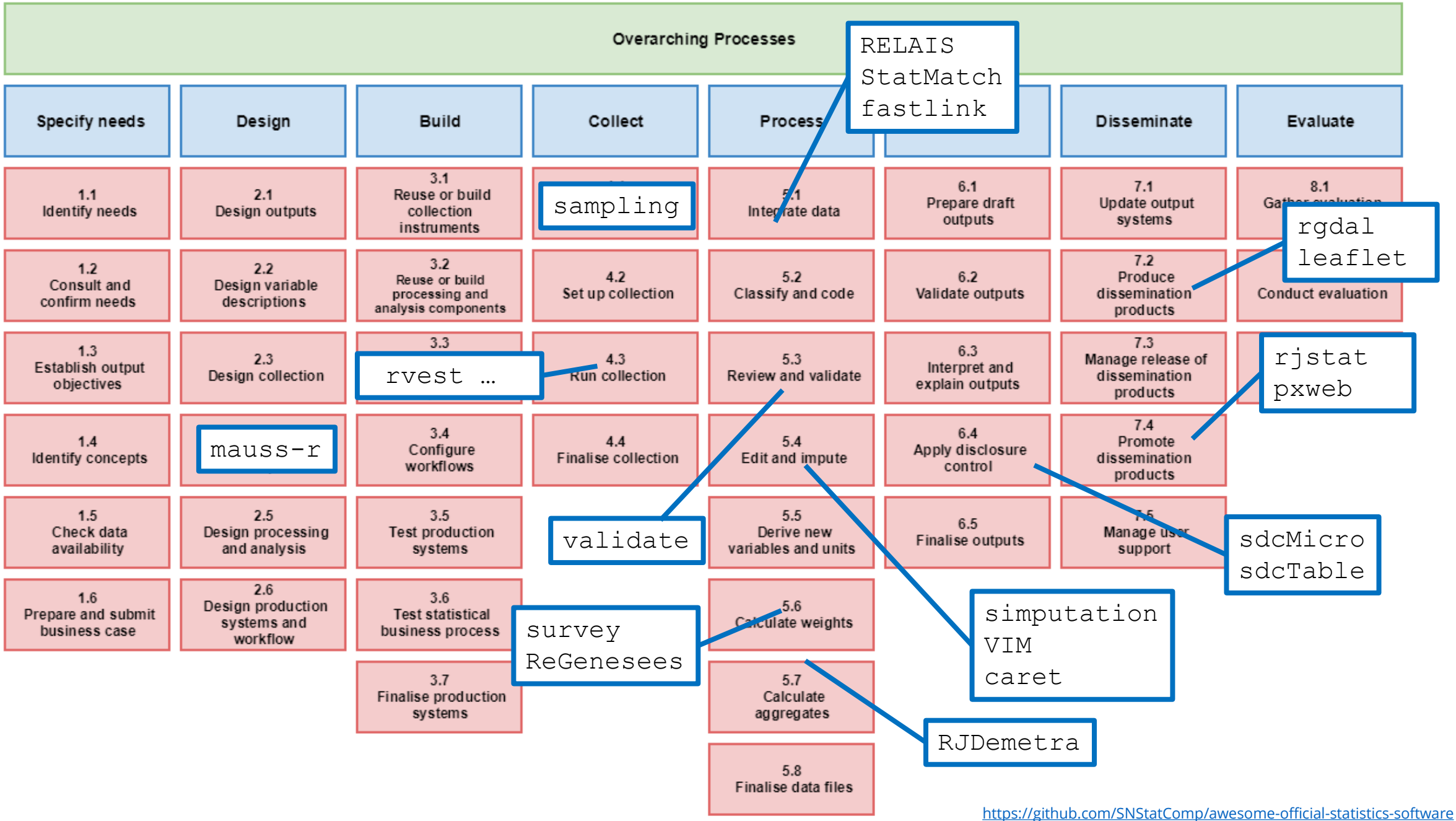


Statistisk sentralbyrå
Statistics Norway

More can be included in packages

- README
  - Markdown file with extra information on the package. Shows up on Github.
- /data
  - Include datasets in package (.Rdata)
  - Should to be documented
  - Access with data(dataname)
- /tests
  - Unit tests can be included
- /vignettes
  - More information on package and how to use it can be included.
  - R Markdown can be used

