Nordsjællands Hospital
Neurologisk afdeling

Introduktion til R/R-studio

Jacob Mesot Liljehult
Klinisk sygeplejespecialist
cand.scient.san, PhD

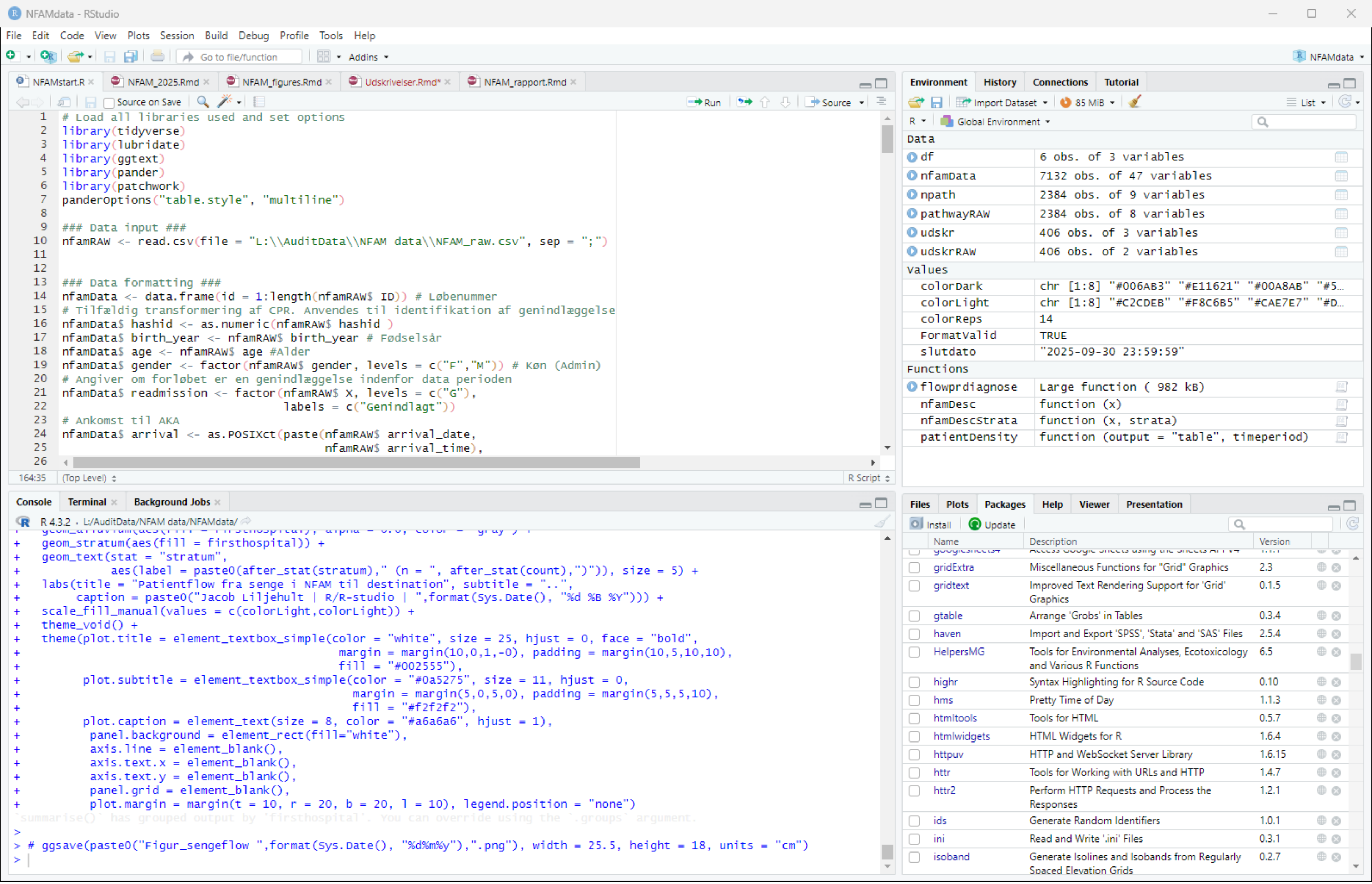
Hvorfor vælge R/RStudio

Fordele

- Open-source og gratis (kræver ikke licens)
- Meget fleksibelt – alle analyser er tilgængelige i et program
- Simplere kodning end i SAS
- Let at få hjælp – mange ressourcer tilgængelige

Ulemper

- Man skal lære at kode
- Kodningen kan være inkonsistent

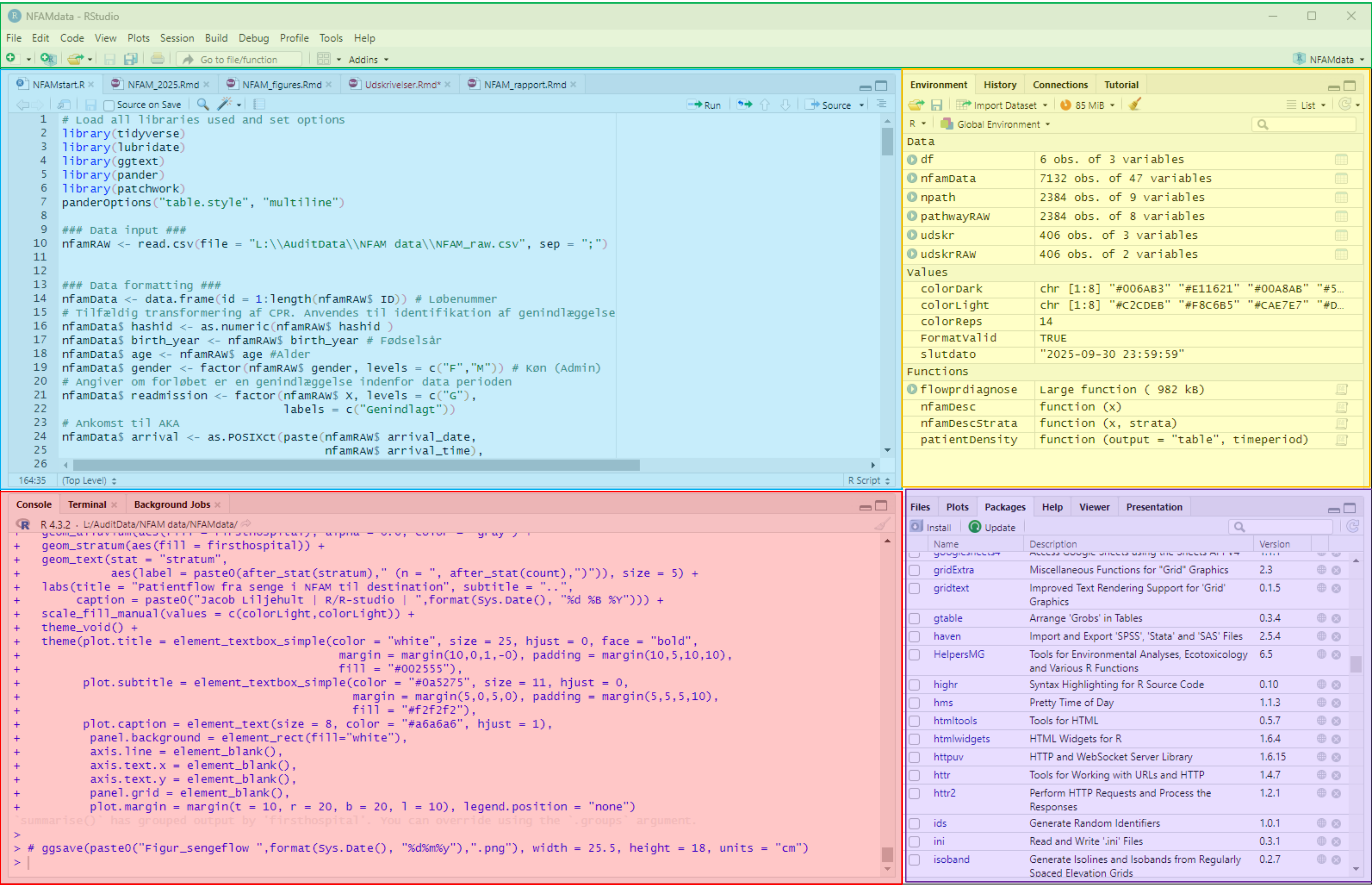


```r
# Load all libraries used and set options
library(tidyverse)
library(lubridate)
library(ggtext)
library(pander)
library(patchwork)
panderOptions("table.style", "multiline")

### Data input ###
nfamRAW <- read.csv(file = "L:\\AuditData\\NFAM data\\NFAM_raw.csv", sep = ";")


### Data formatting ###
nfamData <- data.frame(id = 1:length(nfamRAW$ ID)) # Løbenummer
# Tilfældig transformering af CPR. Anvendes til identifikation af genindlæggelse
nfamData$ hashid <- as.numeric(nfamRAW$ hashid )
nfamData$ birth_year <- nfamRAW$ birth_year # Fødselsår
nfamData$ age <- nfamRAW$ age #Alder
nfamData$ gender <- factor(nfamRAW$ gender, levels = c("F","M")) # Køn (Admin)
# Angiver om forløbet er en genindlæggelse indenfor data perioden
nfamData$ readmission <- factor(nfamRAW$ X, levels = c("G"),
                                labels = c("Genindlagt"))
# Ankomst til AKA
nfamData$ arrival <- as.POSIXct(paste(nfamRAW$ arrival_date,
                                      nfamRAW$ arrival_time),
```

```
+   geom_stratum(aes(fill = firsthospital)) +
+   geom_text(stat = "stratum",
+             aes(label = paste0(after_stat(stratum)," (n = ", after_stat(count),")")), size = 5) +
+   labs(title = "Patientflow fra senge i NFAM til destination", subtitle = "..",
+        caption = paste0("Jacob Liljehult | R/R-studio | ",format(Sys.Date(), "%d %B %Y"))) +
+   scale_fill_manual(values = c(colorLight,colorLight)) +
+   theme_void() +
+   theme(plot.title = element_textbox_simple(color = "white", size = 25, hjust = 0, face = "bold",
+                                             margin = margin(10,0,1,-0), padding = margin(10,5,10,10),
+                                             fill = "#002555"),
+         plot.subtitle = element_textbox_simple(color = "#0a5275", size = 11, hjust = 0,
+                                               margin = margin(5,0,5,0), padding = margin(5,5,5,10),
+                                               fill = "#f2f2f2"),
+         plot.caption = element_text(size = 8, color = "#a6a6a6", hjust = 1),
+          panel.background = element_rect(fill="white"),
+          axis.line = element_blank(),
+          axis.text.x = element_blank(),
+          axis.text.y = element_blank(),
+          panel.grid = element_blank(),
+          plot.margin = margin(t = 10, r = 20, b = 20, l = 10), legend.position = "none")
`summarise()` has grouped output by 'firsthospital'. You can override using the `.groups` argument.
>
> # ggsave(paste0("Figur_sengeflow ",format(Sys.Date(), "%d%m%y"),".png"), width = 25.5, height = 18, units = "cm")
> 
```

Data	
df	6 obs. of 3 variables
nfamData	7132 obs. of 47 variables
npath	2384 obs. of 9 variables
pathwayRAW	2384 obs. of 8 variables
udskr	406 obs. of 3 variables
udskrRAW	406 obs. of 2 variables
Values	
colorDark	chr [1:8] "#006AB3" "#E11621" "#00A8AB" "#5…
colorLight	chr [1:8] "#C2CDEB" "#F8C6B5" "#CAE7E7" "#D…
colorReps	14
FormatValid	TRUE
slutdato	"2025-09-30 23:59:59"
Functions	
flowprdiagnose	Large function (982 kB)
nfamDesc	function (x)
nfamDescStrata	function (x, strata)
patientDensity	function (output = "table", timeperiod)

	Name	Description	Version
☐	gridExtra	Miscellaneous Functions for "Grid" Graphics	2.3
☐	gridtext	Improved Text Rendering Support for 'Grid' Graphics	0.1.5
☐	gtable	Arrange 'Grobs' in Tables	0.3.4
☐	haven	Import and Export 'SPSS', 'Stata' and 'SAS' Files	2.5.4
☐	HelpersMG	Tools for Environmental Analyses, Ecotoxicology and Various R Functions	6.5
☐	highr	Syntax Highlighting for R Source Code	0.10
☐	hms	Pretty Time of Day	1.1.3
☐	htmltools	Tools for HTML	0.5.7
☐	htmlwidgets	HTML Widgets for R	1.6.4
☐	httpuv	HTTP and WebSocket Server Library	1.6.15
☐	httr	Tools for Working with URLs and HTTP	1.4.7
☐	httr2	Perform HTTP Requests and Process the Responses	1.2.1
☐	ids	Generate Random Identifiers	1.0.1
☐	ini	Read and Write '.ini' Files	0.3.1
☐	isoband	Generate Isolines and Isobands from Regularly Spaced Elevation Grids	0.2.7

