

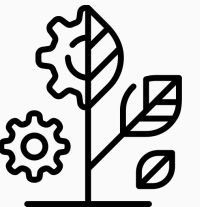
Tutorial 2

Data Visualization in R.

ggplot2: statistics, coordinate system, facets.

Victoria Mironova

Associate Professor, Department of Plant Systems Physiology



Course structure

Week 5:

Lecture 1. Principles of figure design.

Quiz 1.

Week 6:

Tutorial 1. ggplot2: plots and charts.

Quiz 2.

Week 7:

Tutorial 2. ggplot2: statistics, coordinate system, facets.

Tutorial 3. ggplot2: themes and styles.

Practice 1.

Quiz 3.

Week 8:

Practice 2. Project.

Practice 3. Project.

Practice 4. Project.

Assignment.

Course structure

Week 5:

Lecture 1. Principles of figure design.

Quiz 1.

Week 6:

Tutorial 1. ggplot2: plots and charts.

Quiz 2.

Week 7:

Tutorial 2. ggplot2: statistics, coordinate system, facets.

Tutorial 3. ggplot2: themes and styles.

Practice 1.

Quiz 3.

Week 8:

Practice 2. Project.

Practice 3. Project.

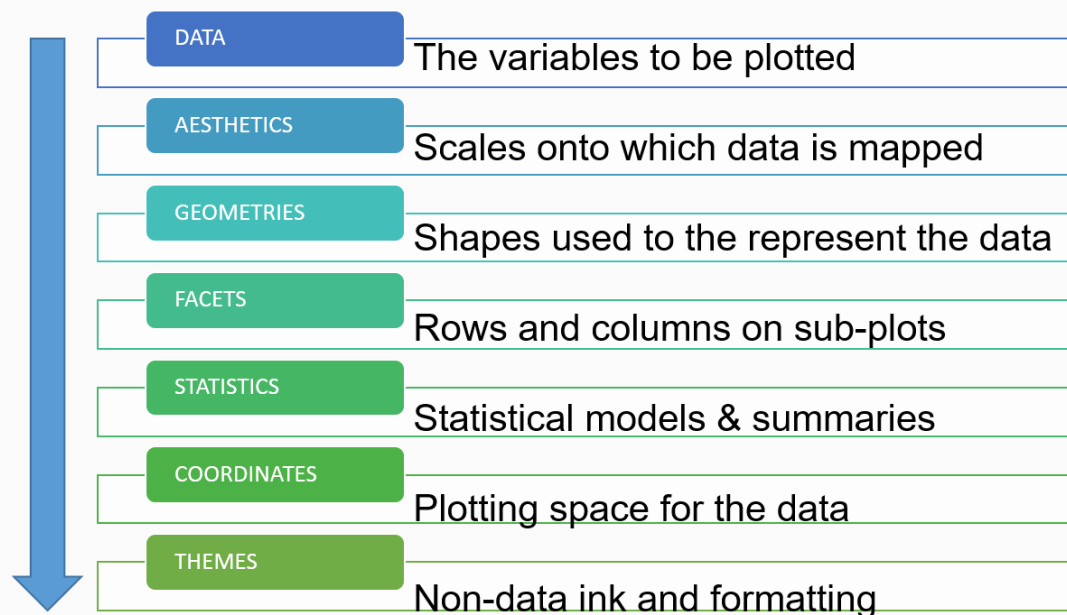
Practice 4. Project.

Assignment.

Learning goals

- Understand the basic principles behind effective data visualization
- **Create data visualizations in R using ggplot2**
- Craft elegant visual presentations of data

Grammar of graphics

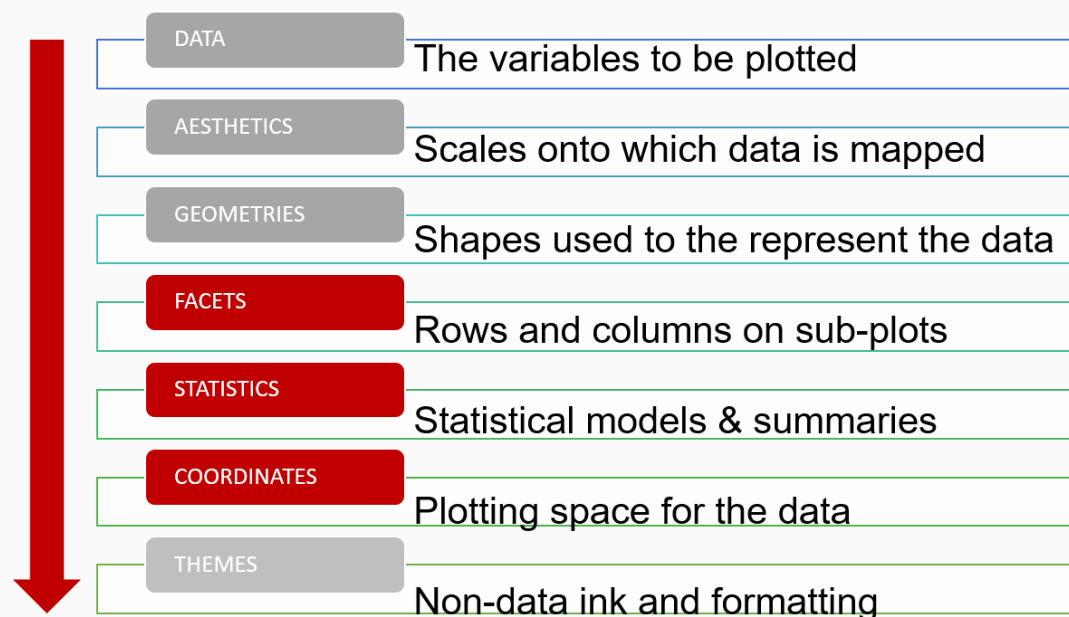


Layers in grammar of graphics

There are two important principles behind *grammar of graphics*:

- Graphics are made of distinct layers of grammatical elements
- Plots are built with appropriate aesthetic mappings to make these plots meaningful

Grammar of graphics



Layers in grammar of graphics

Coordinate system: What kind of a coordinate system should the visualization be based on — should it be cartesian or polar?

