



Introduction course

SUSIE JENTOFT, ASLAUG FOSS

DAY 1: 14. DECEMBER 2020



Statistisk sentralbyrå
Statistics Norway

Goals: Part 1

- Gain an understanding for R and RStudio
- Open RStudio og run calculations
- Read in and manipulate data
- Create tables and summary statistics
- Create figures
- Edit and impute data



Agenda: Part 1

	Monday 14th December	Tuesday 16th December	Wednesday 17th December
12:00	<ul style="list-style-type: none"> • Introduction • Github • Basic calculations • Objects 	<ul style="list-style-type: none"> • Review • Data manipulation 	<ul style="list-style-type: none"> • Review • Sampling • Outlier detection
12:45	Exercise 1 Break	Exercise 3 Break	Exercise 5 Break
13:30	<ul style="list-style-type: none"> • Logical statements • Read in data 	<ul style="list-style-type: none"> • Merging datasets • Plotting 	<ul style="list-style-type: none"> • Imputation
14:15	Exercise 2	Exercise 4	Exercise 6
14:50 – 15:00			Summary



Part 2 (January)

- Generelle programming
- Functions and packages
- Creating documents and visualisation
- Testing and sharing code



Course format and guidelines

- **Use chat** for asking questions
- **Ask questions** if something is unclear (raise hand)
- **Exercises:** Write or ask questions to us and eachother.
- Code for course is on **GitHub**
- **Mute microphone** when you are not speaking
- **Turn on video** so we can see eachother (off when you are on a break)





Susie Jentoft



Aslaug Foss

What is ?

- Programming language and environment for statistics
- Developed by Ross Ihaka & Robert Gentleman (1993)
- Base + Packages
- RStudio:
 - IDE/development environment
 - Open source + commercial licenses