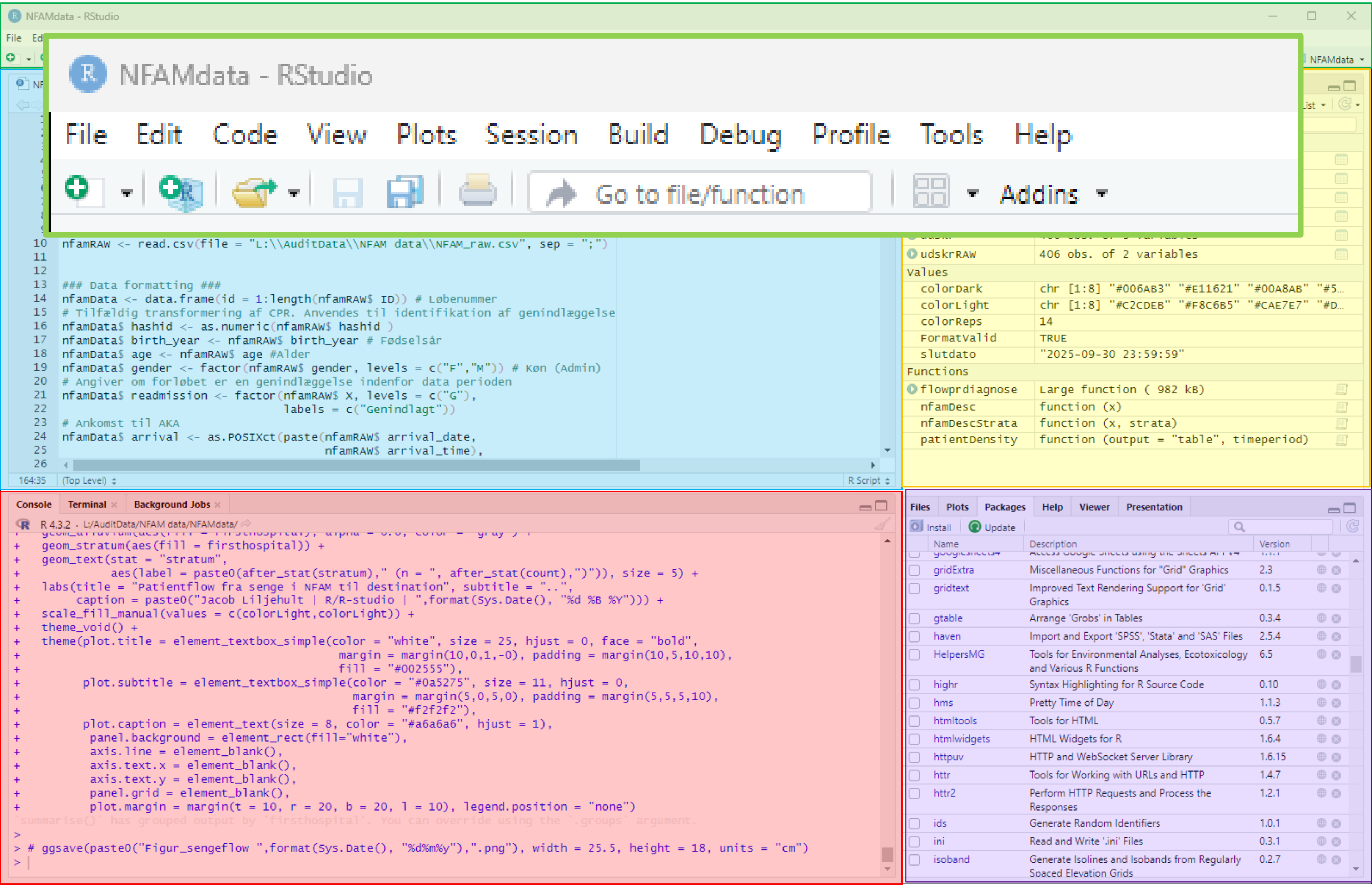


```r
# Load all libraries used and set options
library(tidyverse)
library(lubridate)
library(ggtext)
library(pander)
library(patchwork)
panderOptions("table.style", "multiline")

### Data input ###
nfamRAW <- read.csv(file = "L:\\AuditData\\NFAM data\\NFAM_raw.csv", sep = ";")


### Data formatting ###
nfamData <- data.frame(id = 1:length(nfamRAW$ ID)) # Løbenummer
# Tilfældig transformering af CPR. Anvendes til identifikation af genindlæggelse
nfamData$ hashid <- as.numeric(nfamRAW$ hashid )
nfamData$ birth_year <- nfamRAW$ birth_year # Fødselsår
nfamData$ age <- nfamRAW$ age #Alder
nfamData$ gender <- factor(nfamRAW$ gender, levels = c("F","M")) # Køn (Admin)
# Angiver om forløbet er en genindlæggelse indenfor data perioden
nfamData$ readmission <- factor(nfamRAW$ X, levels = c("G"),
                                labels = c("Genindlagt"))
# Ankomst til AKA
nfamData$ arrival <- as.POSIXct(paste(nfamRAW$ arrival_date,
                                      nfamRAW$ arrival_time),

```

Data	
df	6 obs. of 3 variables
nfamData	7132 obs. of 47 variables
npath	2384 obs. of 9 variables
pathwayRAW	2384 obs. of 8 variables
udskr	406 obs. of 3 variables
udskrRAW	406 obs. of 2 variables
Values	
colorDark	chr [1:8] "#006AB3" "#E11621" "#00A8AB" "#5…
colorLight	chr [1:8] "#C2CDEB" "#F8C6B5" "#CAE7E7" "#D…
colorReps	14
FormatValid	TRUE
slutdato	"2025-09-30 23:59:59"
Functions	
flowprdiagnose	Large function (982 kB)
nfamDesc	function (x)
nfamDescStrata	function (x, strata)
patientDensity	function (output = "table", timeperiod)

```
+   geom_stratum(aes(fill = firsthospital)) +
+   geom_text(stat = "stratum",
+             aes(label = paste0(after_stat(stratum)," (n = ", after_stat(count),")")), size = 5) +
+   labs(title = "Patientflow fra senge i NFAM til destination", subtitle = "..",
+        caption = paste0("Jacob Liljehult | R/R-studio | ",format(Sys.Date(), "%d %B %Y"))) +
+   scale_fill_manual(values = c(colorLight,colorLight)) +
+   theme_void() +
+   theme(plot.title = element_textbox_simple(color = "white", size = 25, hjust = 0, face = "bold",
+                                             margin = margin(10,0,1,-0), padding = margin(10,5,10,10),
+                                             fill = "#002555"),
+         plot.subtitle = element_textbox_simple(color = "#0a5275", size = 11, hjust = 0,
+                                                margin = margin(5,0,5,0), padding = margin(5,5,5,10),
+                                                fill = "#f2f2f2"),
+         plot.caption = element_text(size = 8, color = "#a6a6a6", hjust = 1),
+          panel.background = element_rect(fill="white"),
+          axis.line = element_blank(),
+          axis.text.x = element_blank(),
+          axis.text.y = element_blank(),
+          panel.grid = element_blank(),
+          plot.margin = margin(t = 10, r = 20, b = 20, l = 10), legend.position = "none")
`summarise()` has grouped output by 'firsthospital'. You can override using the `.groups` argument.
>
> # ggsave(paste0("Figur_sengeflow ",format(Sys.Date(), "%d%m%y"),".png"), width = 25.5, height = 18, units = "cm")
> |
```

	Name	Description	Version
☐	gridExtra	Miscellaneous Functions for "Grid" Graphics	2.3
☐	gridtext	Improved Text Rendering Support for 'Grid' Graphics	0.1.5
☐	gtable	Arrange 'Grobs' in Tables	0.3.4
☐	haven	Import and Export 'SPSS', 'Stata' and 'SAS' Files	2.5.4
☐	HelpersMG	Tools for Environmental Analyses, Ecotoxicology and Various R Functions	6.5
☐	highr	Syntax Highlighting for R Source Code	0.10
☐	hms	Pretty Time of Day	1.1.3
☐	htmltools	Tools for HTML	0.5.7
☐	htmlwidgets	HTML Widgets for R	1.6.4
☐	httpuv	HTTP and WebSocket Server Library	1.6.15
☐	httr	Tools for Working with URLs and HTTP	1.4.7
☐	httr2	Perform HTTP Requests and Process the Responses	1.2.1
☐	ids	Generate Random Identifiers	1.0.1
☐	ini	Read and Write '.ini' Files	0.3.1
☐	isoband	Generate Isolines and Isobands from Regularly Spaced Elevation Grids	0.2.7



```r
nfamRAW <- read.csv(file = "L:\\AuditData\\NFAM data\\NFAM_raw.csv", sep = ";")


### Data formatting ###
nfamData <- data.frame(id = 1:length(nfamRAW$ ID)) # Løbenummer
# Tilfældig transformering af CPR. Anvendes til identifikation af genindlæggelse
nfamData$ hashid <- as.numeric(nfamRAW$ hashid )
nfamData$ birth_year <- nfamRAW$ birth_year # Fødselsår
nfamData$ age <- nfamRAW$ age #Alder
nfamData$ gender <- factor(nfamRAW$ gender, levels = c("F","M")) # Køn (Admin)
# Angiver om forløbet er en genindlæggelse indenfor data perioden
nfamData$ readmission <- factor(nfamRAW$ X, levels = c("G"),
                                labels = c("Genindlagt"))
# Ankomst til AKA
nfamData$ arrival <- as.POSIXct(paste(nfamRAW$ arrival_date,
                                      nfamRAW$ arrival_time),

```

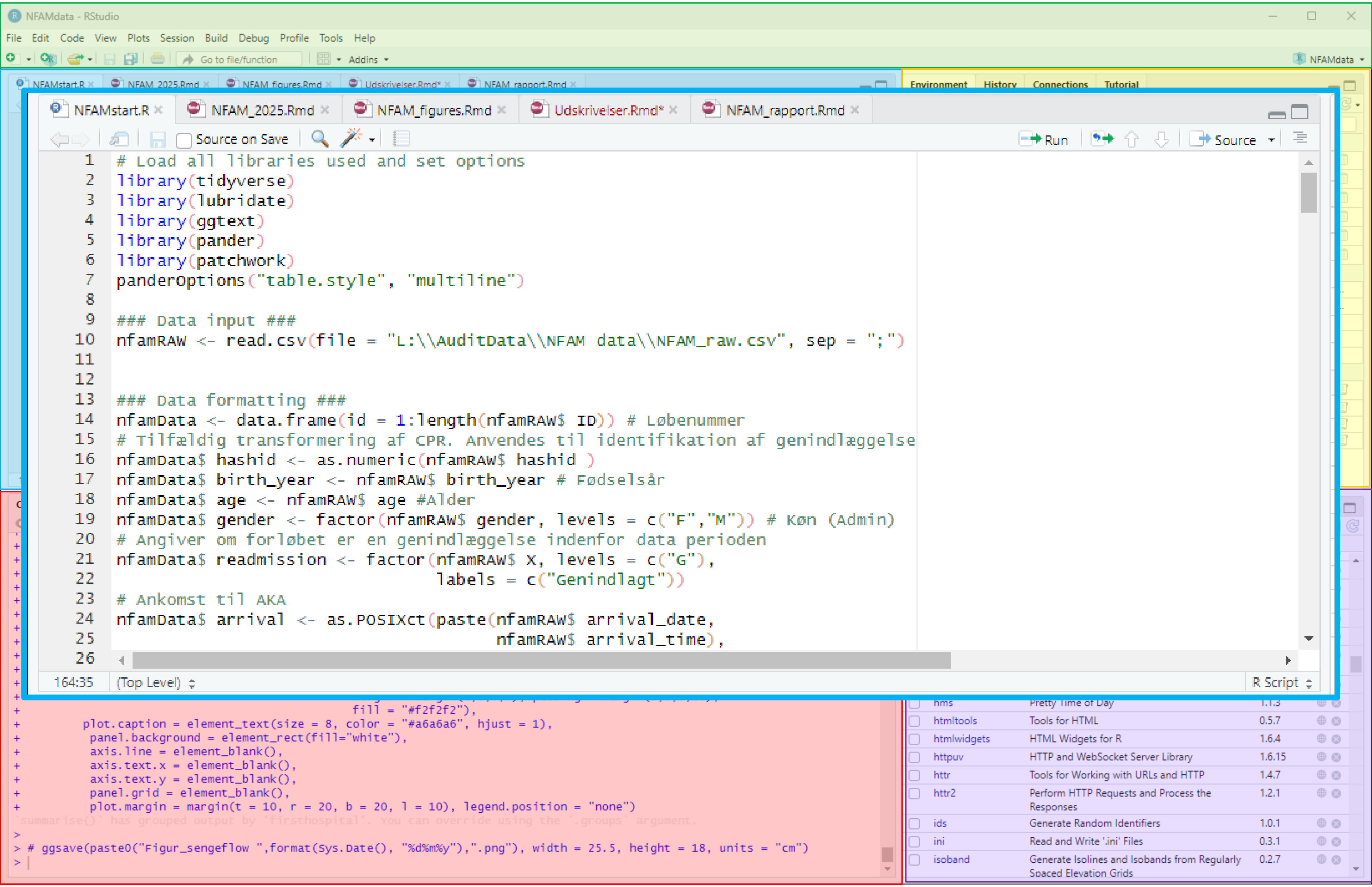
164:35

udskrRAW	406 obs. of 2 variables
Values	
colorDark	chr [1:8] "#006AB3" "#E11621" "#00A8AB" "#5…
colorLight	chr [1:8] "#C2CDEB" "#F8C6B5" "#CAE7E7" "#D…
colorReps	14
FormatValid	TRUE
slutdato	"2025-09-30 23:59:59"
Functions	
flowprdiagnose	Large function (982 kB)
nfamDesc	function (x)
nfamDescStrata	function (x, strata)
patientDensity	function (output = "table", timeperiod)

R 4.3.2 · L:/AuditData/NFAM data/NFAMdata/

```
+   geom_stratum(aes(fill = firsthospital)) +
+   geom_text(stat = "stratum",
+             aes(label = paste0(after_stat(stratum)," (n = ", after_stat(count),")")), size = 5) +
+   labs(title = "Patientflow fra senge i NFAM til destination", subtitle = "..",
+        caption = paste0("Jacob Liljehult | R/R-studio | ",format(Sys.Date(), "%d %B %Y"))) +
+   scale_fill_manual(values = c(colorLight,colorLight)) +
+   theme_void() +
+   theme(plot.title = element_textbox_simple(color = "white", size = 25, hjust = 0, face = "bold",
+                                             margin = margin(10,0,1,-0), padding = margin(10,5,10,10),
+                                             fill = "#002555"),
+         plot.subtitle = element_textbox_simple(color = "#0a5275", size = 11, hjust = 0,
+                                               margin = margin(5,0,5,0), padding = margin(5,5,5,10),
+                                               fill = "#f2f2f2"),
+         plot.caption = element_text(size = 8, color = "#a6a6a6", hjust = 1),
+          panel.background = element_rect(fill="white"),
+          axis.line = element_blank(),
+          axis.text.x = element_blank(),
+          axis.text.y = element_blank(),
+          panel.grid = element_blank(),
+          plot.margin = margin(t = 10, r = 20, b = 20, l = 10), legend.position = "none")
`summarise()` has grouped output by 'firsthospital'. You can override using the `.groups` argument.
>
> # ggsave(paste0("Figur_sengeflow ",format(Sys.Date(), "%d%m%y"),".png"), width = 25.5, height = 18, units = "cm")
> |
```