Facets: Do we need to create subplots based on specific data dimensions?

Statistics: Do we need to show some statistical measures in the visualization like measures of central tendency, spread, confidence intervals?

Plan of the tutorial

- Extended graphing template
- Coordinate system
- Faceting
- Statistics
- Save visualizations

Data

We keep working with the data on: Number of deaths in the population of the Netherlands by main underlying cause of death, by age and sex, 1996-2021.

| ID | Sex | Age | CausesOfDeath | Year | Deaths |
|---------|--------|-------|-------------------|------|--------|
| 2957560 | Female | 85-89 | Perinatal | 2004 | 0 |
| 1424313 | Male | 30-34 | Perinatal | 2003 | 0 |
| 1457025 | Male | 35-39 | CirculatorySystem | 2007 | 110 |
| 1377865 | Male | 25-29 | Perinatal | 2017 | 0 |
| 2327624 | Female | 20-24 | Neoplasms | 1996 | 27 |

- **ID**: Numeric, continuous;
- **Sex**: Categorical, unordered;
- **Age**: Categorical, ordered;
- **CausesOfDeath**: categorical, unordered;
- **Year**: Numeric, discreet;
- **Deaths**: Numeric, continuous.

Updated graphing template

On the last tutorial you saw an incomplete template highlighted in yellow. Below is a complete one.

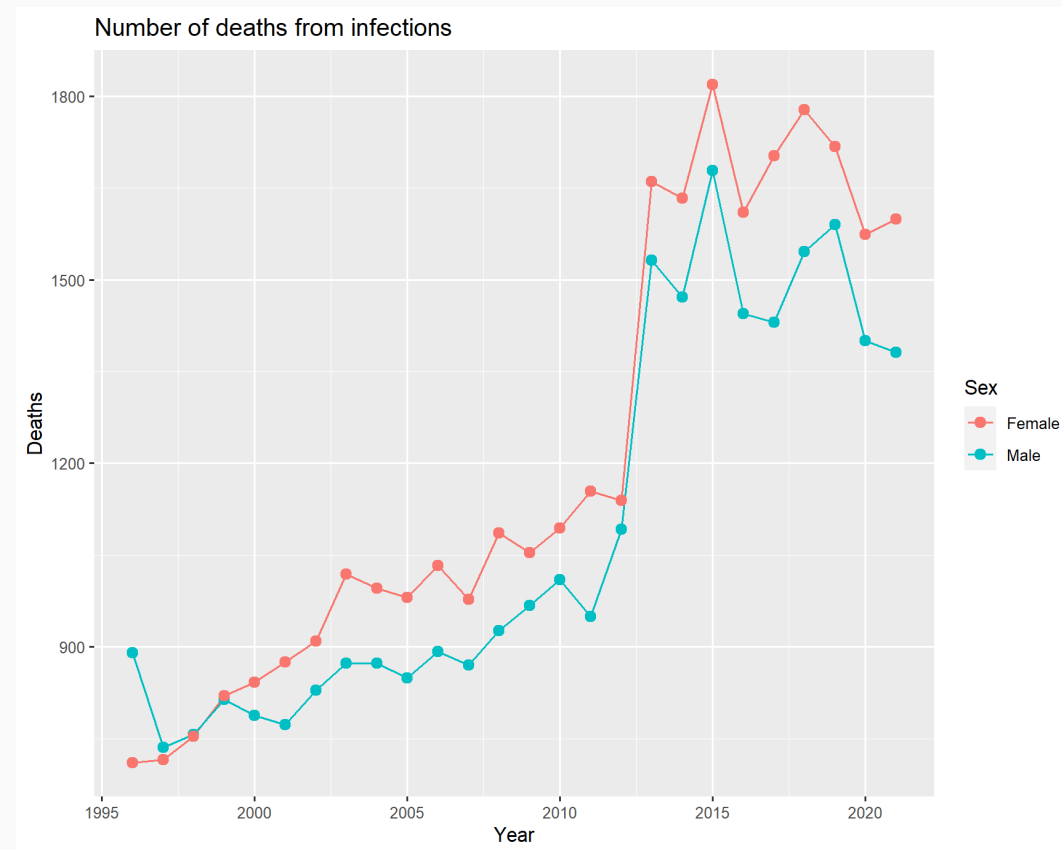
```
ggplot(data = <DATA>)+  
  <GEOM_FUNCTION>(  
    mapping = aes(<MAPPINGS>),  
    method = <STAT>,  
    position = <POSITION>) +  
  <STAT_FUNCTION>()+  
  <COORDINATE_FUNCTION>()+  
  <FACET_FUNCTION>()+  
  <THEME>()
```


