

# Get an Overview with overviewR: : CHEAT SHEET



## Generate Tables

**overview\_tab** generates a data frame that collapses the time condition for each id by taking into account potential gaps in the time frame

id	time	Var1	Var2
A	1990		
A	1991		
A	1992		
B	1990		

id	time
A	1990 - 1992
B	1990

```
output_table <-  
overview_tab(  
  dat = toydata,  
  id = ccode,  
  time = year)
```

add data frame

define your time and scope variables

It also works with multiple time arguments (day, month, and year)

id	year	month	day	Var1	Var2
A	1990	3	1		
A	1990	3	2		
A	1990	3	3		
B	1991	4	10		

id	time
A	1990-03-01 - 1990-03-03
B	1991-04-10

```
overview_tab(  
  dat = toydata,  
  id = ccode,  
  time = list(year = toydata$year,  
              month = toydata$month,  
              day = toydata$day),  
  complex_date = TRUE)
```

define the time variables

set argument complex\_date to TRUE

**overview\_tab** and **overview\_na** can also handle **data.table** objects (all other functions currently only with **data.frames**) to increase the performance on larger data sets

**overview\_crosstab** generates a cross table that divides the data based on two conditions

id	time		

	Yes	No
Yes		
No		

```
output_crosstab <-  
overview_crosstab(  
  dat = toydata,  
  cond1 = gdp,  
  cond2 = population,  
  threshold1 = 25000,  
  threshold2 = 27000,  
  id = ccode,  
  time = year  
)
```

define your conditions with cond1 and cond2

set your thresholds

⚠ If a data set is used that has multiple observations on the id-time unit, the function automatically aggregates the data set using the mean of condition 1 (**cond1**) and condition 2 (**cond2**).

## Workflows

All **overviewR** functions can be easily integrated in the **tidyverse**.

```
toydata %>%  
  dplyr::filter(year > 1993) %>%  
  overview_na()
```

The example shows how to filter or wrangle data before calling **overview\_na**.

To change the visual appearance of plots, the user can rely on **ggplot2** layering logic.

## Export Results

### Tables

**overview\_print** generates a LaTeX output (works with both **overview\_tab** and **overview\_crosstab** output)

```
overview_print(  
  obj = output_table)
```

```
overview_print(  
  obj = output_crosstab,  
  crosstab = TRUE)
```

TRUE for cross tables

The table can be modified with the **title**, **id**, **time**, **cond1**, and **cond2** arguments to replace default names

It also allows to save your output in a .tex file

```
overview_print(  
  obj = output_table,  
  save_out = TRUE,  
  path = "SET-YOUR-PATH",  
  file = "output.tex")
```

define where your output should be stored

The outputs of **overview\_tab** and **overview\_crosstab** are also compatible with other packages and functions such as **xtable**, **flextable**, or **kable** from **knitr**.

To generate a table in Rmarkdown with **knitr::kable**:

```
knitr::kable(output_table)
```

### Plots

As the plots are based on **ggplot2**, plots can be stored with **ggplot2::ggsave**

```
ggplot2::ggsave(  
  output_plot,  
  filename = "FILENAME.png")
```

add plot object

add filename

Alternatively, storing the object also works this way:

```
png("FILENAME.png")  
  
output_plot  
  
dev.off()
```

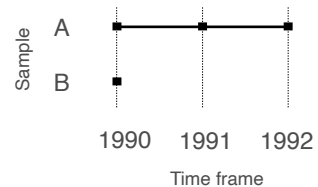
# Get an Overview with overviewR: : CHEAT SHEET



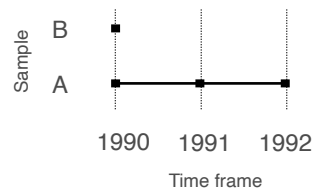
## Generate Plots

### Sample overview

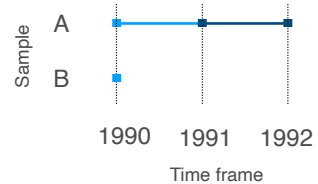
**overview\_plot** illustrates the information that is generated in overview\_tab in a ggplot2 graphic



```
overview_plot(
  dat = toydata,
  id = ccode,
  time = year)
```