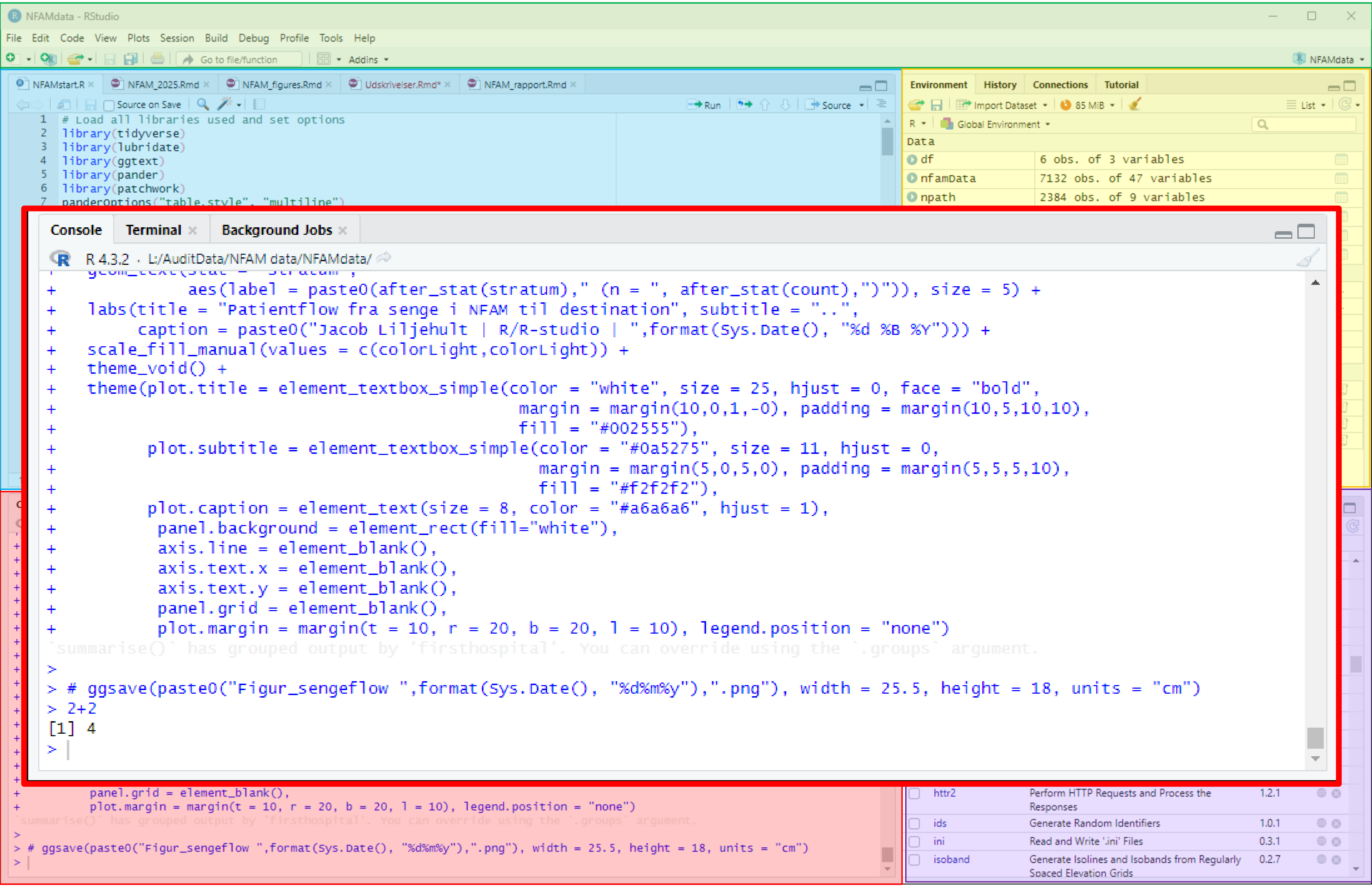
	Name	Description	Version
☐	gridExtra	Miscellaneous Functions for "Grid" Graphics	2.3
☐	gridtext	Improved Text Rendering Support for 'Grid' Graphics	0.1.5
☐	gtable	Arrange 'Grobs' in Tables	0.3.4
☐	haven	Import and Export 'SPSS', 'Stata' and 'SAS' Files	2.5.4
☐	HelpersMG	Tools for Environmental Analyses, Ecotoxicology and Various R Functions	6.5
☐	highr	Syntax Highlighting for R Source Code	0.10
☐	hms	Pretty Time of Day	1.1.3
☐	htmltools	Tools for HTML	0.5.7
☐	htmlwidgets	HTML Widgets for R	1.6.4
☐	httpuv	HTTP and WebSocket Server Library	1.6.15
☐	httr	Tools for Working with URLs and HTTP	1.4.7
☐	httr2	Perform HTTP Requests and Process the Responses	1.2.1
☐	ids	Generate Random Identifiers	1.0.1
☐	ini	Read and Write '.ini' Files	0.3.1
☐	isoband	Generate Isolines and Isobands from Regularly Spaced Elevation Grids	0.2.7



```r
# Load all libraries used and set options
library(tidyverse)
library(lubridate)
library(ggtext)
library(pander)
library(patchwork)
panderOptions("table.style", "multiline")

### Data input ###
nfamRAW <- read.csv(file = "L:\\AuditData\\NFAM data\\NFAM_raw.csv", sep = ";")


### Data formatting ###
nfamData <- data.frame(id = 1:length(nfamRAW$ ID)) # Løbenummer
# Tilfældig transformering af CPR. Anvendes til identifikation af genindlæggelse
nfamData$ hashid <- as.numeric(nfamRAW$ hashid )
nfamData$ birth_year <- nfamRAW$ birth_year # Fødselsår
nfamData$ age <- nfamRAW$ age #Alder
nfamData$ gender <- factor(nfamRAW$ gender, levels = c("F","M")) # Køn (Admin)
# Angiver om forløbet er en genindlæggelse indenfor data perioden
nfamData$ readmission <- factor(nfamRAW$ X, levels = c("G"),
                                labels = c("Genindlagt"))
# Ankomst til AKA
nfamData$ arrival <- as.POSIXct(paste(nfamRAW$ arrival_date,
                                      nfamRAW$ arrival_time),
```

```
+                                fill = "#f2f2f2"),
+       plot.caption = element_text(size = 8, color = "#a6a6a6", hjust = 1),
+        panel.background = element_rect(fill="white"),
+        axis.line = element_blank(),
+        axis.text.x = element_blank(),
+        axis.text.y = element_blank(),
+        panel.grid = element_blank(),
+        plot.margin = margin(t = 10, r = 20, b = 20, l = 10), legend.position = "none")
`summarise()` has grouped output by 'firsthospital'. You can override using the `.groups` argument.
>
> # ggsave(paste0("Figur_sengeflow ",format(Sys.Date(), "%d%m%y"),".png"), width = 25.5, height = 18, units = "cm")
> |
```

☐	hms	Pretty Time of Day	1.1.3
☐	htmltools	Tools for HTML	0.5.7
☐	htmlwidgets	HTML Widgets for R	1.6.4
☐	httpuv	HTTP and WebSocket Server Library	1.6.15
☐	httr	Tools for Working with URLs and HTTP	1.4.7
☐	httr2	Perform HTTP Requests and Process the Responses	1.2.1
☐	ids	Generate Random Identifiers	1.0.1
☐	ini	Read and Write '.ini' Files	0.3.1
☐	isoband	Generate Isolines and Isobands from Regularly Spaced Elevation Grids	0.2.7



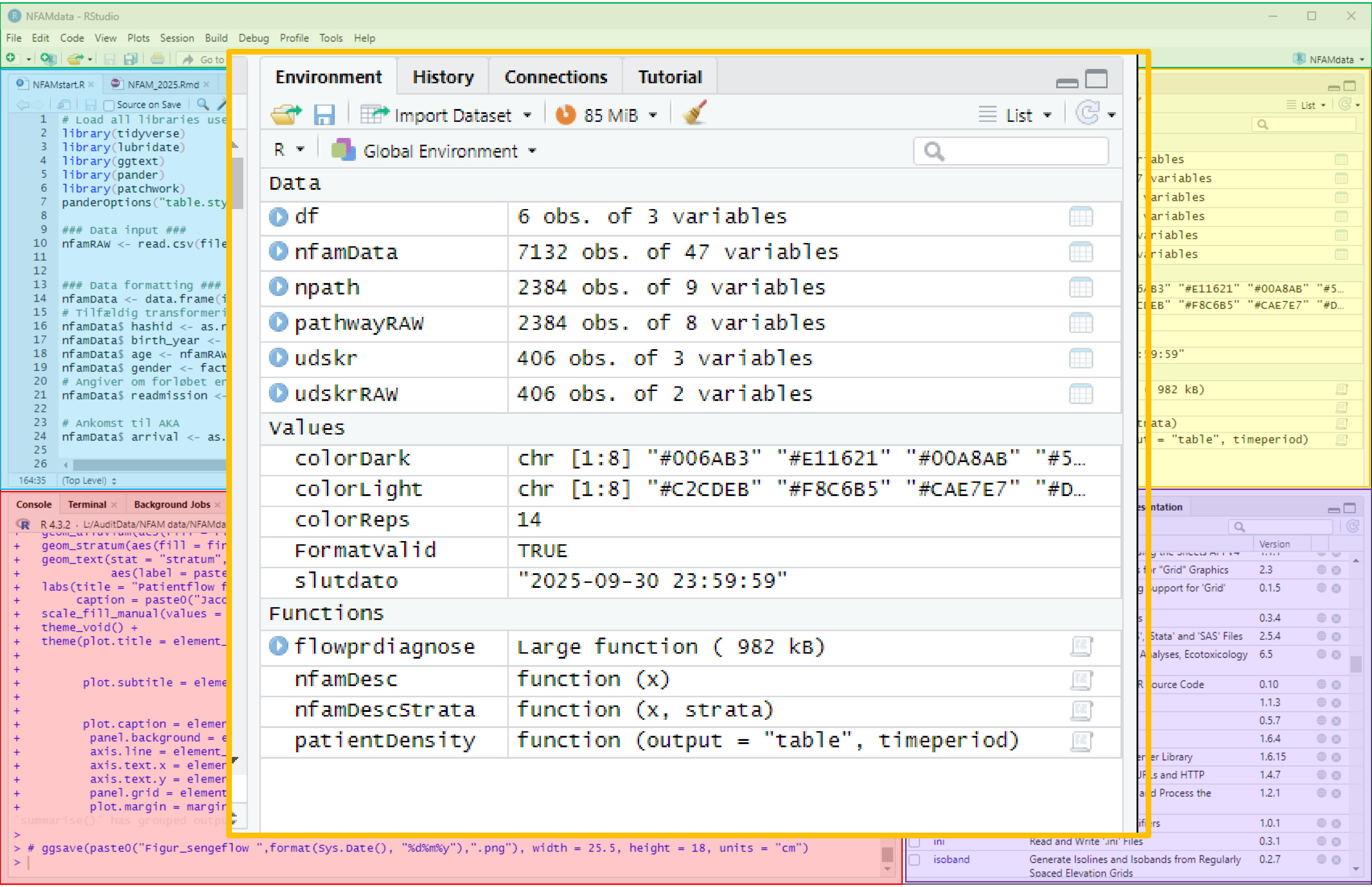
```
# Load all libraries used and set options
library(tidyverse)
library(lubridate)
library(ggtext)
library(pander)
library(patchwork)
panderOptions("table.style", "multiline")
```

Data

df	6 obs. of 3 variables
nfamData	7132 obs. of 47 variables
npath	2384 obs. of 9 variables

```
+   aes(label = paste0(after_stat(stratum)," (n = ", after_stat(count),")")), size = 5) +
+   labs(title = "Patientflow fra senge i NFAM til destination", subtitle = "..",
+        caption = paste0("Jacob Liljehult | R/R-studio | ",format(Sys.Date(), "%d %B %Y"))) +
+   scale_fill_manual(values = c(colorLight,colorLight)) +
+   theme_void() +
+   theme(plot.title = element_textbox_simple(color = "white", size = 25, hjust = 0, face = "bold",
+                                             margin = margin(10,0,1,-0), padding = margin(10,5,10,10),
+                                             fill = "#002555"),
+         plot.subtitle = element_textbox_simple(color = "#0a5275", size = 11, hjust = 0,
+                                               margin = margin(5,0,5,0), padding = margin(5,5,5,10),
+                                               fill = "#f2f2f2"),
+         plot.caption = element_text(size = 8, color = "#a6a6a6", hjust = 1),
+          panel.background = element_rect(fill="white"),
+          axis.line = element_blank(),
+          axis.text.x = element_blank(),
+          axis.text.y = element_blank(),
+          panel.grid = element_blank(),
+          plot.margin = margin(t = 10, r = 20, b = 20, l = 10), legend.position = "none")
`summarise()` has grouped output by 'firsthospital'. You can override using the `.groups` argument.
>
> # ggsave(paste0("Figur_sengeflow ",format(Sys.Date(), "%d%m%y"),".png"), width = 25.5, height = 18, units = "cm")
> 2+2
[1] 4
>
```

httr2	Perform HTTP Requests and Process the Responses	1.2.1
ids	Generate Random Identifiers	1.0.1
ini	Read and Write '.ini' Files	0.3.1
isoband	Generate Isolines and Isobands from Regularly Spaced Elevation Grids	0.2.7



```
# Load all libraries use
library(tidyverse)
library(lubridate)
library(ggtext)
library(pander)
library(patchwork)
panderOptions("table.sty

### Data input ###
nfamRAW <- read.csv(file


### Data formatting ###
nfamData <- data.frame(i
# Tilfældig transformeri
nfamData$ hashid <- as.r
nfamData$ birth_year <-
nfamData$ age <- nfamRAW
nfamData$ gender <- fact
# Angiver om forløbet er
nfamData$ readmission <-

# Ankomst til AKA
nfamData$ arrival <- as.
```

Data

df	6 obs. of 3 variables
nfamData	7132 obs. of 47 variables
npath	2384 obs. of 9 variables
pathwayRAW	2384 obs. of 8 variables
udskr	406 obs. of 3 variables
udskrRAW	406 obs. of 2 variables