Exercise 11



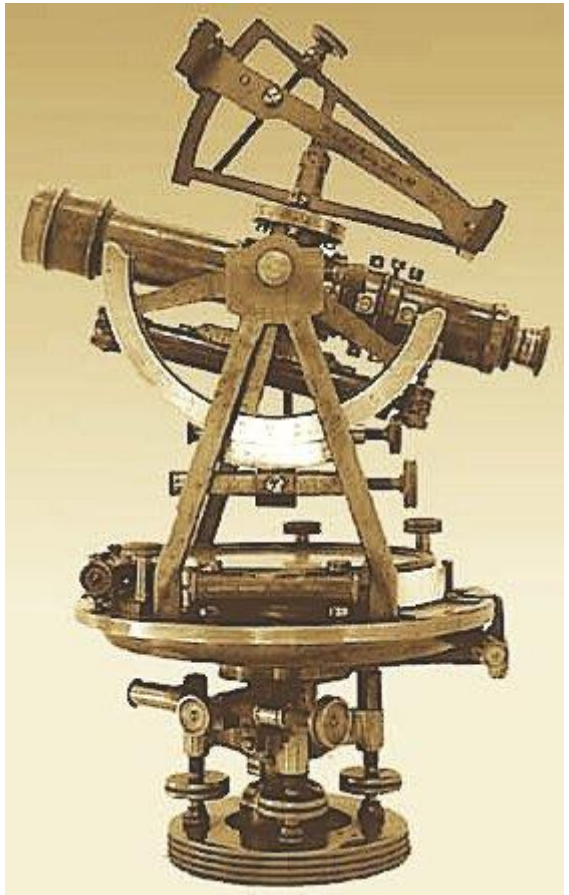
https://github.com/SNStatComp/awesome-official-statistics-software

Packges, resources and ideas

- Estimation: `survey` and `ReGenesees`
- Modelling and machine learning: `caret`
- Maps/geodata in R
- `learnr` – create R tutorials