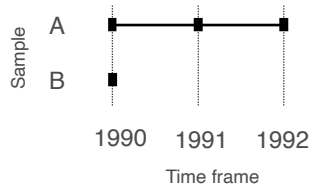


```
overview_plot(
  dat = toydata,
  id = ccode,
  time = year,
  asc = FALSE)
```



```
overview_plot(
  dat = toydata,
  id = ccode,
  time = year,
  color = before)
```

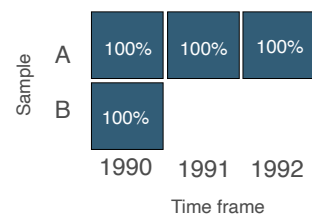
Add variable to identify time periods



```
overview_plot(
  dat = toydata,
  id = ccode,
  time = year,
  dot_size = 5)
```

change size of dots

**overview\_heat** is similar to overview\_plot but presents the frequency of data points by id-time-unit in a heat map

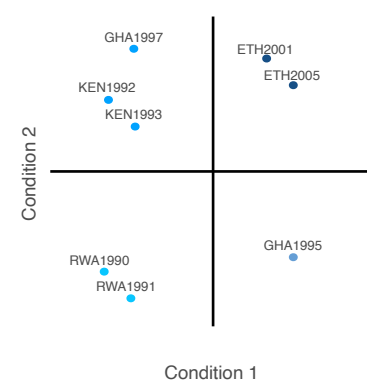


```
overview_heat(
  dat = toydata,
  id = ccode,
  time = year,
  perc = TRUE,
  exp_total = 12)
```

displays percentage

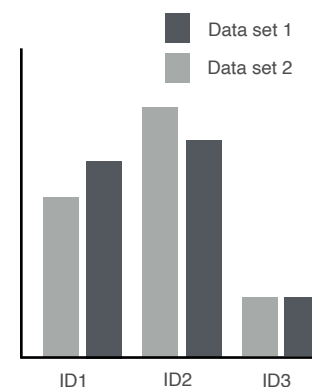
max observations by id-time unit

**overview\_crossplot** allows to illustrate similar information as presented in a cross table (overview\_crosstab)



```
overview_crossplot(
  toydata,
  id = ccode,
  time = year,
  cond1 = gdp,
  cond2 = population,
  threshold1 = 25000,
  threshold2 = 27000,
  color = TRUE,
  label = TRUE
)
```

**overview\_overlap** plots the overlap between two data sets



```
overview_overlap(
  dat1 = toydata,
  dat2 = toydata2,
  dat1_id = ccode,
  dat2_id = ccode,
  plot_type = "bar"
)
```

This is the default

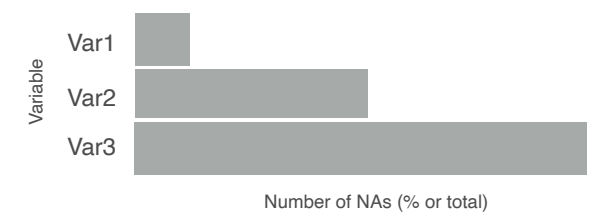


```
overview_overlap(
  dat1 = toydata,
  dat2 = toydata2,
  dat1_id = ccode,
  dat2_id = ccode,
  plot_type = "venn"
)
```

Represents a Venn diagram

### Missing values (NAs)

**overview\_na** returns a horizontal ggplot2 bar plot that indicates the amount of missing data (NAs) for each variable (column-wise)



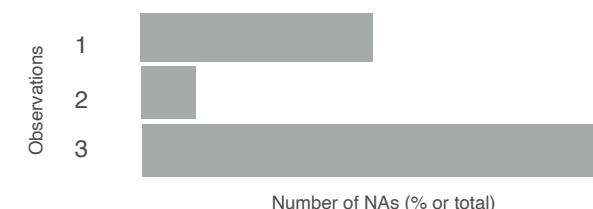
relative distribution

```
overview_na(toydata_with_na)
```

```
overview_na(toydata_with_na,
  perc = FALSE)
```

FALSE gives total number (instead of percentage)

It also allows you to plot the NAs row-wise. This can be helpful when looking at survey data where each respondent is represented in a row. Using this function then helps to understand the share of unit or item non-responses.



```
overview_na(toydata_with_na,
  row_wise = TRUE)
```

Plot NAs row-wise

```
overview_na(toydata_with_na,
  row_wise = TRUE,
  perc = FALSE)
```

FALSE gives total number (instead of percentage)