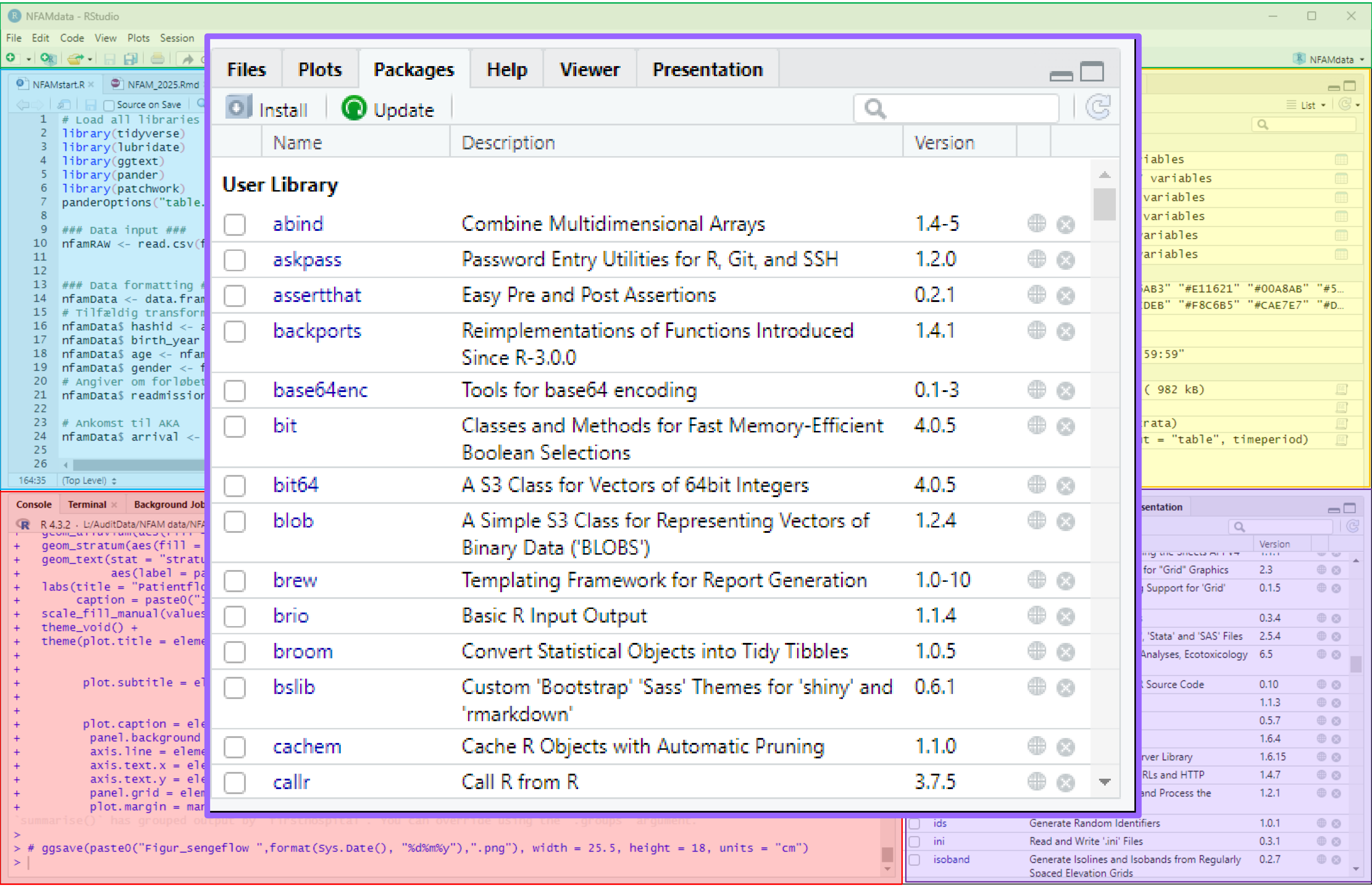
Values

colorDark	chr [1:8] "#006AB3" "#E11621" "#00A8AB" "#5…
colorLight	chr [1:8] "#C2CDEB" "#F8C6B5" "#CAE7E7" "#D…
colorReps	14
FormatValid	TRUE
slutdato	"2025-09-30 23:59:59"

Functions

flowprdiagnose	Large function (982 kB)
nfamDesc	function (x)
nfamDescStrata	function (x, strata)
patientDensity	function (output = "table", timeperiod)

```
+   geom_stratum(aes(fill = fir
+   geom_text(stat = "stratum",
+             aes(label = paste
+   labs(title = "Patientflow f
+        caption = paste0("Jacc
+   scale_fill_manual(values =
+   theme_void() +
+   theme(plot.title = element_
+
+
+         plot.subtitle = eleme
+
+
+         plot.caption = elemer
+          panel.background = e
+          axis.line = element_
+          axis.text.x = elemer
+          axis.text.y = elemer
+          panel.grid = element
+          plot.margin = margir
`summarise()` has grouped outpu
>
> # ggsave(paste0("Figur_sengeflow ",format(Sys.Date(), "%d%m%y"),".png"), width = 25.5, height = 18, units = "cm")
>
```



```
# Load all libraries
library(tidyverse)
library(lubridate)
library(ggtext)
library(pander)
library(patchwork)
panderOptions("table.

### Data input ###
nfamRAW <- read.csv(f


### Data formatting #
nfamData <- data.fran
# Tilfældig transforn
nfamData$ hashid <- a
nfamData$ birth_year
nfamData$ age <- nfan
nfamData$ gender <- f
# Angiver om forløbet
nfamData$ readmission

# Ankomst til AKA
nfamData$ arrival <-
```

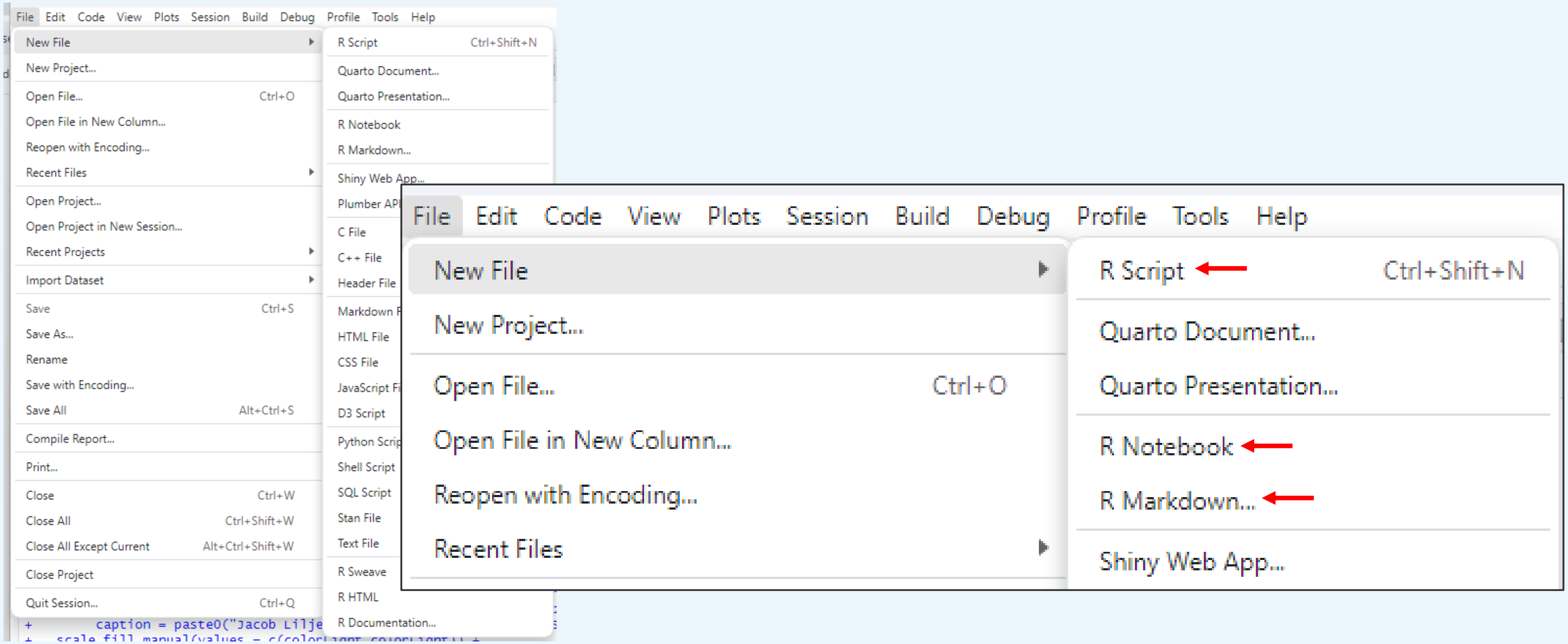
```
+   geom_stratum(aes(fill =
+   geom_text(stat = "stratu
+             aes(label = pa
+   labs(title = "Patientflo
+        caption = paste0("
+   scale_fill_manual(values
+   theme_void() +
+   theme(plot.title = eleme
+
+
+         plot.subtitle = el
+
+
+         plot.caption = ele
+          panel.background
+          axis.line = eleme
+          axis.text.x = ele
+          axis.text.y = ele
+          panel.grid = elem
+          plot.margin = mar
`summarise()` has grouped output by 'firsthospital'. You can override using the `.groups` argument.
>
> # ggsave(paste0("Figur_sengeflow ",format(Sys.Date(), "%d%m%y"),".png"), width = 25.5, height = 18, units = "cm")
> |
```

User Library

	Name	Description	Version
☐	abind	Combine Multidimensional Arrays	1.4-5
☐	askpass	Password Entry Utilities for R, Git, and SSH	1.2.0
☐	assertthat	Easy Pre and Post Assertions	0.2.1
☐	backports	Reimplementations of Functions Introduced Since R-3.0.0	1.4.1
☐	base64enc	Tools for base64 encoding	0.1-3
☐	bit	Classes and Methods for Fast Memory-Efficient Boolean Selections	4.0.5
☐	bit64	A S3 Class for Vectors of 64bit Integers	4.0.5
☐	blob	A Simple S3 Class for Representing Vectors of Binary Data ('BLOBS')	1.2.4
☐	brew	Templating Framework for Report Generation	1.0-10
☐	brio	Basic R Input Output	1.1.4
☐	broom	Convert Statistical Objects into Tidy Tibbles	1.0.5
☐	bslib	Custom 'Bootstrap' 'Sass' Themes for 'shiny' and 'rmarkdown'	0.6.1
☐	cachem	Cache R Objects with Automatic Pruning	1.1.0
☐	callr	Call R from R	3.7.5

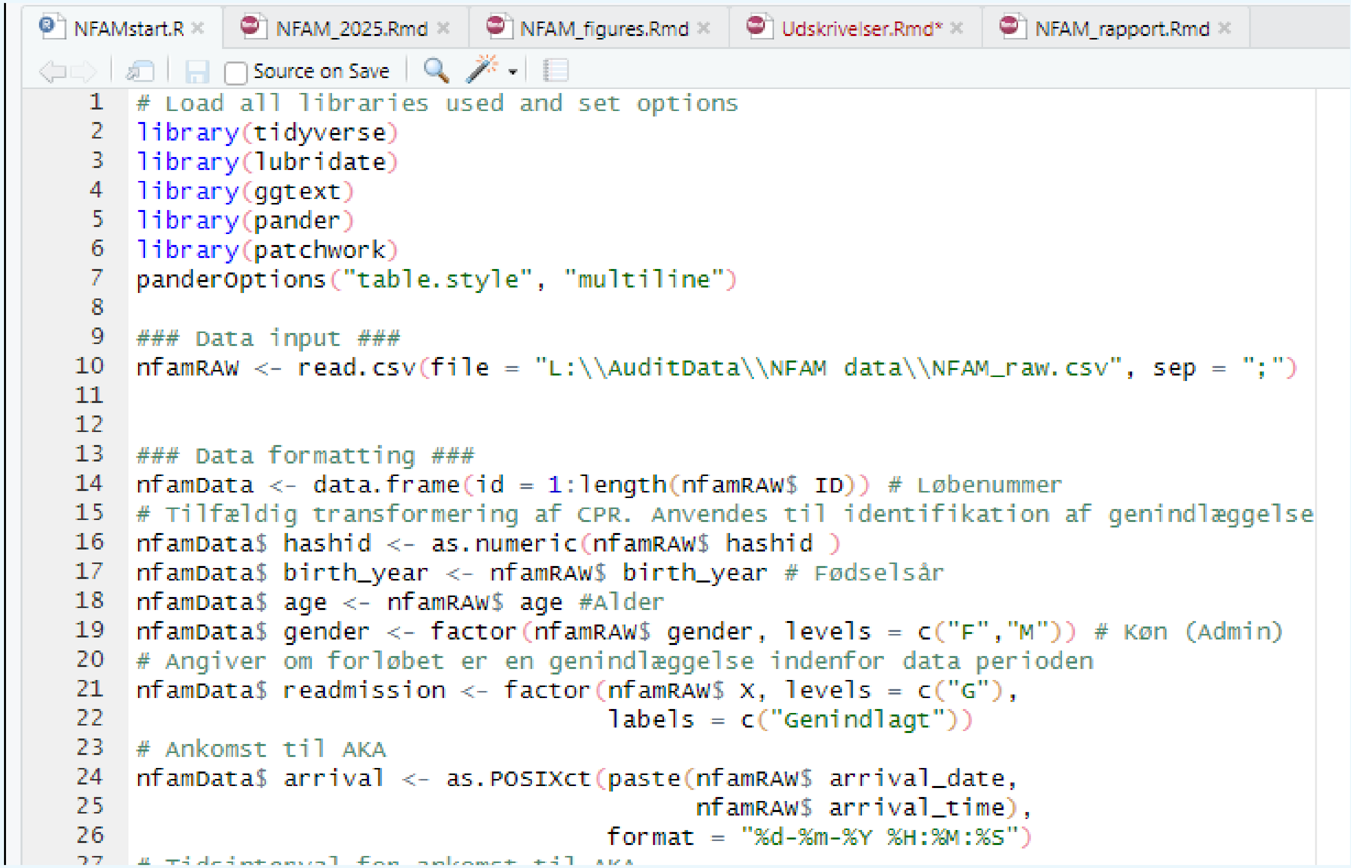
	Name	Description	Version
☐	ids	Generate Random Identifiers	1.0.1
☐	ini	Read and Write '.ini' Files	0.3.1
☐	isoband	Generate Isolines and Isobands from Regularly Spaced Elevation Grids	0.2.7