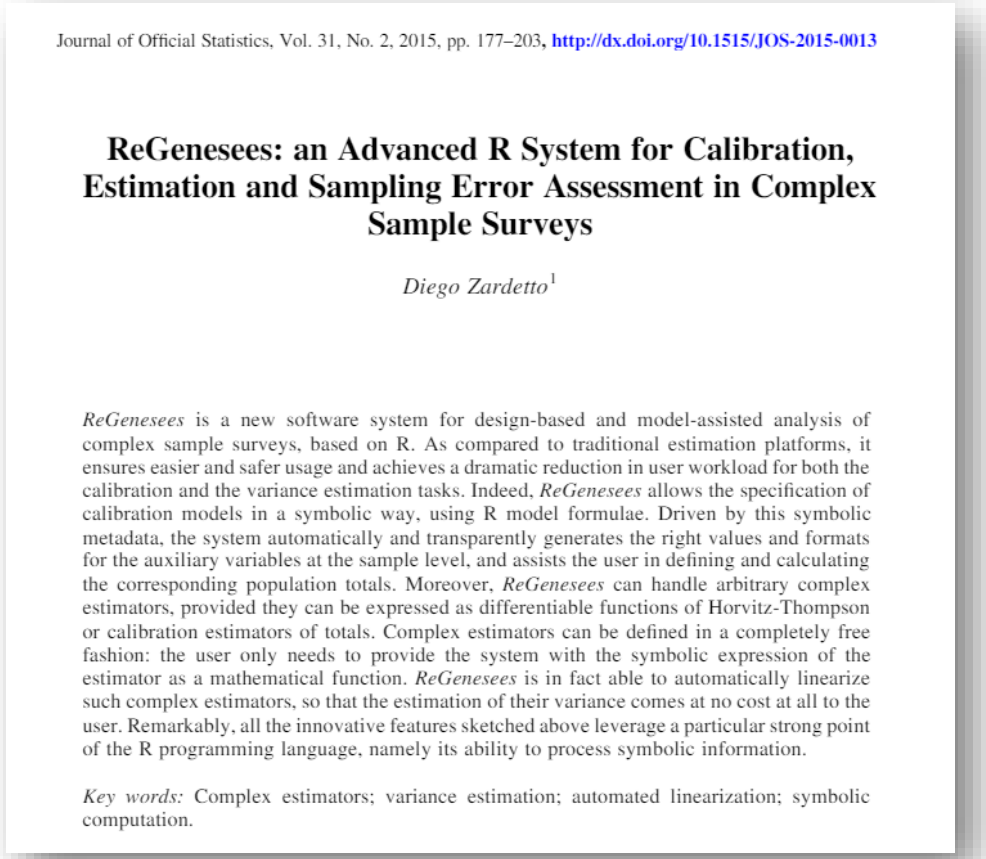
survey

- Facilities in R for analyzing data from complex surveys
- Point og variance estimation
  - Horvitz-Thompson and Yates_Grundy estimators
- Multistage sampling designs and multivariate analysis
- GREG estimation generalized raking/calibration
- https://r-survey.r-forge.r-project.org/survey/



ReGenesees

- R package for design-based and model-assisted analysis of complex sample surveys.
- Handles more complex sampling designs
- Additional calibration options
- Point and variance estimation methods
- https://diegozardetto.github.io/ReGenesees/



Journal of Official Statistics, Vol. 31, No. 2, 2015, pp. 177–203, http://dx.doi.org/10.1515/JOS-2015-0013

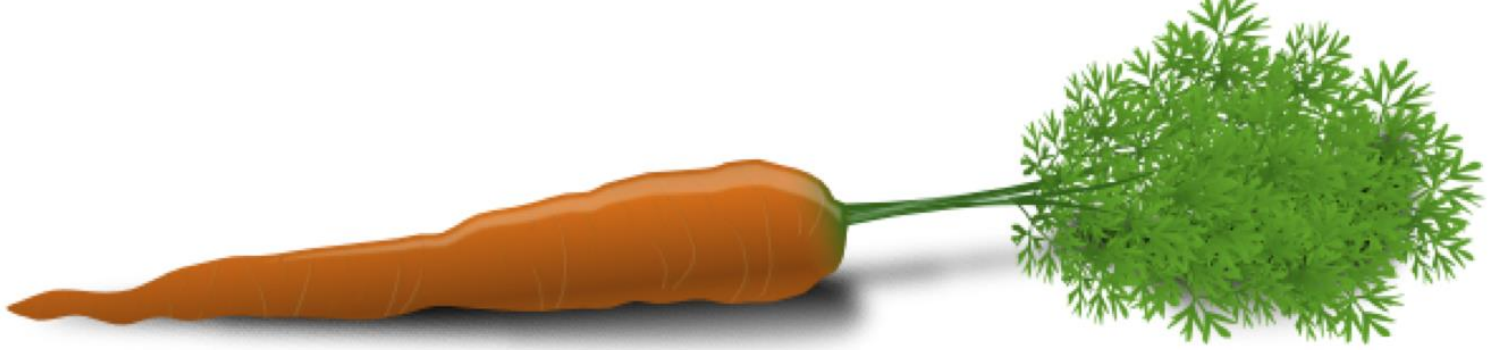
ReGenesees: an Advanced R System for Calibration, Estimation and Sampling Error Assessment in Complex Sample Surveys

Diego Zardetto[1]