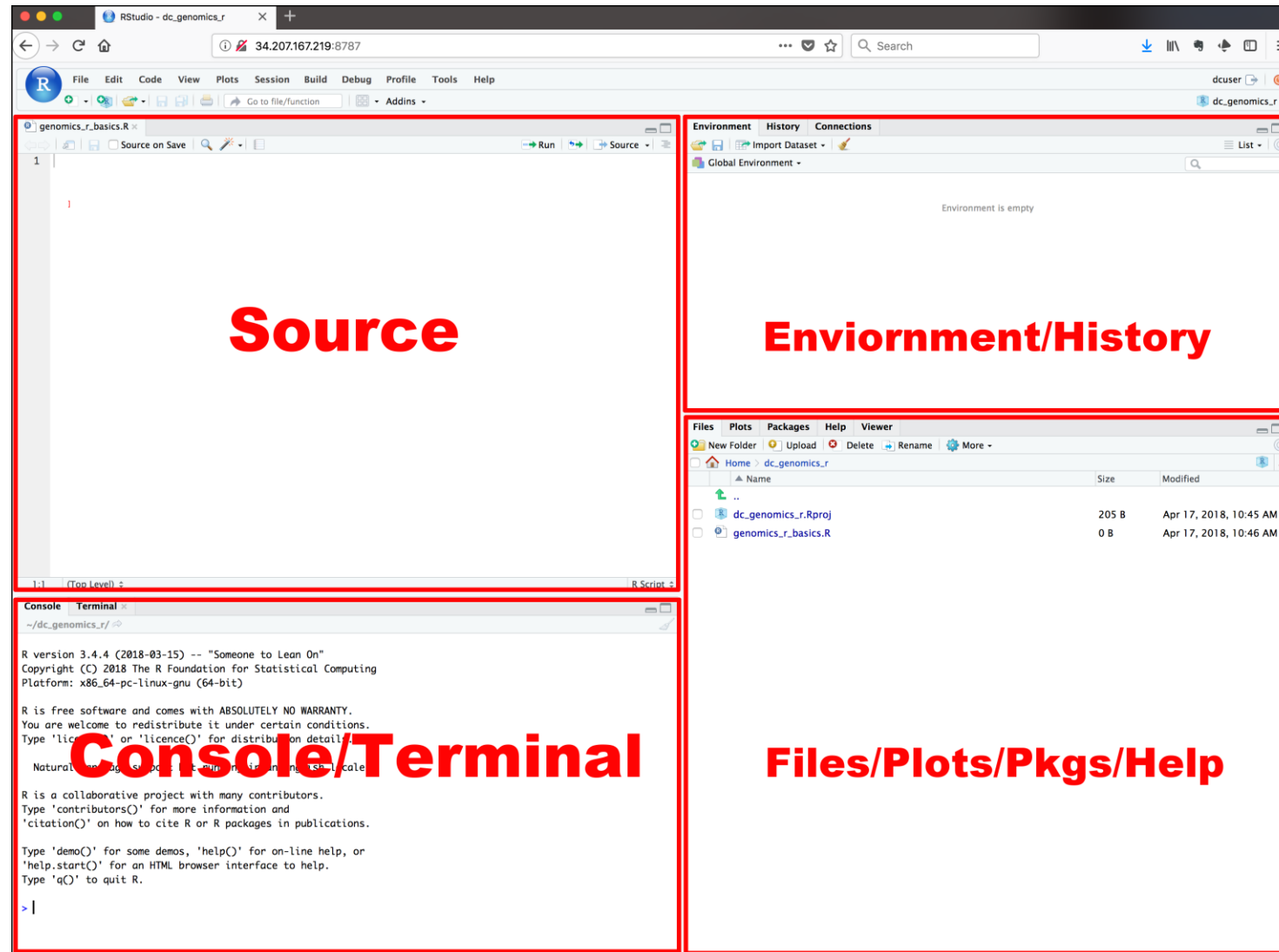


Why learn ?

- Open source and free
- Large society and support
- Good graphics
- Developed specifically for statistics
- Used within many statistical bureaus:
 - <https://github.com/SNStatComp/awesome-official-statistics-software>







Working with code files

Create a new file

- File > New File > R script

or



Open an existing file

- File > Open File ...

or





Save file

- File > Save

or

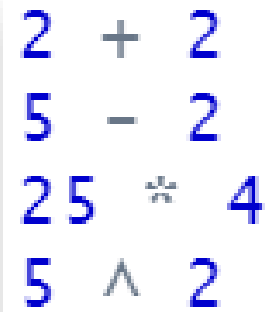


Running code

- Write code in source/code files
- Run code by clicking on  **Run** or ctrl + enter
 - This will run the line your mouse is on
 - Highlight an area to run a segment or several lines
 - Also possible to run whole code with  **Source** ▾
- Lines that start with # are for comments (not run)
- Use 4x# to create headings (#### Heading ####)

Basic operations

- Can be used as a calculator



$2 + 2$
 $5 - 2$
 $25 \div 4$
 $5 \wedge 2$



Base functions

Log and exponential :

Functions

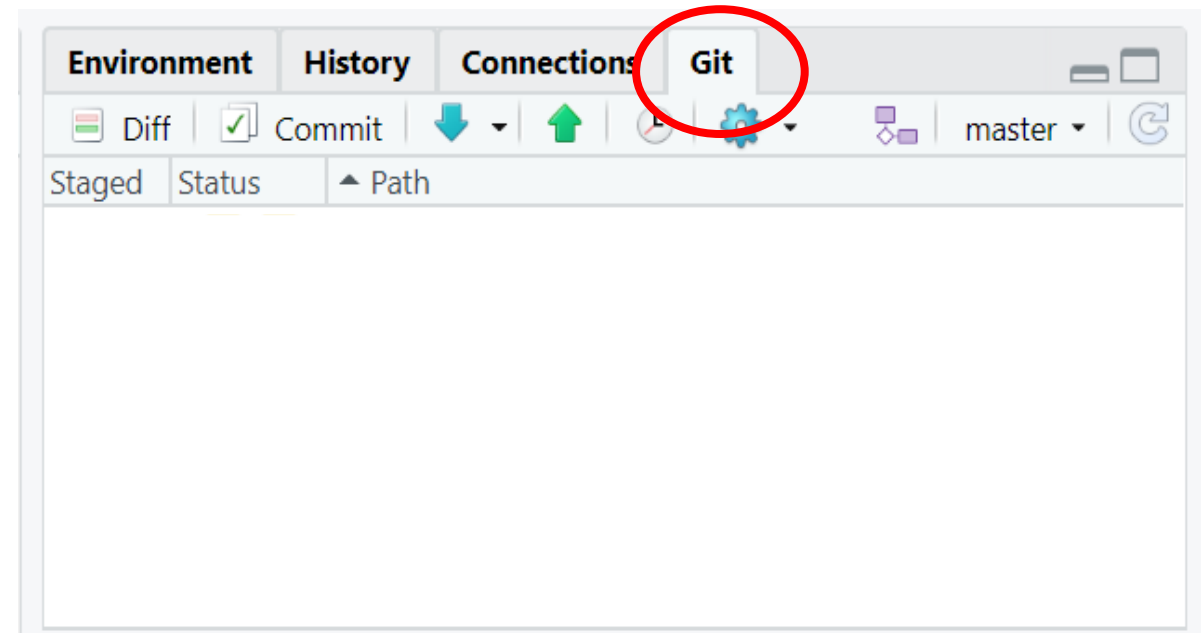
```
log(10)  
exp(2)
```

All functions in R have help files:

```
?log
```

Git and GitHub

- Git: version control for code
- Distribute code and manage projects with several contributors
- Git is well integrated in RStudio