Projekter og filer



```
+       caption = paste0("Jacob Lilje
```

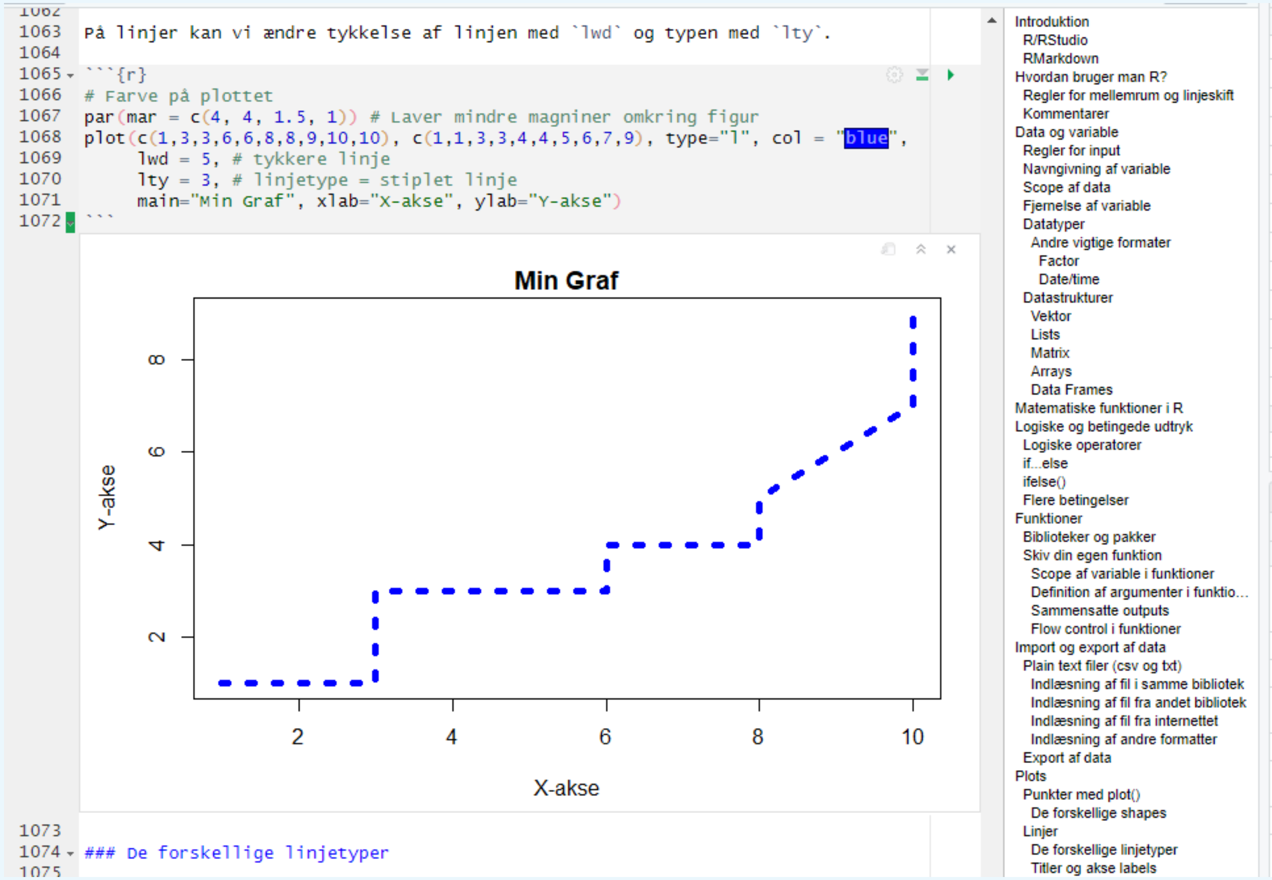
R-script fil



```
# Load all libraries used and set options
library(tidyverse)
library(lubridate)
library(ggtext)
library(pander)
library(patchwork)
panderOptions("table.style", "multiline")

### Data input ###
nfamRAW <- read.csv(file = "L:\\AuditData\\NFAM data\\NFAM_raw.csv", sep = ";")


### Data formatting ###
nfamData <- data.frame(id = 1:length(nfamRAW$ ID)) # Løbenummer
# Tilfældig transformering af CPR. Anvendes til identifikation af genindlæggelse
nfamData$ hashid <- as.numeric(nfamRAW$ hashid )
nfamData$ birth_year <- nfamRAW$ birth_year # Fødselsår
nfamData$ age <- nfamRAW$ age #Alder
nfamData$ gender <- factor(nfamRAW$ gender, levels = c("F","M")) # Køn (Admin)
# Angiver om forløbet er en genindlæggelse indenfor data perioden
nfamData$ readmission <- factor(nfamRAW$ X, levels = c("G"),
                                labels = c("Genindlagt"))
# Ankomst til AKA
nfamData$ arrival <- as.POSIXct(paste(nfamRAW$ arrival_date,
                                      nfamRAW$ arrival_time),
                                format = "%d-%m-%Y %H:%M:%S")
# Tidsinterval for ankomst til AKA
```

- Ren kode
- Koden køres i *Console* (Ctrl+Enter eller ‘Run’) Run
- Tekst kan sættes ind som kommentar, men *køres* ikke
- Kan anvendes i andre dokumenter via funktionen *source()*

R-Markdown fil



~~~
På linjer kan vi ændre tykkelse af linjen med `lwd` og typen med `lty`.

```{r}
# Farve på plottet
par(mar = c(4, 4, 1.5, 1)) # Laver mindre magniner omkring figur
plot(c(1,3,3,6,6,8,8,9,10,10), c(1,1,3,3,4,4,5,6,7,9), type="l", col = "blue",
     lwd = 5, # tykkere linje
     lty = 3, # linjetype = stiplet linje
     main="Min Graf", xlab="X-akse", ylab="Y-akse")
```
~~~

````
### De forskellige linjetyper
````

- Introduktion
  - R/RStudio
  - RMarkdown
- Hvordan bruger man R?
  - Regler for mellemrum og linjeskift
  - Kommentarer
- Data og variable
  - Regler for input
  - Navngivning af variable
  - Scope af data
  - Fjernelse af variable
  - Datatyper
    - Andre vigtige formater
      - Factor
      - Date/time
  - Datastrukturer
    - Vektor
    - Lists
    - Matrix
    - Arrays
    - Data Frames
- Matematiske funktioner i R
- Logiske og betingede udtryk
  - Logiske operatorer
  - if...else
  - ifelse()
  - Flere betingelser
- Funktioner
  - Biblioteker og pakker
  - Skiv din egen funktion
    - Scope af variable i funktioner
    - Definition af argumenter i funktio...
    - Sammensatte outputs
    - Flow control i funktioner
- Import og export af data
  - Plain text filer (csv og txt)
    - Indlæsning af fil i samme bibliotek
    - Indlæsning af fil fra andet bibliotek
    - Indlæsning af fil fra internettet
    - Indlæsning af andre formatter
  - Export af data
- Plots
  - Punkter med plot()
    - De forskellige shapes
  - Linjer
    - De forskellige linjetyper
    - Titler og akse labels

- Kombineret kode og tekst
- Kode indsættes i ‘*chunks*’
- Outputtet vises i samme vindue
- Kan knittes til et dokument

Syntaks

Funktioner består af et navn og en række argumenter, som skrives inde i parentesen

```
functionName(argument1, argument2, ...)
```

```
sample(x = 1:10, size = 100, replace=TRUE)
sample(1:10, 100, TRUE)
sample(replace=TRUE, size = 100, x = 1:10)
```

```
sample(TRUE, 100, 1:10)
Fejl i sample.int(length(x), size, replace,
prob): ugyldigt 'replace'-argument
```

```
# Uden mellemrum
seq(from=10,to=100,by=10)
## [1] 10 20 30 40 50 60 70 80 90 100
# Med mellemrum
seq ( from = 10, to = 100, by = 10)
## [1] 10 20 30 40 50 60 70 80 90 100

# Med linjeskift
seq(
  from=10,
  to=100,
  by=10)
## [1] 10 20 30 40 50 60 70 80 90 100
```

```
se q( from = 10, to = 100, by = 10)
Fejl: uventet symbol in "se q"
```

Pakker og biblioteker

R indeholder allerede en lang række basisfunktioner, som er klar til brug (kaldes *BaseR*)

Derudover findes der forskellige **biblioteker** (*Libraries*) med funktioner, som kan installeres via **pakker** (*packages*)

Pakker skal kun installeres én gang.

Men de enkelte biblioteker skal indlæses `library()` eller refereres `::` hver gang R startes op.

Installation af pakke

Via menu:

Tools → Install packages…
Skriv navnet i *Packages* feltet

Via console/kode

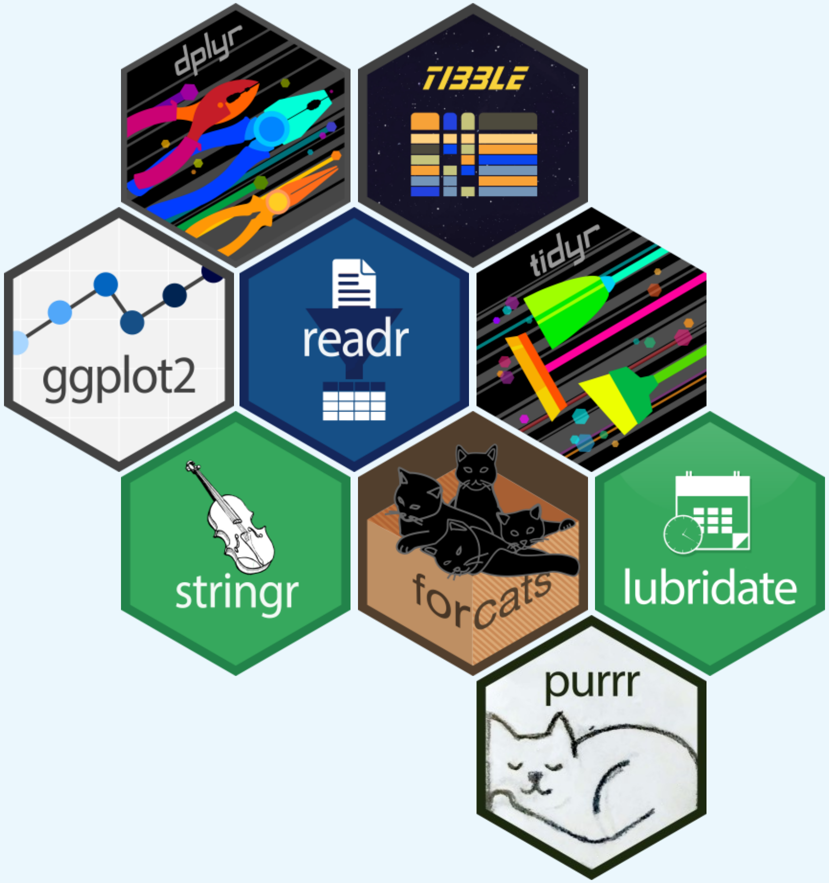
```
install.packages(”Tidyverse”)
```

Indlæsning af bibliotek

```
library(lunar)
```

```
lunar::lunar.phase(date, name=T)
```

Tidyverse



Pakke med forskellige biblioteker til datahåndtering og fremstilling

- **readr**: Værktøjer til indlæsning af datasæt fra forskellige formater
- **dplyr**: Værktøjer til datahåndtering
- **ggplot2**: Datavisualisering
- **tidyr**: Rensning af data
- **tibble**: Tabulering af data
- **stringr**: Værktøjer til string/tekst data
- **forcats**: Værktøjer til kategorisk data
- **lubridate**: Værktøjer til tid/dato data
- **purr**: Værktøjer til funktionel programmering

Data-input

Definition af data-variabel

```
myVariable <- "Hello World”
myVariable = "Hello World"
```

- Variablen skal have et **gyldigt navn**
- Tekst variable skal omsluttes af enkelt (’) eller dobbelt (“) citationstegn
- Heltal indtastes som de er
- Der kan kun bruges punktum som decimalseperator *(3.14195)*
- Konstanterne **`TRUE`** og **`FALSE`** bruges kun til logiske variable

Det er vigtigt at skelne mellem **NULL** og **NA**, da de håndteres meget forskelligt: **NULL** = *udefineret* element (Har ingen værdi), mens **NA** = manglende værdi.

Tilladte navne:

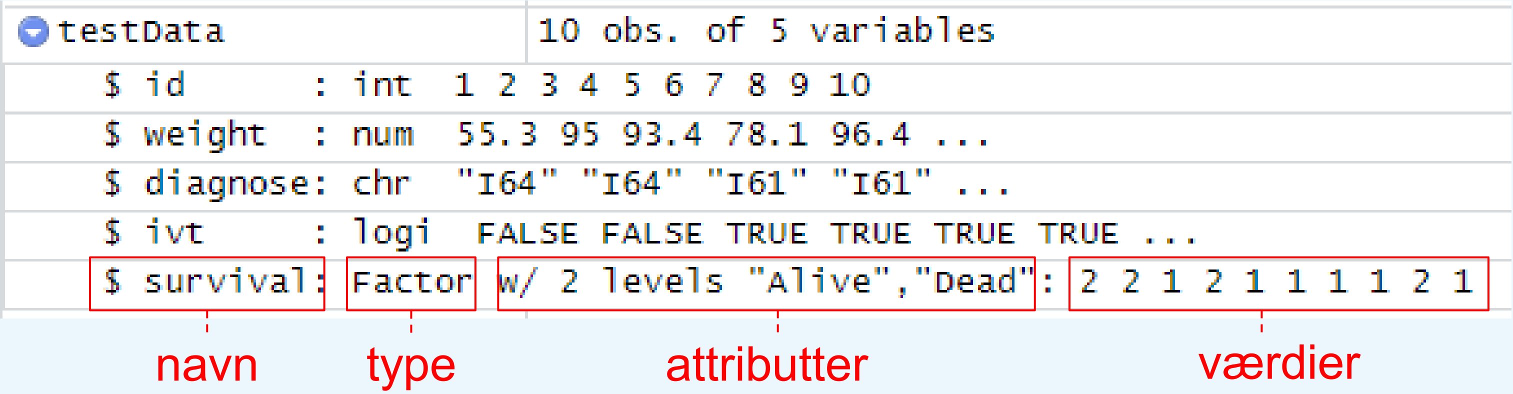
```
minvariabel <- "Værdi"
min_variabel <- "Værdi"
minVariabel <- "Værdi"
MINVARIABEL <- "Værdi"
minvariabel2 <- "Værdi"
.minvariabel <- "Værdi"
```

Forbudte navne:

```
2minvariabel <- "Værdi"
min-variabel <- "Værdi"
min variabel <- "Værdi"
_min_variabel <- "Værdi"
minv@ariabel <- "Værdi"
TRUE <- "Værdi"
```