ReGenesees is a new software system for design-based and model-assisted analysis of complex sample surveys, based on R. As compared to traditional estimation platforms, it ensures easier and safer usage and achieves a dramatic reduction in user workload for both the calibration and the variance estimation tasks. Indeed, *ReGenesees* allows the specification of calibration models in a symbolic way, using R model formulae. Driven by this symbolic metadata, the system automatically and transparently generates the right values and formats for the auxiliary variables at the sample level, and assists the user in defining and calculating the corresponding population totals. Moreover, *ReGenesees* can handle arbitrary complex estimators, provided they can be expressed as differentiable functions of Horvitz-Thompson or calibration estimators of totals. Complex estimators can be defined in a completely free fashion: the user only needs to provide the system with the symbolic expression of the estimator as a mathematical function. *ReGenesees* is in fact able to automatically linearize such complex estimators, so that the estimation of their variance comes at no cost at all to the user. Remarkably, all the innovative features sketched above leverage a particular strong point of the R programming language, namely its ability to process symbolic information.

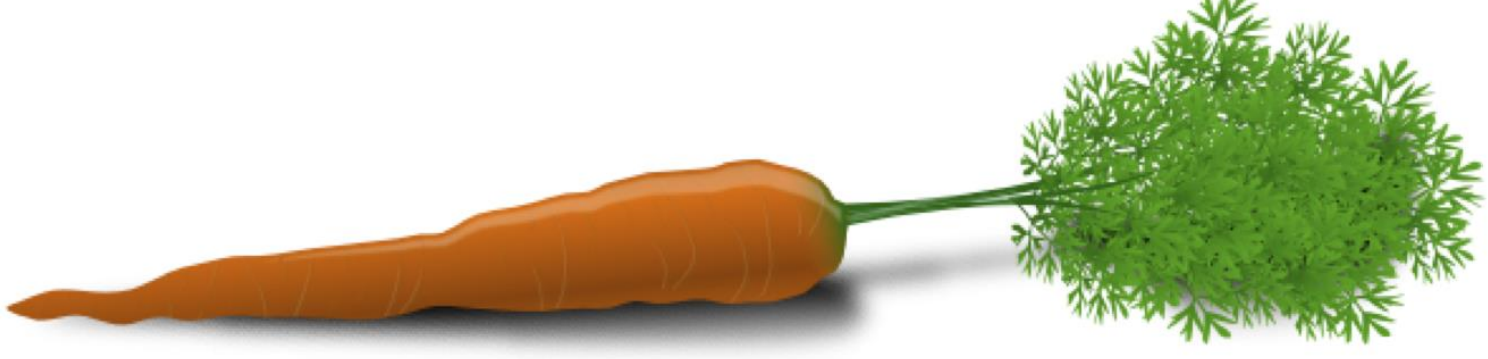
Key words: Complex estimators; variance estimation; automated linearization; symbolic computation.

caret

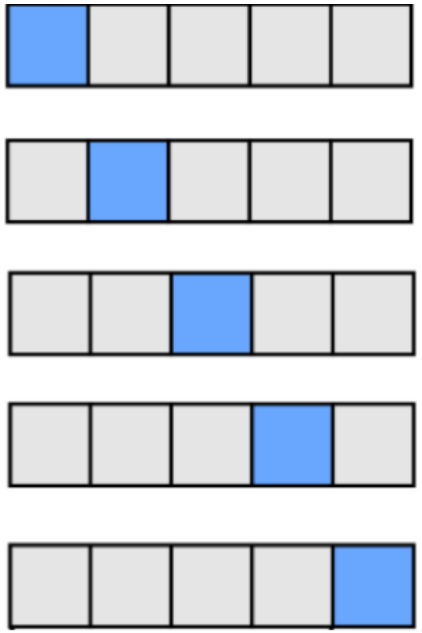


- Consistent interface to train a large number of algorithms
  - kNN, randomforest, lasso, svm, neural networks
  - https://topepo.github.io/caret/available-models.html
- Tools to evaluate optimal models
- Evaluate model performance