Coordinate systems

Cartesian Coordinates

When we plot a chart using ggplot2, Cartesian Coordinates is the default coordinate system.

```
Death_in_NL %>%  
  filter(Age == "Total", CausesOfDeath == "Infections")  
  ggplot(mapping = aes(x = Year, y = Deaths, color = Sex))  
  geom_path()+  
  geom_point(size = 2.5)+  
  ggtitle("Number of deaths from infections")
```

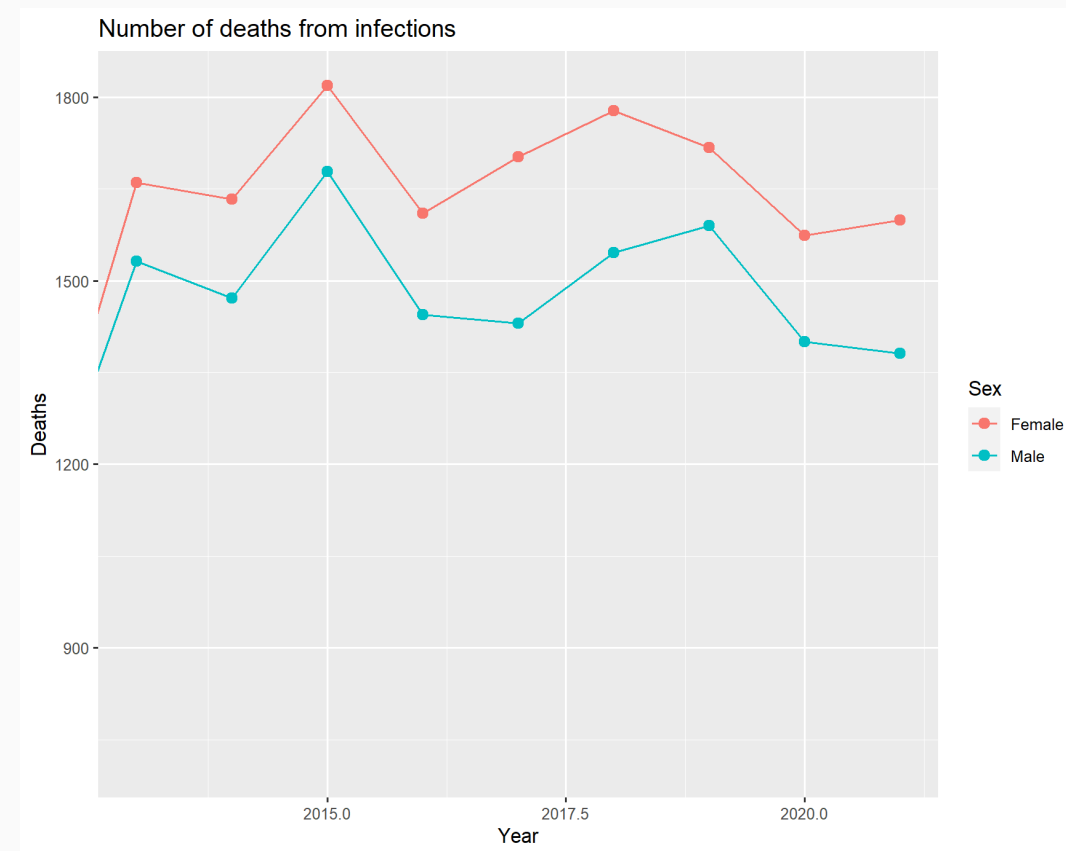


Cartesian coordinates: zooming in

When we plot a chart using ggplot2, Cartesian Coordinates is the default coordinate system.

```
Death_in_NL %>%  
  filter(Age = "Total", CausesOfDeath = "Infections")  
  ggplot(aes(x = Year, y = Deaths, color = Sex))+  
  geom_path()+  
  geom_point(size = 2.5)+  
  coord_cartesian(xlim = c(2013, 2021))+  
  ggtitle("Number of deaths from infections")
```

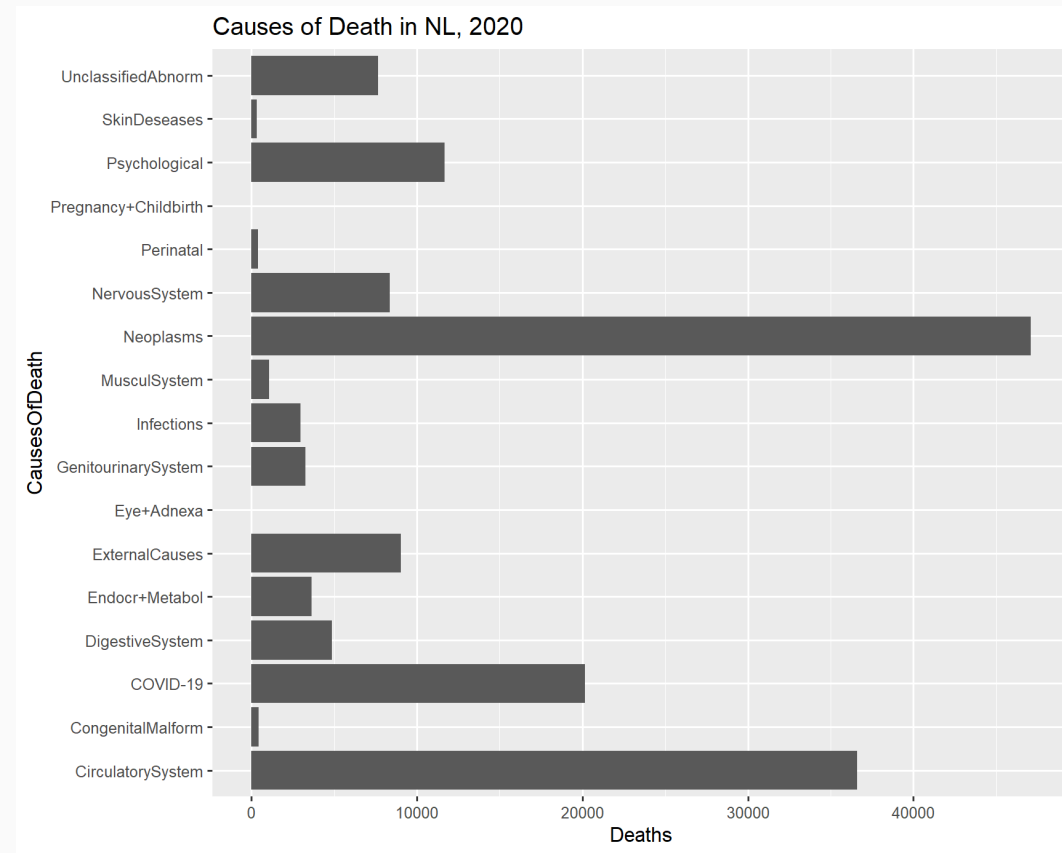
But you can still set it specifying parameters, for example, to zoom in the plot display.