Datatyper

`heltalVar <- 42`	Integer	Heltal
`decimalVar <- 3.14195`	Numeric	Decimaltal
`tekstVar <- "Tekst"`	Character	Tekst
`logiskVar <- TRUE`	Logical	Logiske værdier; Kan være enten TRUE / FALSE
`nullVar <- NULL`		Udefineret data
`naVar <- NA`		Manglende data / tomt felt
`factVar <- factor("Tekst")`	Factor	Kategorisk data



```
testData          10 obs. of 5 variables
   $ id      : int  1 2 3 4 5 6 7 8 9 10
   $ weight  : num  55.3 95 93.4 78.1 96.4 ...
   $ diagnose: chr  "I64" "I64" "I61" "I61" ...
   $ ivt     : logi  FALSE FALSE TRUE TRUE TRUE TRUE ...
   $ survival: Factor w/ 2 levels "Alive","Dead": 2 2 1 2 1 1 1 1 2 1
```

Datastrukturer

Datastrukturer er metoder til at organiserer data

Vektor: Samling af data af samme datatype i en dimension:

```
drikkevarer <- c("vand","kaffe","te")
```

Lists: Samling af data af forskellig datatype i en dimension:

```
minListe <- list("kaffe","vand", 42,
TRUE)
```

Matrix: Samling af data af samme datatype i to dimensioner:

```
minMatrix <- matrix(c(1,2,3,4,5,6),
nrow = 3, ncol = 2)
```

Dataframe: Datastruktur, der består af sammenhængende vektorer

```
minDataFrame <- data.frame (
     patientnr = c(1, 2, 3, 4,
       5, 6, 7, 8),
     diagnose = c("I61", "I61",
       "I61", "I64", "I64",
       "I63", "I64", "I61"),
     sysBT = c(142, 165, 153,
       120, 145, 168,
       165, 133),
     køn = factor(c(1, 1, 2, 1,
       2, 2, 2, 1),
       labels = c("Mand", "Kvinde")
       ))
```

Matematiske operatorer

Simple operatorer

```
5 + 5
## [1] 10
5 - 5
## [1] 0
5 * 5
## [1] 25
5 / 5
## [1] 1
```

Regning med elementer

```
tal <- 8
tal + 10
## [1] 18
```

Indbyggede matematiske funktioner

```
sqrt(100)
## [1] 10
4^2
## [1] 16
log(2)
## [1] 0.6931472
exp(1)
## [1] 2.718282
sum(2,18,5)
## [1] 25
```

Logiske operatorer

```
10 < 12 # TRUE fordi 10 er mindre end 12
## [1] TRUE

10 <= 12 # TRUE fordi 10 er mindre end ELLER lig med 12
## [1] TRUE

10 > 12 # FALSE fordi 10 ikke er større end 12
## [1] FALSE

10 >= 12 # FALSE fordi 10 ikke er større end ELLER lig med 12
## [1] FALSE

10 == 12 # FALSE fordi 10 ikke er lig med 12
## [1] FALSE

10 != 12 # TRUE fordi 10 ikke er lig med 12
## [1] TRUE
```

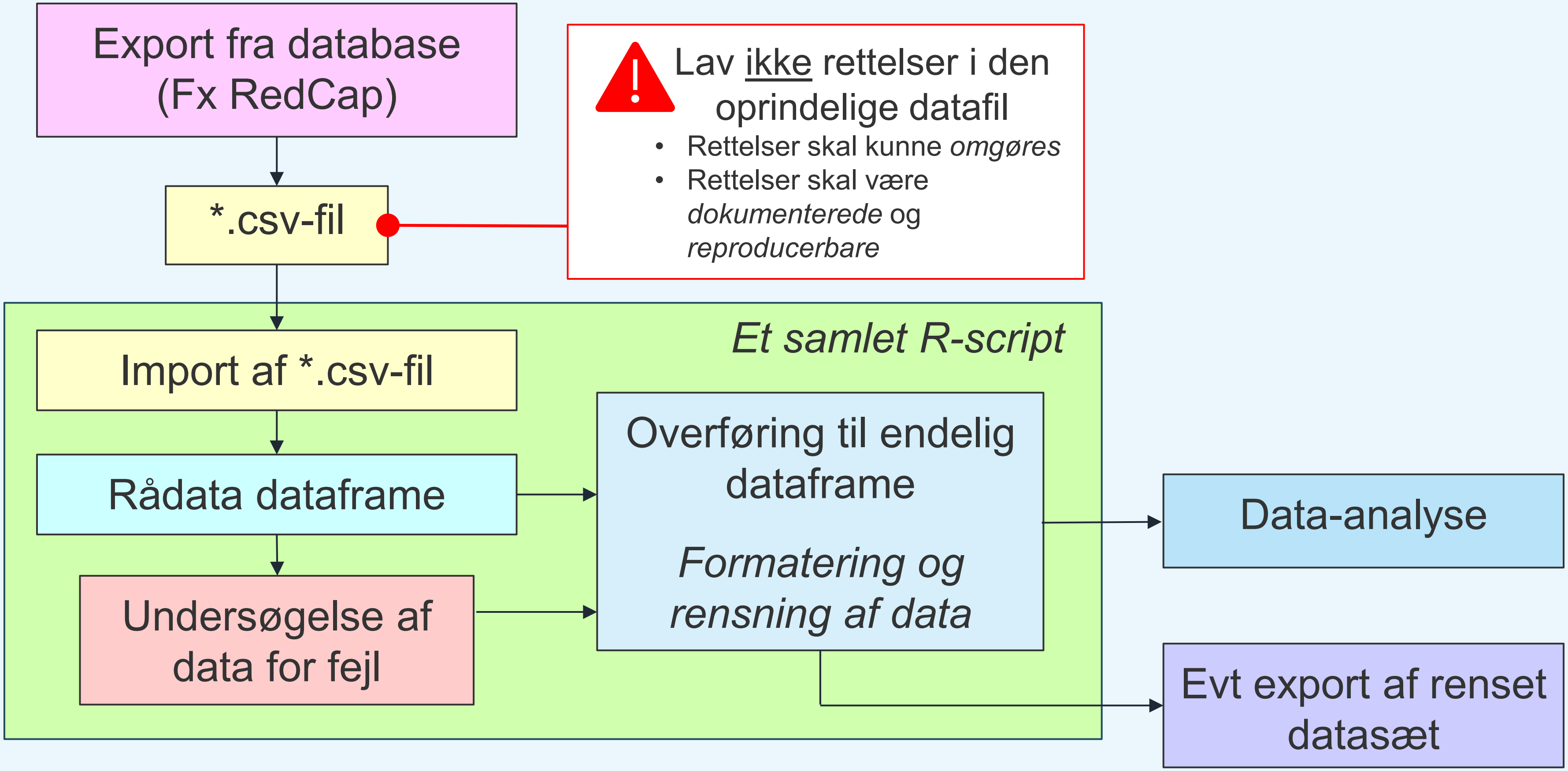
If…else funktioner

```
a <- 10
b <- 12

if (a < b){
print("a er mindre end b")
} else {
print("a er større end b") }
## [1] "a er mindre end b"
```

```
a <- 7562

ifelse(a > 1000, "Ja", "Nej")
## [1] "Ja"
```

```
a <- 1000
b <- 25
c <- 756
if(a > b & a > c){ print("begge betingelser er opfyldt") }
## [1] "begge betingelser er opfyldt"
if(a > b | a > c){ print("mindst en betingelse er opfyldt") }
## [1] "mindst en betingelse er opfyldt"
```

Klargøre af data til analyse



Export fra database (Fx RedCap)

*.csv-fil

⚠ Lav ikke rettelser i den oprindelige datafil

- Rettelser skal kunne *omgøres*
- Rettelser skal være *dokumenterede* og *reproducerbare*

Et samlet R-script

Import af *.csv-fil

Rådata dataframe

Undersøgelse af data for fejl

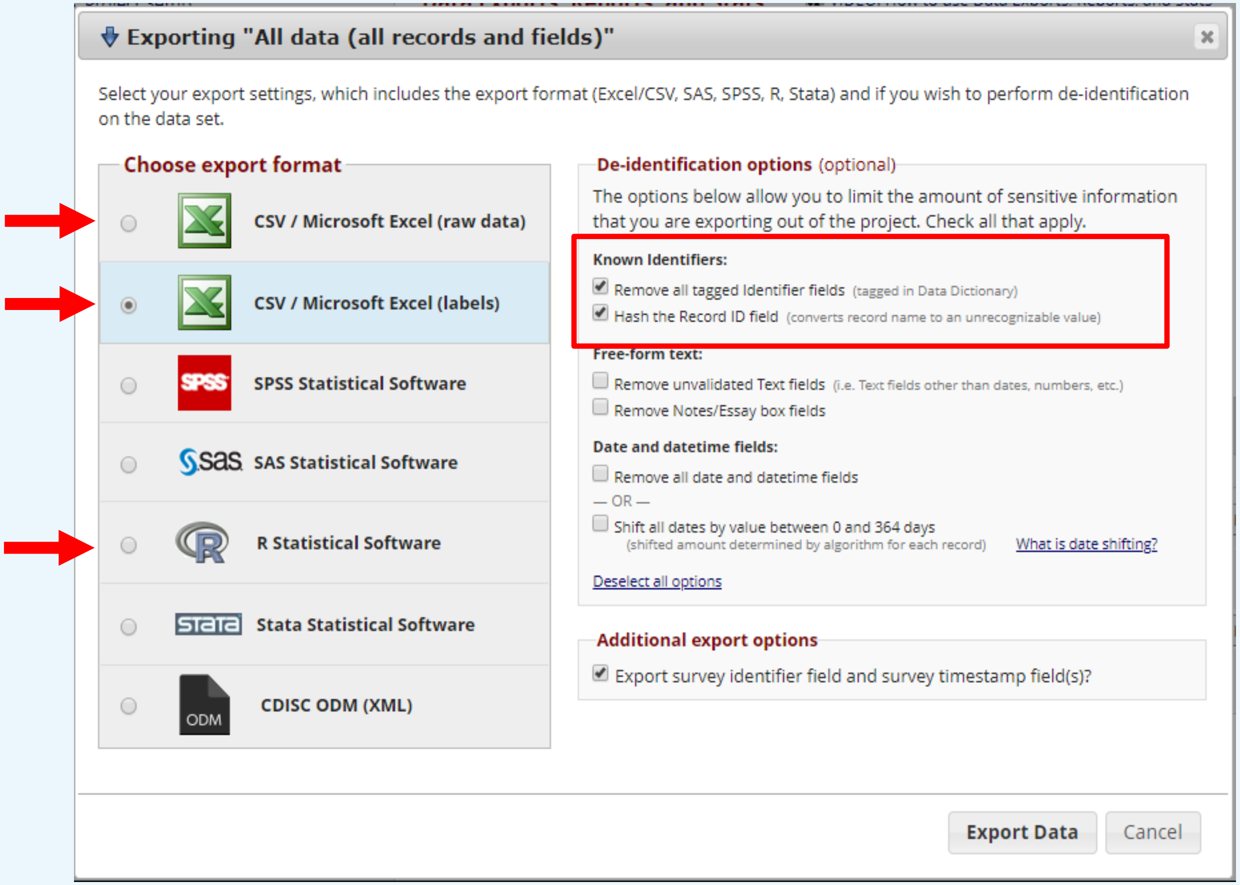
Overføring til endelig dataframe

Formatering og rensning af data

Data-analyse

Evt export af renset datasæt

Export fra RedCap



Exporting "All data (all records and fields)"

Select your export settings, which includes the export format (Excel/CSV, SAS, SPSS, R, Stata) and if you wish to perform de-identification on the data set.

Choose export format

- ○ CSV / Microsoft Excel (raw data)
- ● CSV / Microsoft Excel (labels)
- ○ SPSS Statistical Software
- ○ SAS Statistical Software
- ○ R Statistical Software
- ○ Stata Statistical Software
- ○ CDISC ODM (XML)

De-identification options (optional)

The options below allow you to limit the amount of sensitive information that you are exporting out of the project. Check all that apply.

Known Identifiers:

☑ Remove all tagged Identifier fields (tagged in Data Dictionary)
☑ Hash the Record ID field (converts record name to an unrecognizable value)

Free-form text:

☐ Remove unvalidated Text fields (i.e. Text fields other than dates, numbers, etc.)
☐ Remove Notes/Essay box fields

Date and datetime fields:

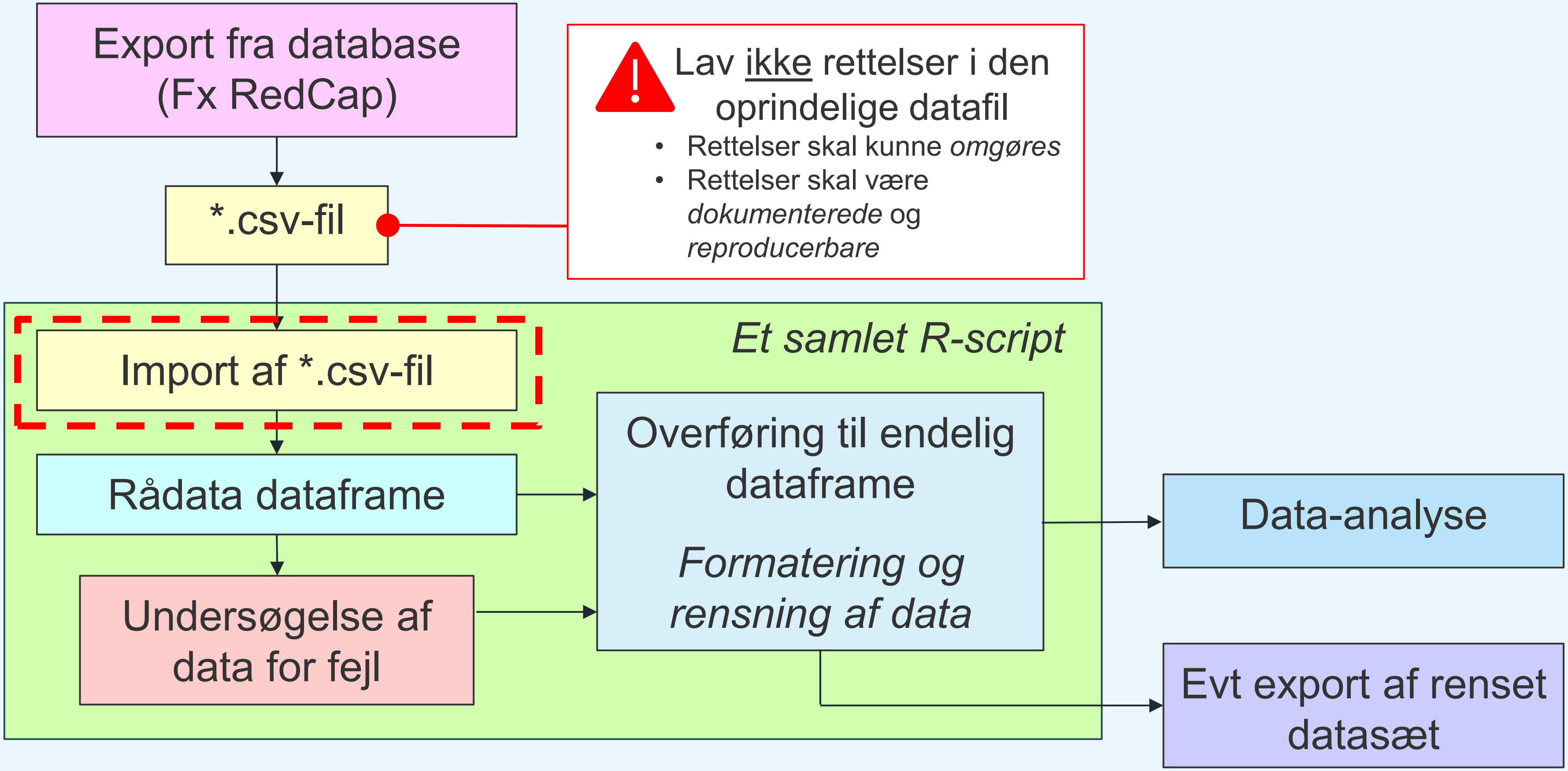
☐ Remove all date and datetime fields
— OR —
☐ Shift all dates by value between 0 and 364 days
(shifted amount determined by algorithm for each record) What is date shifting?