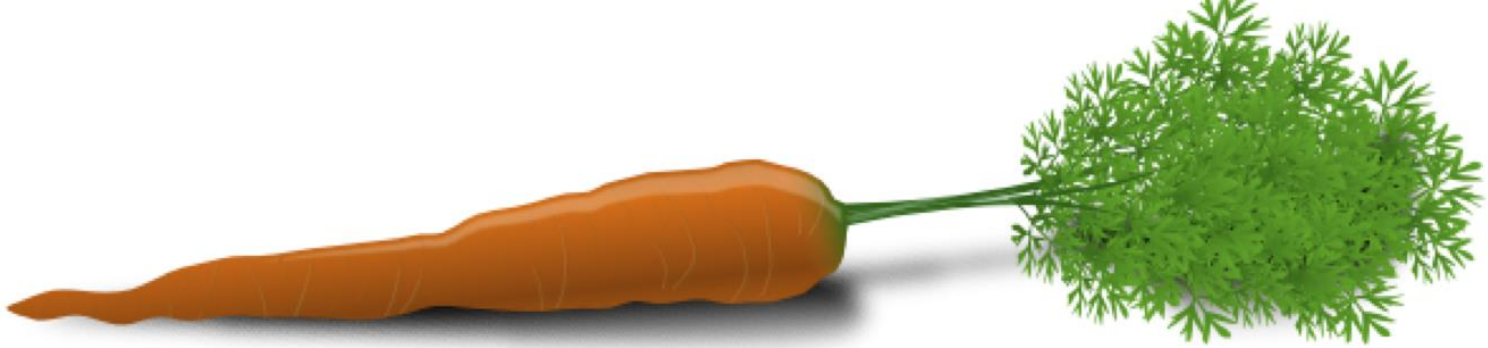
caret



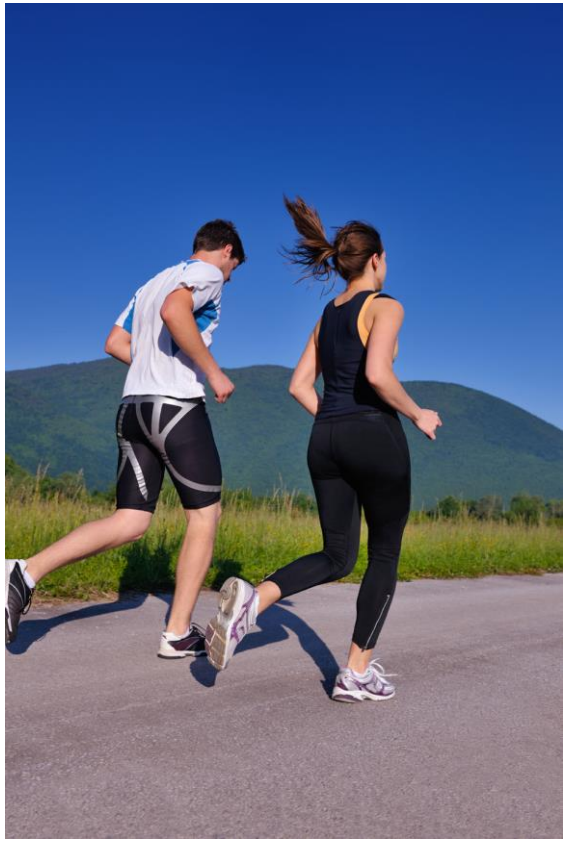
- Data splitting:
  - `createDataPartition()`
  - `createFolds()`