Clone a repository

- **Github:** https://github.com/statisticsnorway/R_introduction_Ukraine

- Click on



- Copy HTTPS address (URL)

- **In RStudio:**

- File > New project > Version control > Git

- Paste in address under «Repository URL»

- Click Create Project

- **Save files** you change (for example exercises) with a new name



Statistisk sentralbyrå
Statistics Norway



Example code for today

Example code for today is called: **Rcode_day1.R**



Strings

```
"Hello, world!"
```

- Single (' ') or double quotes (" ") – be consistent!
- CASE SENSITIVE
- Combine two or more with
- Take out part of a string:

```
paste()
```

```
substr(string, start, stopp)
```

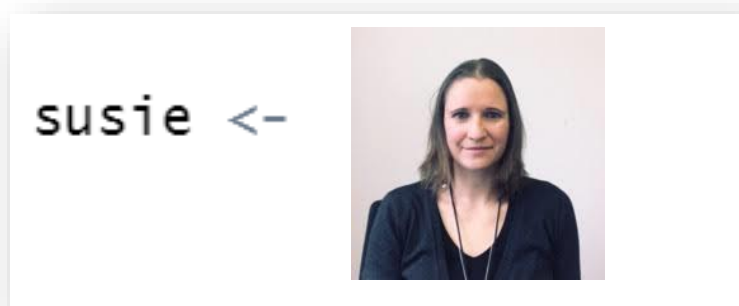
- Run to print or use

```
print()
```



Objects

- Creating an objects gives a name to a value/or string
- We use `<-` to allocate (give a name to) an object



- To see what is stored in the object, write and run the name

Objects

- CASE SENSITIVE
- Must start with a letter
- No spaces (use _)
- Different contents: number, string
- Can be written over/replaced

Oblast ≠ oblast

susie <-



Object type

- Find out type with

```
str()
```

Put object name inside brackets

- Common object types:

Object type	Description
chr	Character/string
num	Number
Date	Date
Factor	Categorical variable (fixed levels)
data.frame	Dataset

Vectors

- Objects which hold several values
- Use `c()` to create vector (combine)
- Use comma to separate elements
- Must be same object type

```
c(1, 7, 10)
```

```
c("Kongsvinger", "Oslo")
```



Vectors as objects

- Give vectors a name with

```
<-
```

- Find the length of a vector with

```
length()
```

- Determine the object type with

```
str()
```



Statistisk sentralbyrå
Statistics Norway

Exercise 1

- Clone repository for the course
- Open file Exercises_day1.R and do exercise 1



Logical statements/tests

- Compare objects
 - TRUE/FALSE

Code i R	Description
==	Compare if the same
!=	Compare if they are different
>	Greater than
>=	Greater than or equal
<	Less than
<=	Less than or equal
%in%	Is in/contains

- Can be used for single values or vectors
- Combine with **&** (and), **|** (or)

ifelse

- Compare and then do something

```
ifelse()
```

```
ifelse(location == "Kyiv", "capital", "other")
```

condition

if condition is TRUE: do this

If condition is FALSE: do this

Create a dataset

```
data.frame()
```

- Objects can also be a dataset (with rows and columns)
- Combine different types of data (numbers and strings)

```
data.frame(object1, object2)
```

```
data.frame(column_name1 = c(1, 2), column_name2 = c("Industry", "Agriculture"))
```



Look at the dataset

- Write dataset name
- Click on dataset name under Environment
- Or



View()

R Packages

- Collection of code and functions
- CRAN (www.r-project.org)
- Install package one time

```
install.packages("package_name")
```

- Load package each time RStudio starts

```
library(package_name)
```



Read in dataset: .csv file

```
library(tidyverse)  
read_csv()
```

Path to dataset & name

separator = , decimal = .

```
library(tidyverse)  
read_csv2()
```

Path to dataset & name

separator = ; decimal = ,



Read in dataset: stata file

- We can read in stata datasets direct to R using

```
library(haven)  
read_dta()
```

Path to dataset & name

Look at the data

```
glimpse(dataset)
```

Variables and type

```
summary(dataset)
```

Average, min, max, osv

```
names(dataset)
```

Variable names



Exercise 2:

- Exercise 2 is in the file : **Exercises_day1.R**