Deselect all options

Additional export options

☑ Export survey identifier field and survey timestamp field(s)?

Klargøre af data til analyse



Export fra database (Fx RedCap)

*.csv-fil

⚠ Lav ikke rettelser i den oprindelige datafil

- Rettelser skal kunne *omgøres*
- Rettelser skal være *dokumenterede* og *reproducerbare*

Et samlet R-script

Import af *.csv-fil

Rådata dataframe

Undersøgelse af data for fejl

Overføring til endelig dataframe

Formatering og rensning af data

Data-analyse

Evt export af renset datasæt

Indlæsning af data

Indlæsning af *.csv-fil

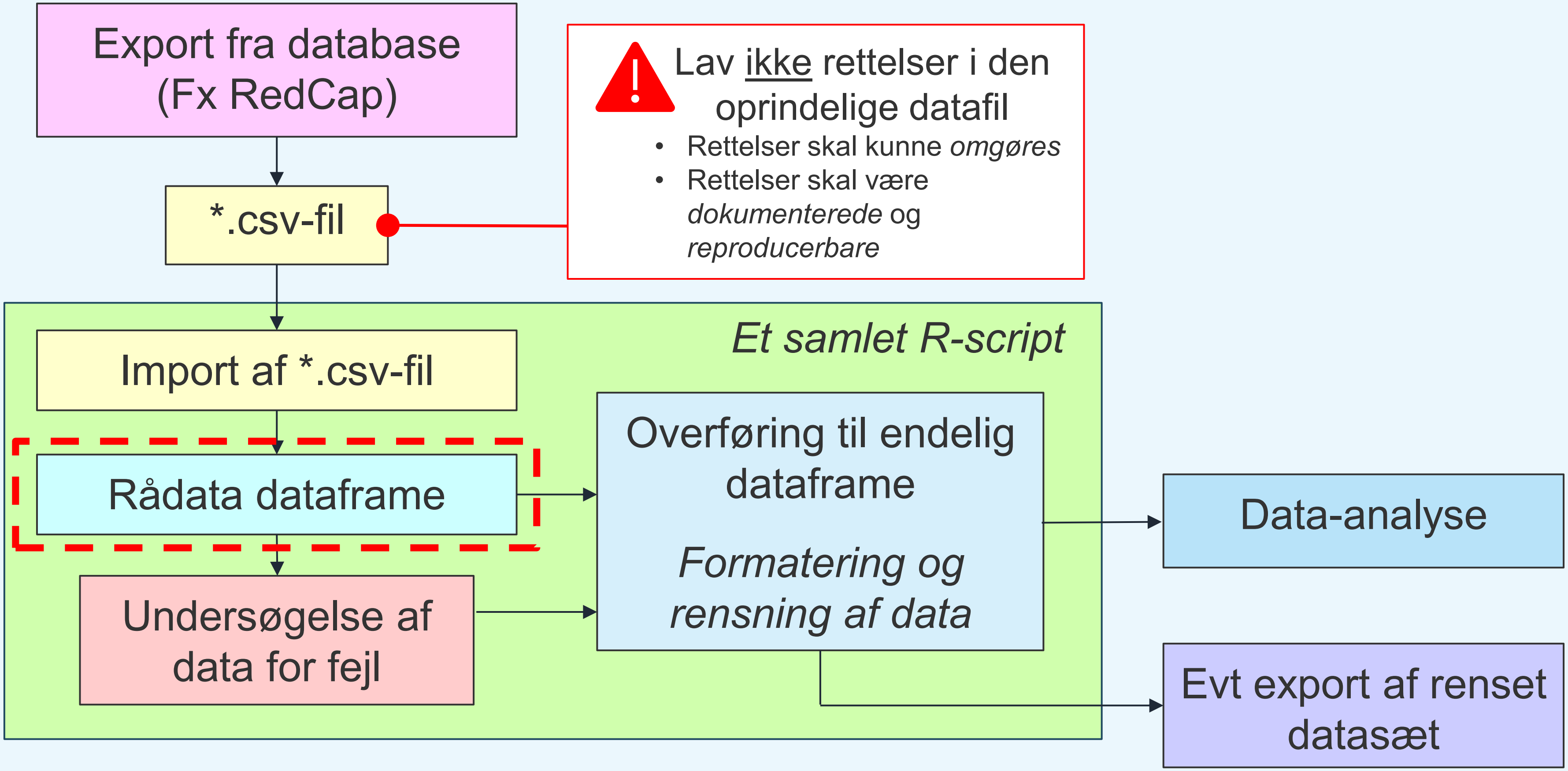
```
# Indlæsning af fil i engelsk formatering
strokedf <- read.csv("strokedata.csv")
# Indlæsning af fil i dansk formatering, altså
# semikolon mellem værdier og komma som decimal separator
strokedf <- read.csv("strokedata_dk.csv", sep=";", dec = ",")
```

```
# Indlæsning af fil fra et fælles drev
strokedf <- read.csv("L:\\AuditData\\ApoData\\strokedata.csv")
```

Indlæsning fra web-side

```
strokedf <-
read.csv("https://jacobliljehult.github.io/research/strokedata.csv",
sep=",", header=T, stringsAsFactors = TRUE)
```

Klargøre af data til analyse



Export fra database (Fx RedCap)

*.csv-fil

⚠ Lav ikke rettelser i den oprindelige datafil

- Rettelser skal kunne *omgøres*
- Rettelser skal være *dokumenterede* og *reproducerbare*

Et samlet R-script

Import af *.csv-fil

Rådata dataframe

Undersøgelse af data for fejl

Overføring til endelig dataframe

Formatering og rensning af data

Data-analyse

Evt export af renset datasæt

Dataflow

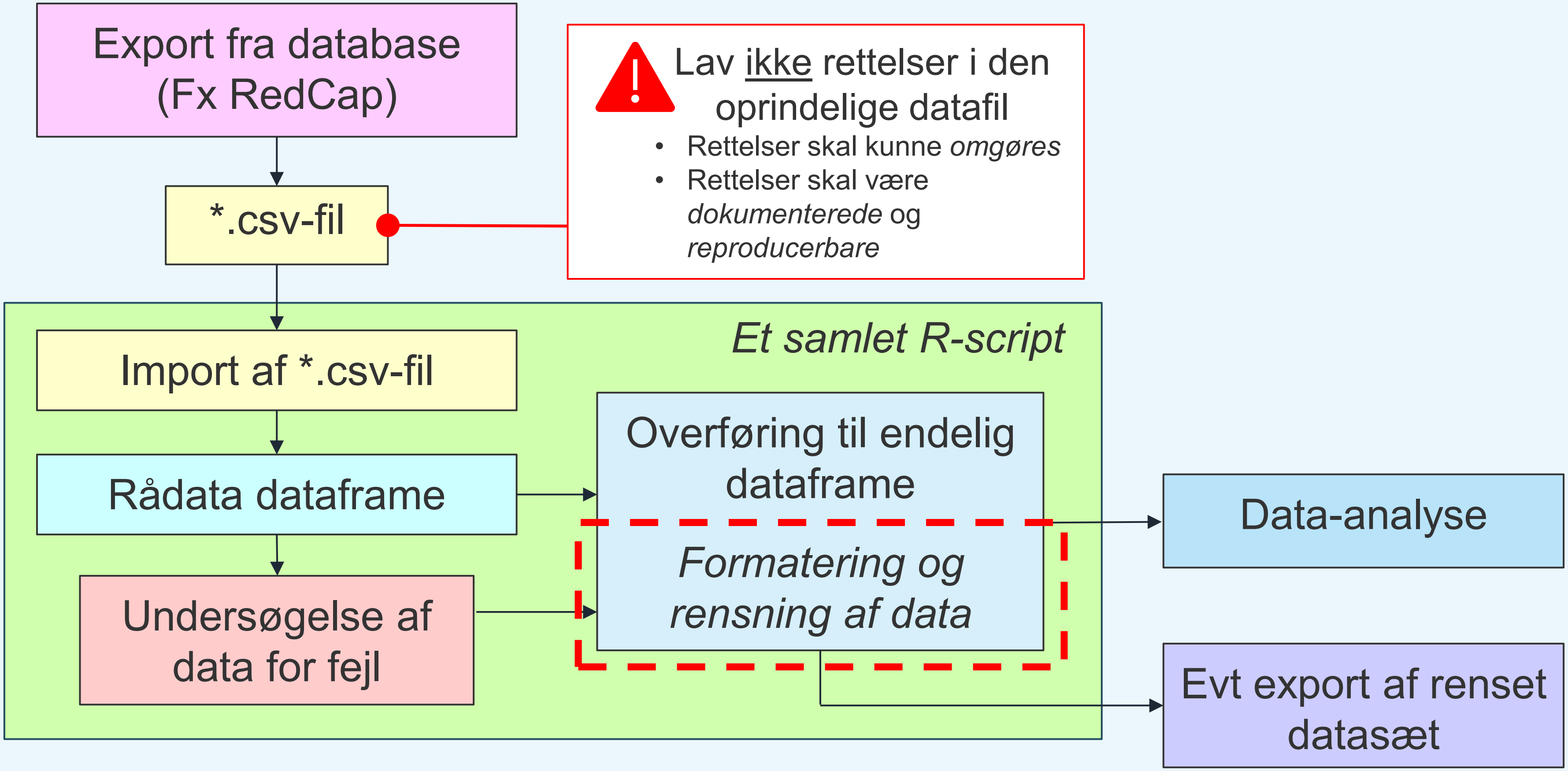
NFAMstart.R

```r
# Load all libraries used
library(tidyverse)
library(lubridate)
### Data input ###
nfamRAW <- read.csv(file =
"L:\\AuditData\\NFAM data\\NFAM_raw.csv",
sep = ";")

### Data formatting ###
nfamData <- data.frame(id =
1:length(nfamRAW$ ID)) # Løbenummer
nfamData$ age <- nfamRAW$ age #Alder
nfamData$ gender <- factor(nfamRAW$
gender, levels = c("F","M")) # Køn (Admin)
# Ankomst til AKA
nfamData$ arrival <-
as.POSIXct(paste(nfamRAW$ arrival_date,
nfamRAW$ arrival_time),
format = "%d-%m-%Y %H:%M:%S")
rm(nfamRAW) # Fjerner rådata
```

NFAM_rapport.Rmd

~~~
```{r}
source(”NFAMstart.R”)
```


# Patientgrundlag
Siden 4. septemter har der været `r
length(nfamData$id)` neurologiske
patientforløb i AKA.

```{r}
nfamData %>% select(npathway_bed,
destination,destination_ward) %>%
filter(!is.na(npathway_bed) &
!is.na(destination_ward)) %>%
mutate(grp = factor(paste( npathway_bed,
"|",destination_ward))) %>%
group_by(grp) %>% summarise(Antal = n())
%>% mutate(Procent = paste( round(
Antal/sum(Antal)*100,1),"%"))
```
~~~

Klargøre af data til analyse



Export fra database (Fx RedCap)

*.csv-fil

⚠ Lav ikke rettelser i den oprindelige datafil

- Rettelser skal kunne *omgøres*
- Rettelser skal være *dokumenterede* og *reproducerbare*

Et samlet R-script

Import af *.csv-fil

Rådata dataframe

Undersøgelse af data for fejl

Overføring til endelig dataframe

Formatering og rensning af data

Data-analyse

Evt export af renset datasæt

Struktur af datasæt

```r
str(testData)
```

```
## 'data.frame': 100 obs. of 11 variables:
## $ id : int 1 2 3 4 5 6 7 8 9 10 ...
## $ hospital : Factor w/ 5 levels "NOH","BBH","HGH",..: 2 4 3 1 4 3 3 3 5 2 ...
## $ admission : POSIXct, format: "2024-01-04 15:02:00" "2024-01-18 13:50:00" ...
## $ discharge : POSIXct, format: "2024-01-12 16:35:00" "2024-01-26 15:44:00" ...
## $ gender : Factor w/ 2 levels "F","M": 2 2 1 1 2 2 1 1 1 1 ...
## $ age : num 66 65 60 53 88 55 68 69 59 72 ...
## $ height : num 183 169 173 184 177 191 162 162 163 155 ...
## $ weight : num 86.6 72.5 34.7 53.2 76 ...
## $ allocation: Factor w/ 2 levels "Control","Treatment": 2 2 2 2 2 1 1 2 1 1 ...
## $ sbp_pre : num 151 127 146 137 141 152 148 122 144 137 ...
## $ sbp_post : num 144 120 144 128 137 151 151 119 146 135 ...
```

navn type værdier

Formatering af datatyper

```
# Formattering af numeriske værdier
testData$age <- as.numeric(testDataRaw$age)

# Formattering af heltal
testData$age <- as.integer(testDataRaw$age)

# Formattering af tekst
testData$hospital <- as.character(testDataRaw$hospital)

# Formattering af factor fra tekstvariabel
testData$gender <- as.factor(testDataRaw$gender)

# Formattering af factor med labels
testData$hospital <- as.factor(testDataRaw$hospital,
        levels = c(1,2,3,4,5),
        labels = c("NOH", "BBH", "HGH", "AHH", "RH"))
```