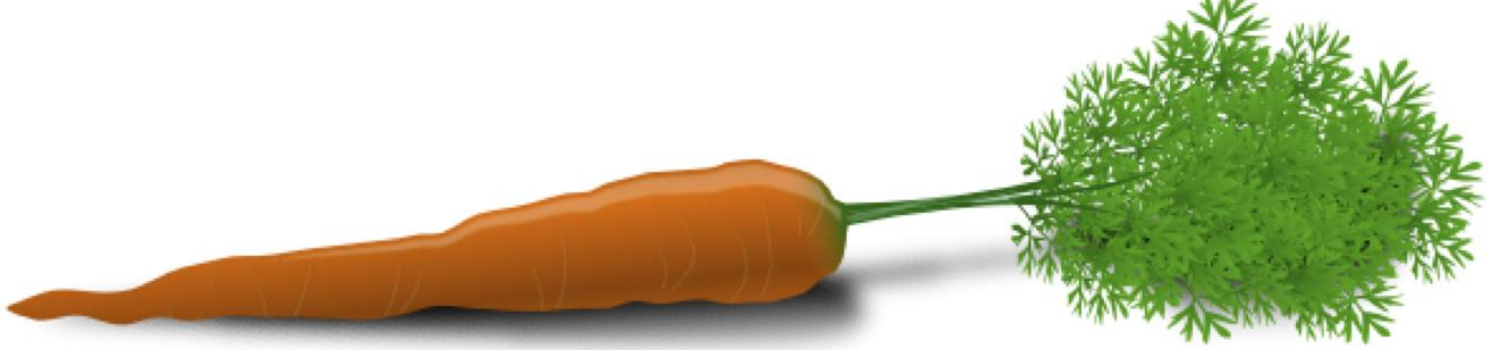
caret



- Model training and parameter tuning
  - `trainControl()`
  - `train()`



caret

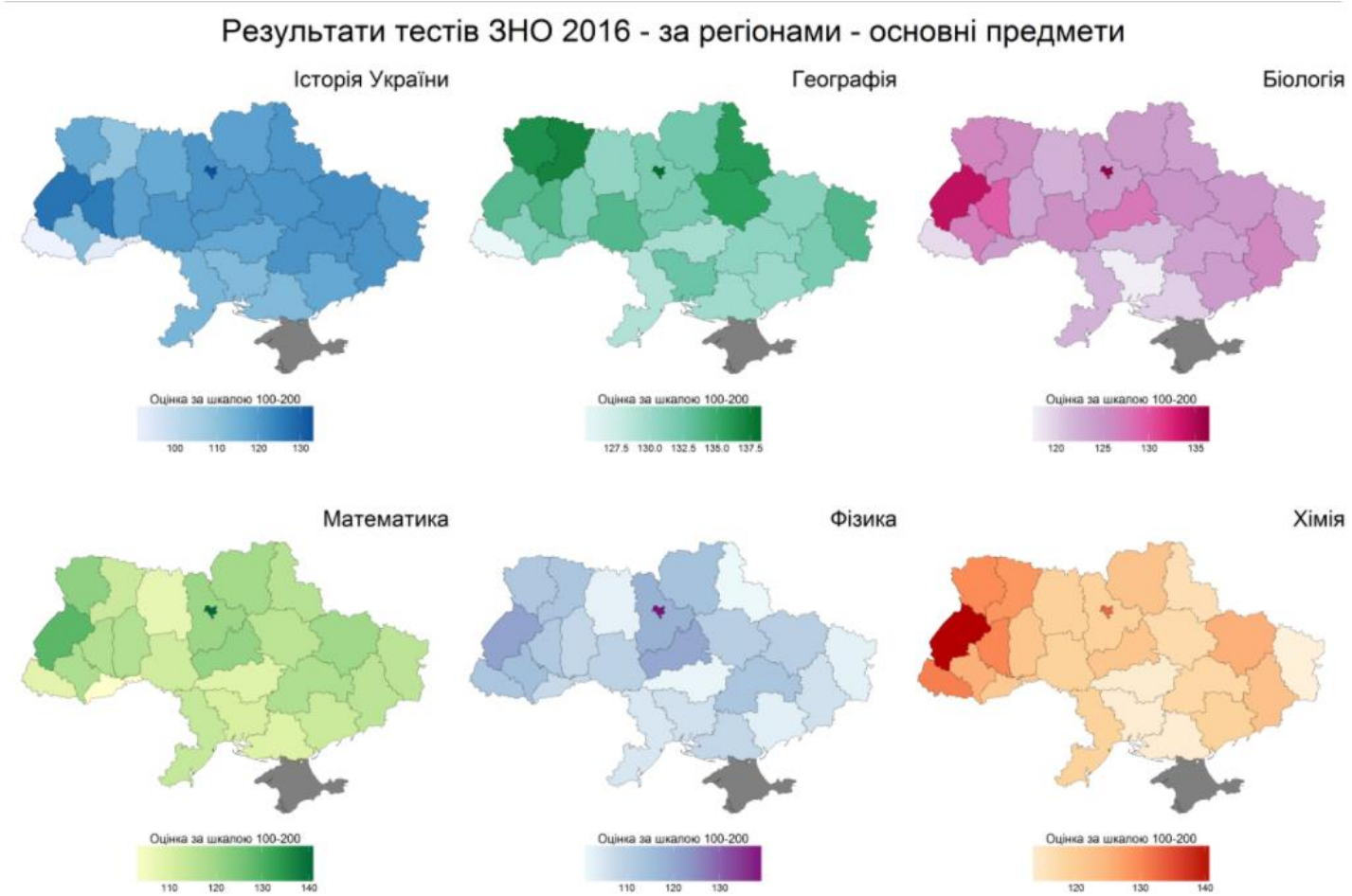


- Fit model on new data
  - `predict()`

Geodata in R

- Lot of packages!
- **`rgdal`** for reading in spatial data to R
  - Shape files
  - geojson
- **`ggplot() + geom_sf() or geom_polygon()`**
- **`leaflet`** for creating interactive maps
  - Combine with shiny for more user input

ggplot() + geom_polygon()