Flipped cartesian coordinates

It is convenient to flip the axes when the name of the groups on the x axis is long.

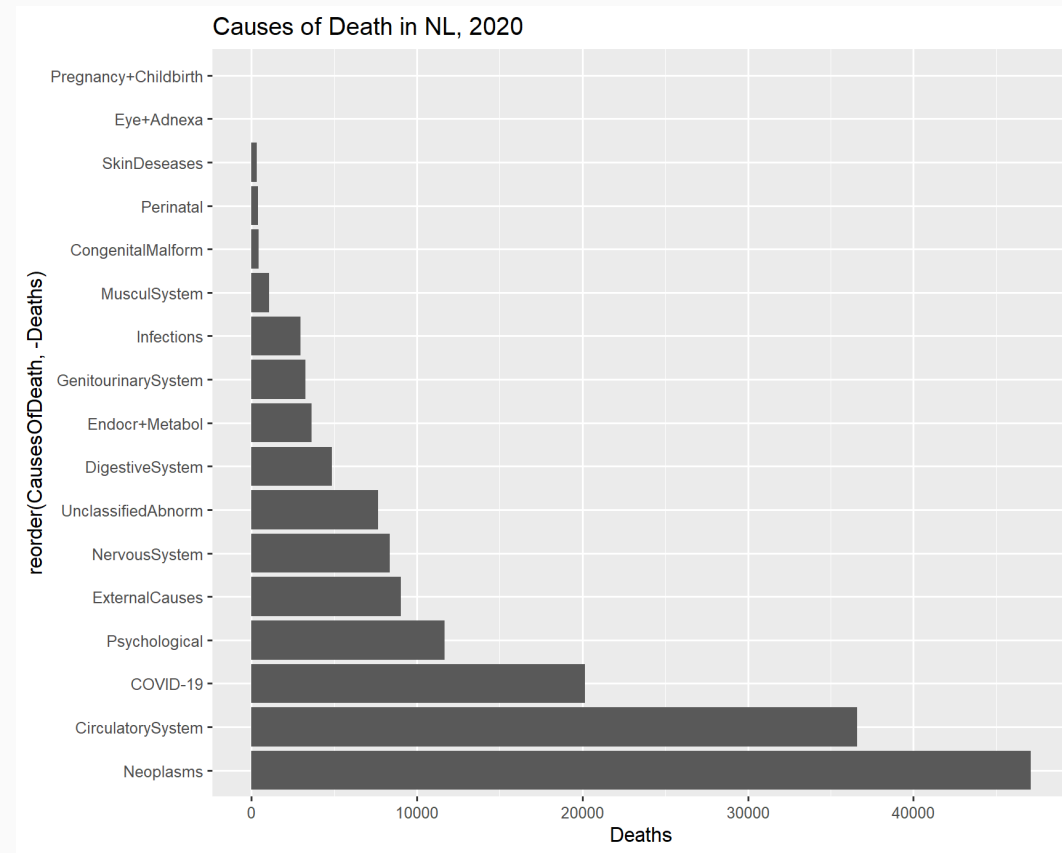
```
Death_in_NL%>%  
  filter(Year = 2020, Age = "Total")%>%  
  ggplot(aes(x = CausesOfDeath, y = Deaths))+  
  geom_col()+  
  coord_flip()+  
  ggtitle("Causes of Death in NL, 2020")
```



Change the order of the groups on the axis

You might also need to change the order of unordered categorical data on the axis.