Formatering af dato og tid

Dato

Tekstformat: `"2024-01-04"`

```
as.POSIXct(testDataRaw$admission_date, format = "%Y-%m-%d")
```

```
lubridate::ymd(testDataRaw$admission_date)
```

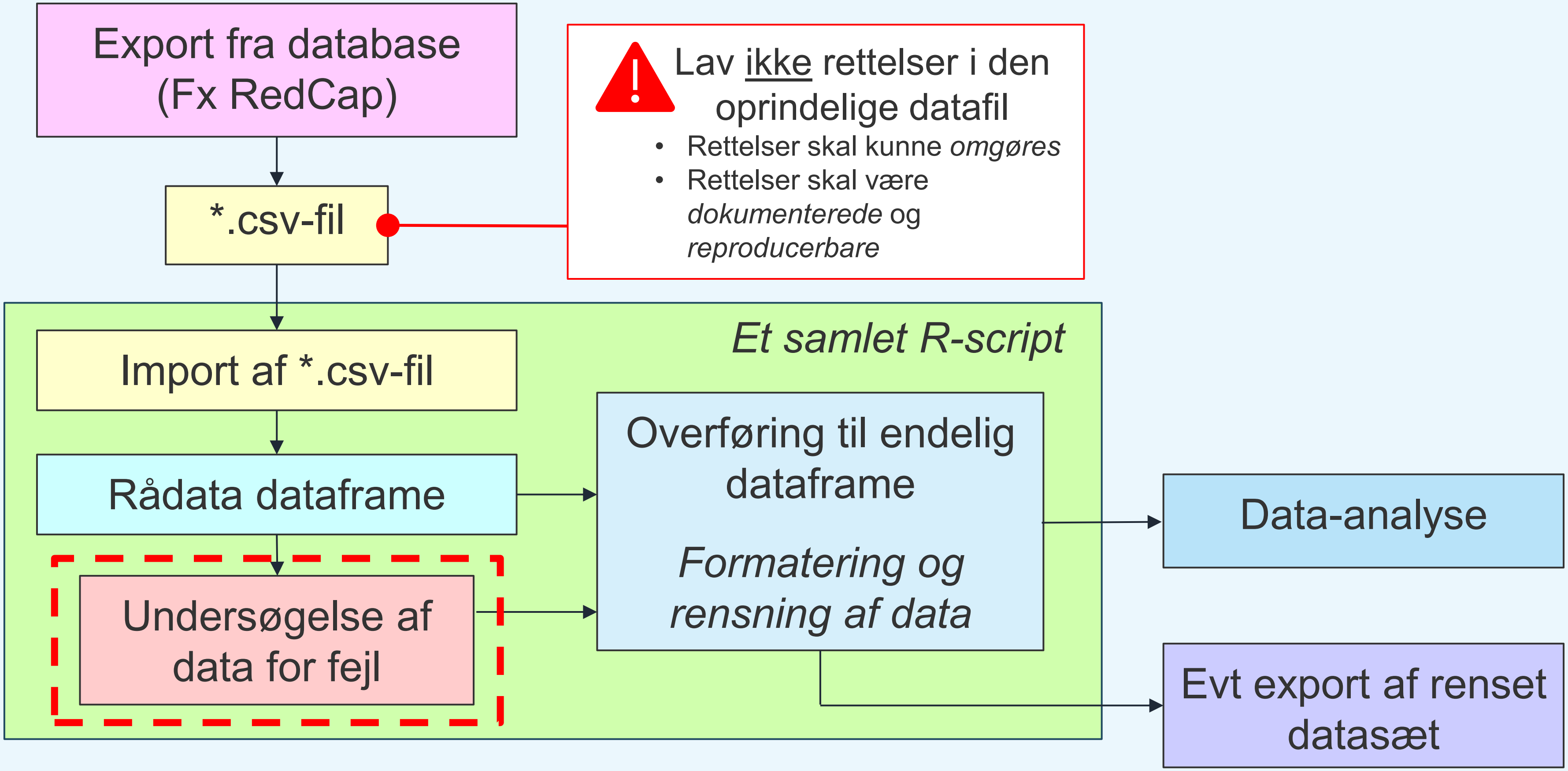
Dato + tid

Tekstformat: `"2024-01-04 10:35"`

```
as.POSIXct(testDataRaw$admission, format = "%Y-%m-%d %H:%M")
```

```
lubridate::ymd_hm(testDataRaw$admission)
```

Klargøre af data til analyse



Export fra database (Fx RedCap)

*.csv-fil

Lav ikke rettelser i den oprindelige datafil

- Rettelser skal kunne *omgøres*
- Rettelser skal være *dokumenterede* og *reproducerbare*

Et samlet R-script

Import af *.csv-fil

Rådata dataframe

Undersøgelse af data for fejl

Overføring til endelig dataframe

Formatering og rensning af data

Data-analyse

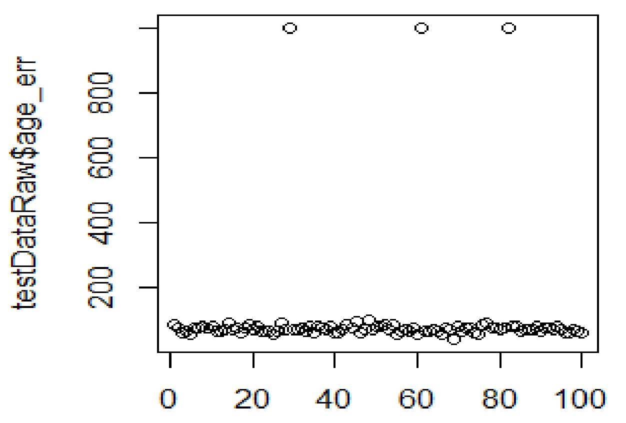
Evt export af renset datasæt

Undersøgelse for ekstreme værdier

```r
summary(testDataRaw$age_err)
```

```
## Min.     1st Qu.    Median    Mean     3rd Qu.    Max.
## 45.00    64.00      69.00     96.43    74.50      999.00
```

```r
plot(testDataRaw$age_err)
```



Rettelse af ekstreme værdier til NA

```r
ifelse(testDataRaw$age_err != 999, testDataRaw$age_err, NA)
```

Forkerte kategorier

```
unique(testDataRaw$gender_err)
```

```
## [1] 1 0 2
```

```
table(testDataRaw$gender_err)
```

```
## 0  1  2
## 39 53 8
```

```
table(testDataRaw$gender_err, testDataRaw$hospital)
```

```
##     NOH BBH HGH AHH RH
## 0   8   11  7   13  0
## 1   12  9   13  7   12
## 2   0   0   0   0   8
```

Betinget rettelse af værdier

```
ifelse(testDataRaw$hospital == ”RH”,
       ifelse(testDataRaw$gender_err == 1, 0, 1) ,
       testDataRaw$gender_err)
```

Fejl i decimaltegn

```
testDataRaw$ weight_err
```

```
## [1] "86,65" "72,5" "34,69" "53,17" "76,03" "100,65" "64,01" "53,79"
```

```
testData$ weight_err <- as.numeric(testDataRaw$ weight_err)
```

```
## Warning: NAs introduced by coercion
```

Rettelse af tegn i tekst + formatering til tal

```
testData$ weight_err <- as.numeric(
                   gsub(",", ".", testDataRaw$ weight_err) )
```

Forkert kodning af NA-værdi

```
unique(testDataRaw$ height_err)
```

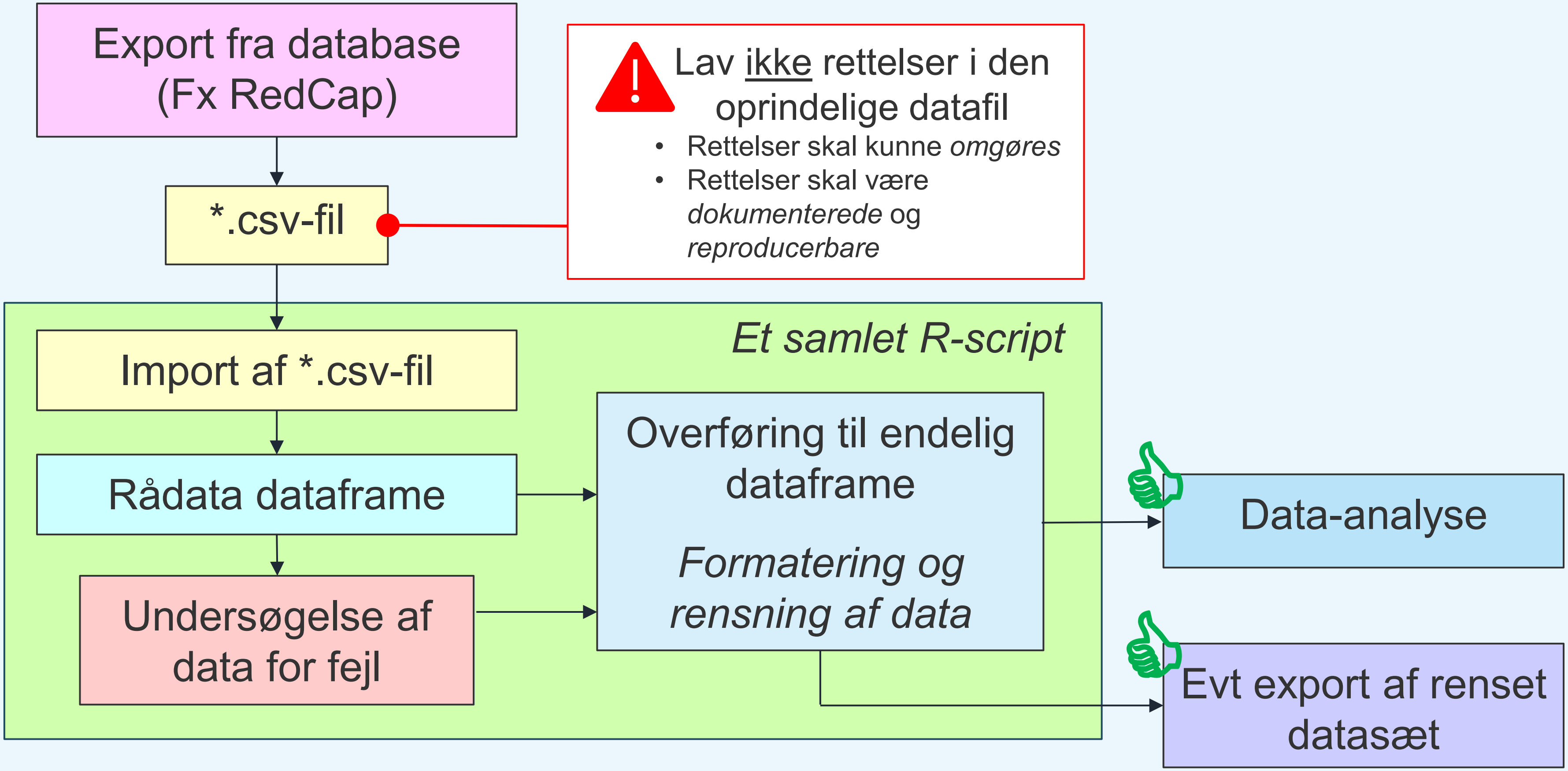
```
## [1] "183" "169" "173" "184" "177" "191" "162" "." "163" "155" "164" "189"
```

Rettelse af manglende værdi

```
testData$ height_err <- as.numeric(testDataRaw$ height_err)
```

```
## Warning: NAs introduced by coercion
## [1] 183 169 173 184 177 191 162 NA 163 155 164 189 168 184 171
```

Klargøre af data til analyse



Export fra database (Fx RedCap)

*.csv-fil

Lav ikke rettelser i den oprindelige datafil

- Rettelser skal kunne *omgøres*
- Rettelser skal være *dokumenterede* og *reproducerbare*

Et samlet R-script

Import af *.csv-fil

Rådata dataframe

Undersøgelse af data for fejl

Overføring til endelig dataframe

Formatering og rensning af data

Data-analyse

Evt export af renset datasæt

Deskriptiv statistik

Kvantitative data

```
summary(strokedata$age)

 Min.  1st Qu.  Median   Mean  3rd Qu.   Max.
21.00   64.00   73.00   71.85   81.00   99.00
```

Konfidensinterval

En nem måde at udregne et konfidensinterval for en kontinuert variable er at lave en *one-sample t-test* mod variablens eget gennemsnit

```
t.test(strokedata$age,
       mu = mean(strokedata$age)
       )$conf.int[1:2]
[1] 71.09142 72.60984
```

`length(x)`	længden af en variabel
`mean(x)`	gennemsnit
`median(x)`	median
`min(x)`	mindste værdi
`max(x)`	maksimal værdi
`range(x)`	range
`var(x)`	variation
`sd(x)`	standard deviation
`quantile(x, p)`	percentiler

Argumentet `na.rm = TRUE` kan bruges til at ignorere NA værdier

Deskriptiv statistik

Kategorisk data

Enkel variabel

```
table(strokedata$inhospmors)
Alive Dead
  971   60
```

2 x 2 tabel *(Absolutte tal)*

```
with(strokedata,
    table(diagnosis, inhospmors))
inhospmors
diagnosis Alive Dead
      ICH    88   22
      IS    883   38
```

2 x 2 tabel *(procent)*

```
with(strokedata,
    prop.table(
    table(diagnosis, inhospmors)))
          inhospmors
diagnosis      Alive       Dead
      ICH 0.08535403 0.02133851
       IS 0.85645005 0.03685742
```

→

```
# Procent for rækker
with(df, prop.table(table(x, y) ,1))
# Procent for koloner
with(df, prop.table(table(x, y) ,2))
```

Konfidensinterval

Konfidensintervaller for proportioner kan udregnes med funktionerne `prop.test()` eller `binom.test()`

```
prop.test(75, 100,
    conf.level = 0.95)$conf.int[1:2]
[1] 0.6516159 0.8288245
```

Slides

Scan mig

http://jacobliljehult.github.io/research/