https://rpubs.com/mgontar/214015

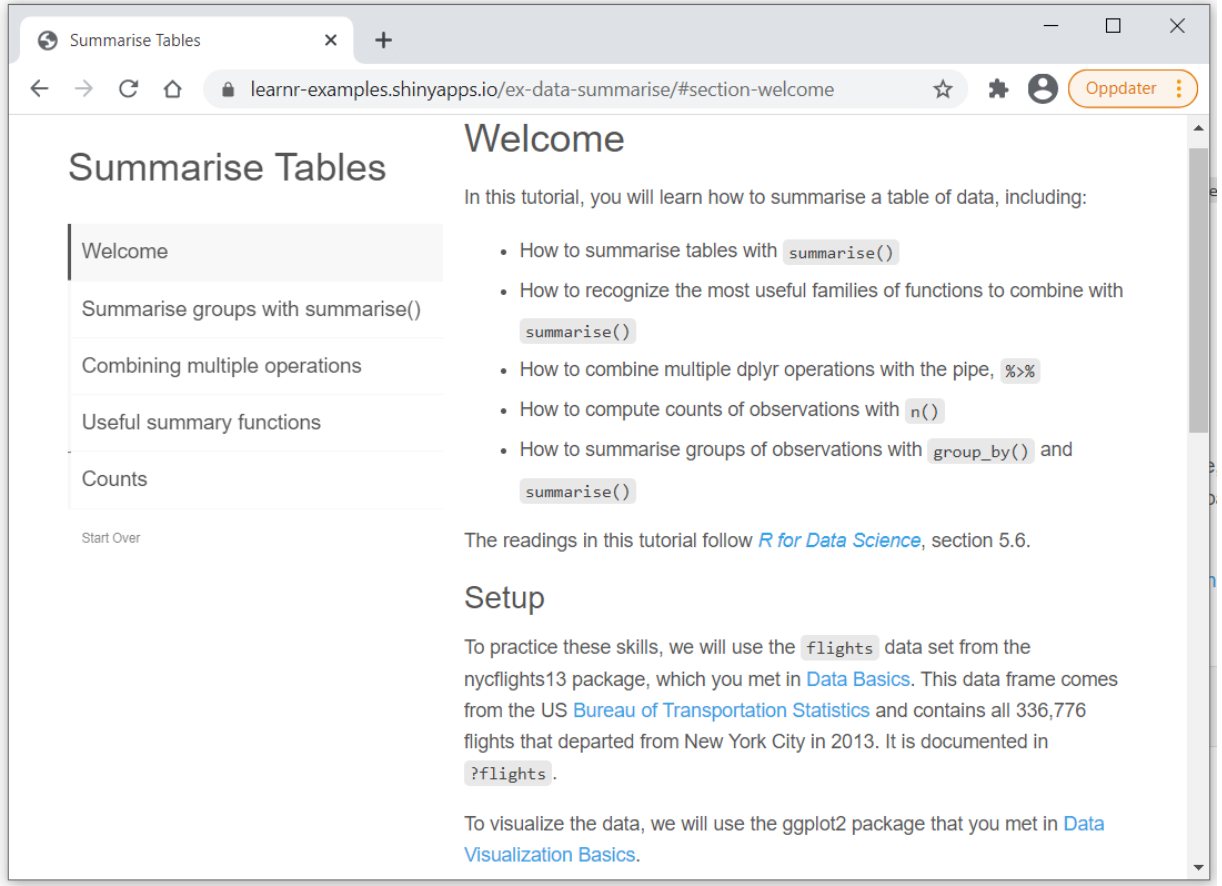
Leaflet + shiny example

https://statisticsnorway.shinyapps.io/pendling/

https://shiny.rstudio.com/gallery/

learnr

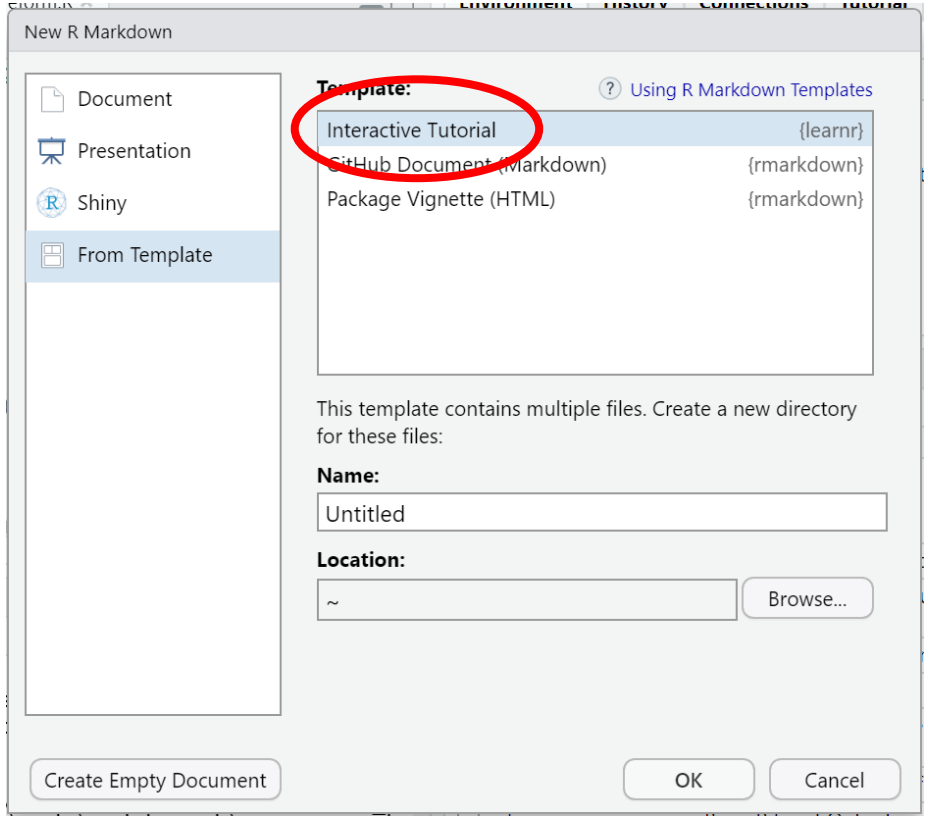
- Create online interactive courses in R
- Uses Rmarkdown
- Add examples, pictures, videos and exercises
- https://rstudio.github.io/learnr/



Statistisk sentralbyrå
Statistics Norway

How to create a learnr tutorial

- File > New File > R Markdown …



Publish as a shinyapp when finished
www.shinyapps.io/

Statistisk sentralbyrå
Statistics Norway

learnr demo

https://ssbviz.shinyapps.io/ApiData/

Discussion