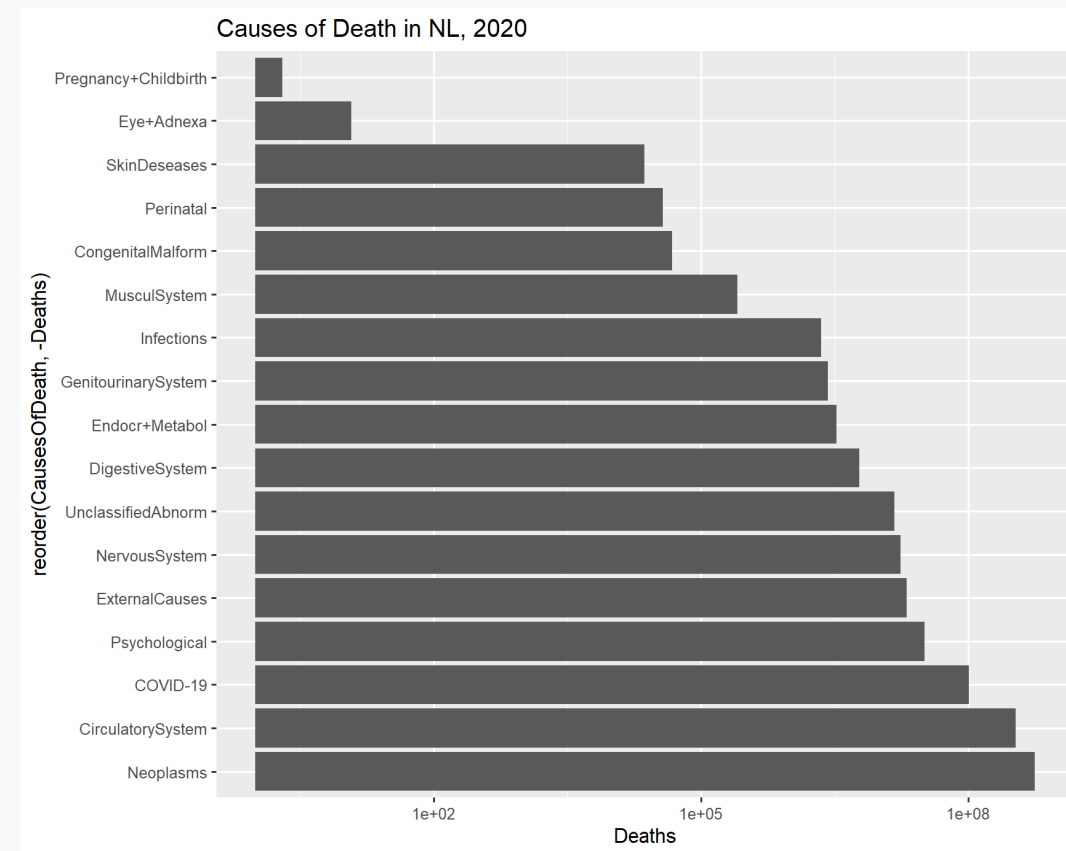
```
Death_in_NL%>%  
  filter(Year = 2020, Age = "Total")%>%  
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [  
    geom_col()+  
    coord_flip()+  
    ggtitle("Causes of Death in NL, 2020")
```



Change an axis scale

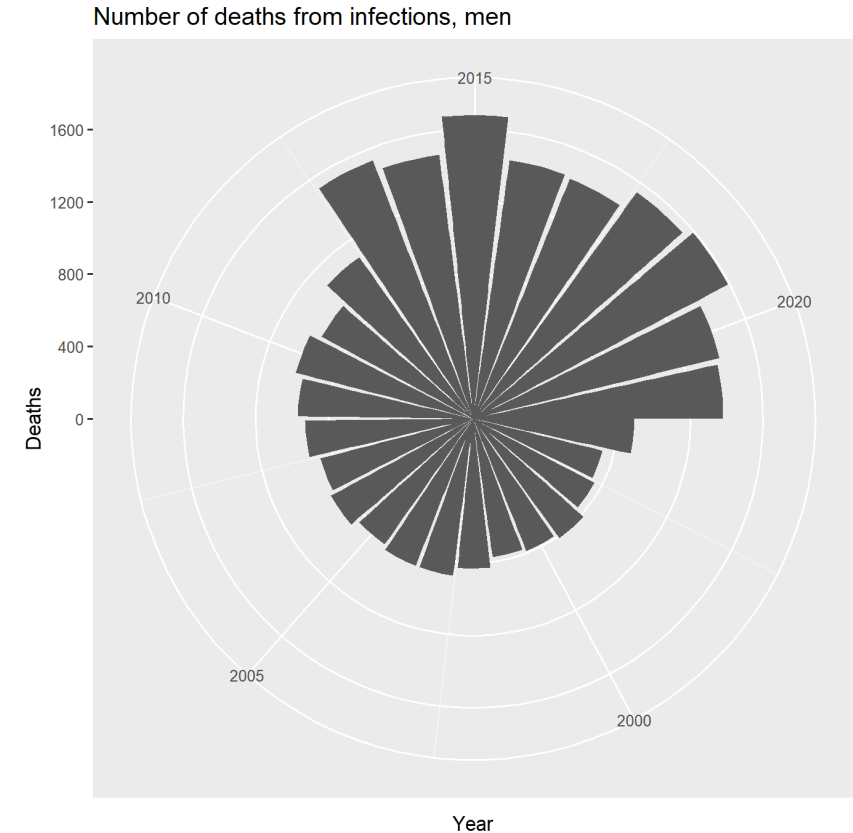
```
Death_in_NL%>%  
  filter(Year = 2020, Age = "Total")%>%  
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [  
    geom_col()+  
    coord_flip()+  
    scale_y_log10()+  
    ggtitle("Causes of Death in NL, 2020")
```

You can learn more about specifying the axes [here](#)



Polar coordinates

```
Death_in_NL %>%  
  filter(Age = "Total", Sex = "Male", CausesOfDeath =  
    ggplot(aes(x = Year, y = Deaths))+  
    geom_col()+  
    coord_polar(start = pi/2)+  
    ggtitle("Number of deaths from infections, men")
```

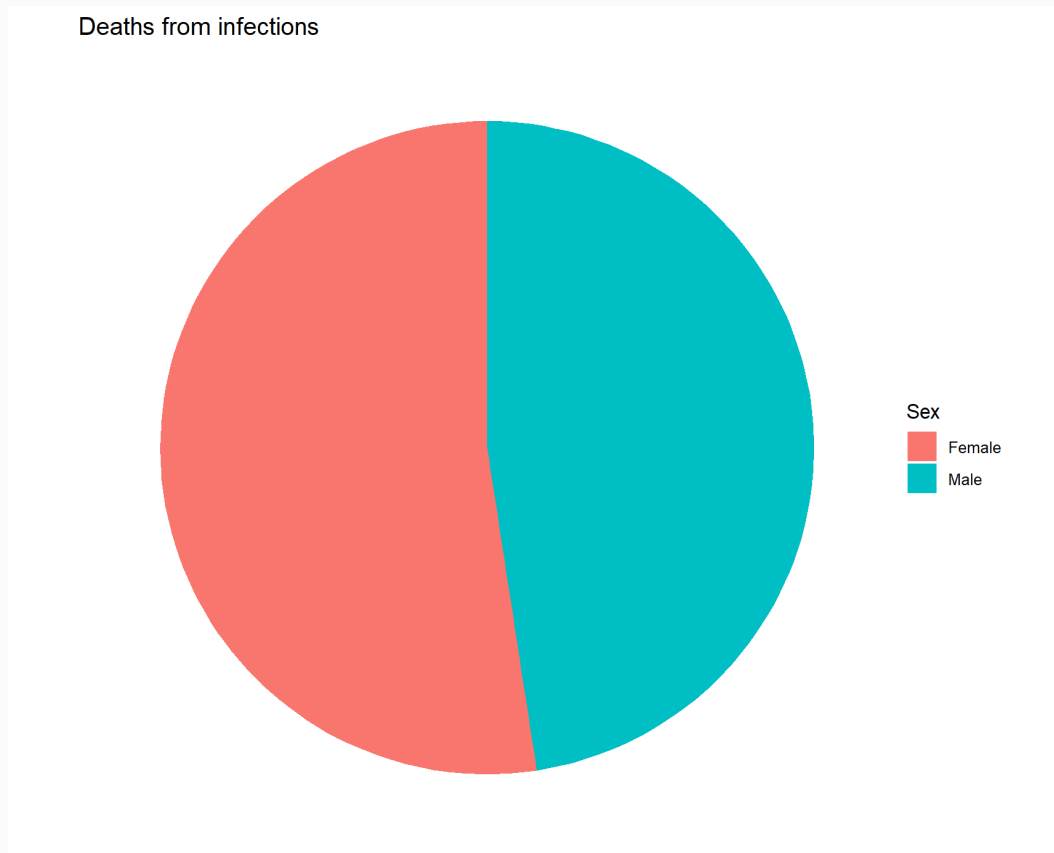


Polar coordinate system: pie chart

We use polar coordinate system to build pie charts

```
Death_in_NL %>%  
  filter(Age == "Total", CausesOfDeath == "Infections")  
  ggplot(aes(x = "", y = Deaths, fill = Sex))+  
  geom_col()+  
  coord_polar("y", start = 0)+  
  ggtitle("Deaths from infections")+  
  theme_void() # remove background, grid, numeric labels
```

Here you can learn how to add the labels on the pie chart.



Facetted graphs

Vizualizing data slices, line plot

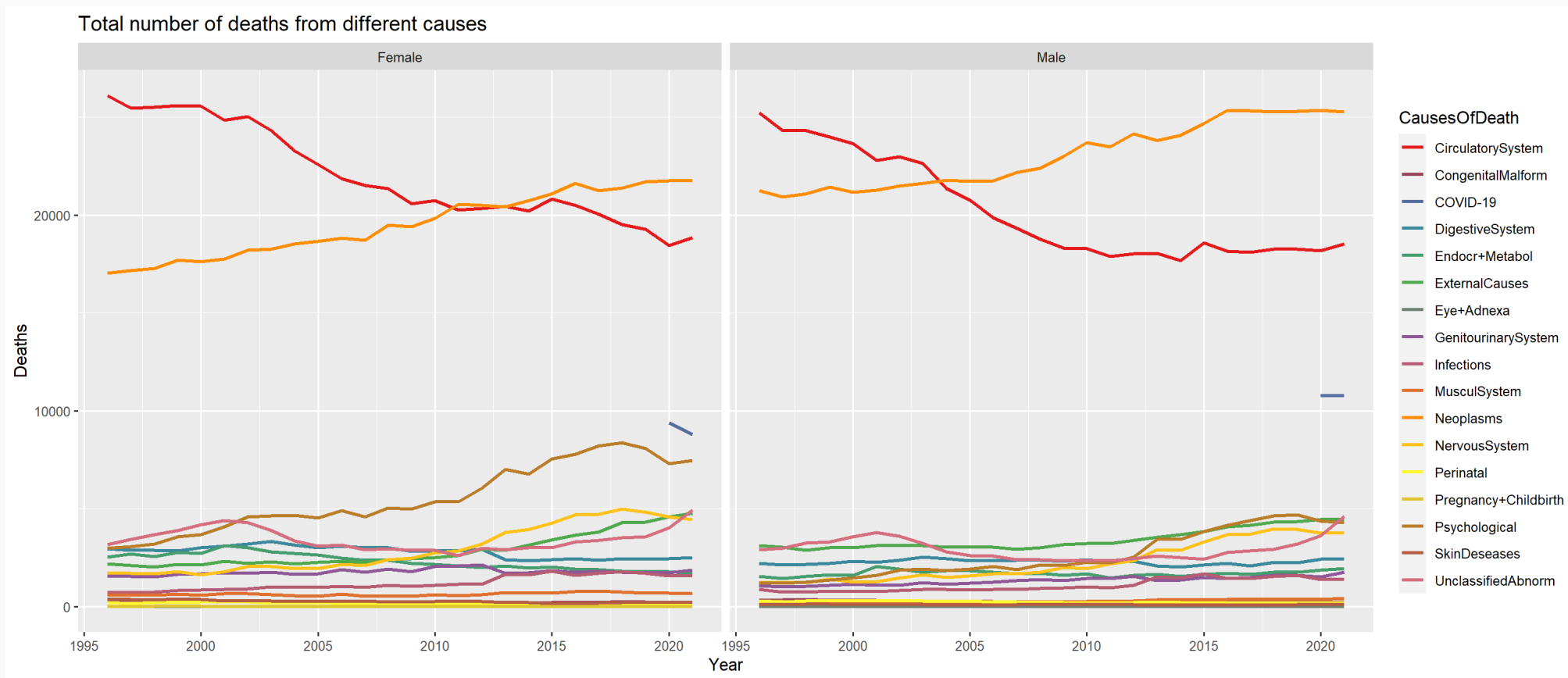
| Code | Output |
|------|--------|
|------|--------|

```
nb.cols <- 18
mycolors <- colorRampPalette(brewer.pal(8, "Set1"))(nb.cols)

Death_in_NL %>%
  filter(Age == "Total")%>%
  ggplot(aes(x = Year, y = Deaths, color = CausesOfDeath))+
  geom_path(linewidth = 1)+
  scale_colour_manual(values = mycolors)+
  ggtitle("Total number of deaths from different causes")+
  facet_wrap( ~ Sex)
```

Vizualizing data slices, line plot

Code Output



Vizualizing data slices, heatmap

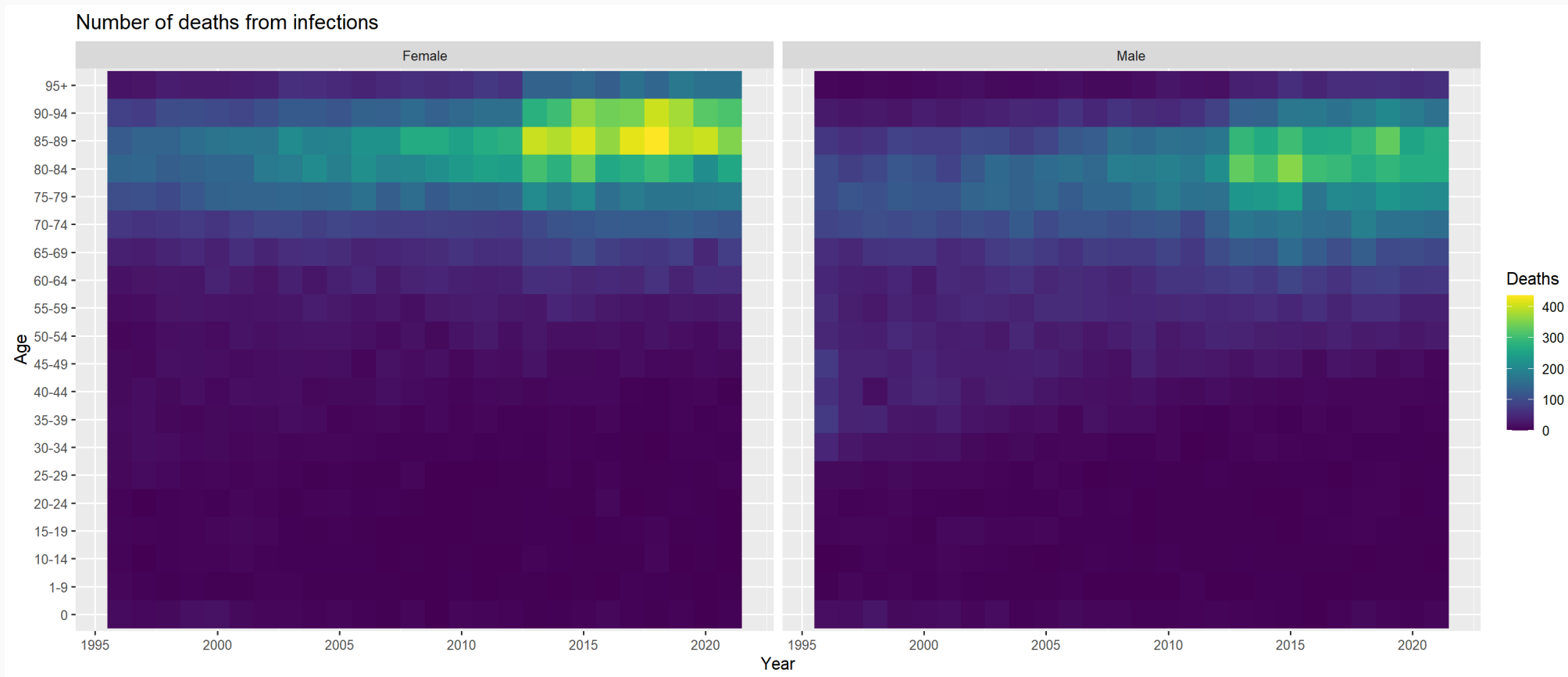
| Code | Output |
|------|--------|
|------|--------|

```
Death_in_NL %>%  
  filter(CausesOfDeath == "Infections", Age != "Total") %>%  
  ggplot(mapping = aes(x = Year, y = Age, fill = Deaths))+  
  geom_tile()+  
  scale_fill_viridis(discrete = FALSE)+  
  ggtitle("Number of deaths from infections")+  
  facet_wrap(~Sex)
```

Vizualizing data slices, heatmap

Code

Output



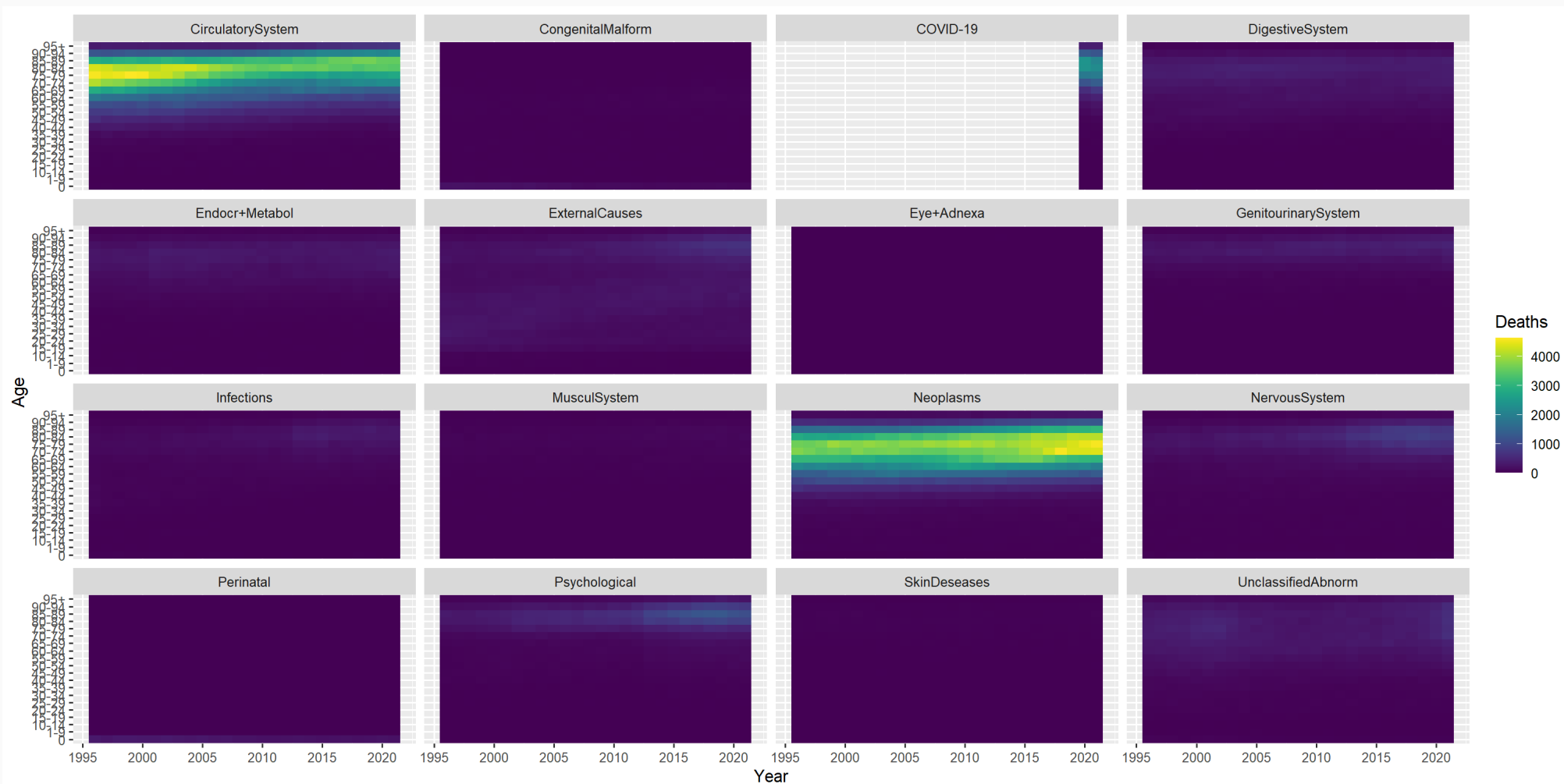
One can generate many facets of data

| Code | Output |
|------|--------|
|------|--------|

```
Death_in_NL %>%  
  filter(Sex = "Male", Age ≠ "Total") %>%  
  ggplot(mapping = aes(x = Year, y = Age, fill = Deaths))+  
  geom_tile()+  
  scale_fill_viridis(discrete = FALSE)+  
  facet_wrap(~CausesOfDeath)
```

One can generate many facets of data

Code Output



Arrange multiple plots in a grid

You can use other packages to organize individual plots altogether. E.g `plot_grid` from `cowplot` package.

| Code | Output |
|------|--------|
|------|--------|

```
smoothp <- Death_in_NL %>%  
  filter(CausesOfDeath == "Infections", Age == "Total") %>%  
  ggplot(mapping = aes(x = Year, y = Deaths, color = Sex))+  
  geom_smooth()+  
  ggtitle("Smoothed conditional means")
```

```
boxp <- Death_in_NL %>%  
  filter(Age == "Total") %>%  
  ggplot(mapping = aes(x = Sex, y = Deaths, fill = Sex))+  
  geom_boxplot()+  
  ggtitle("Boxplot")
```

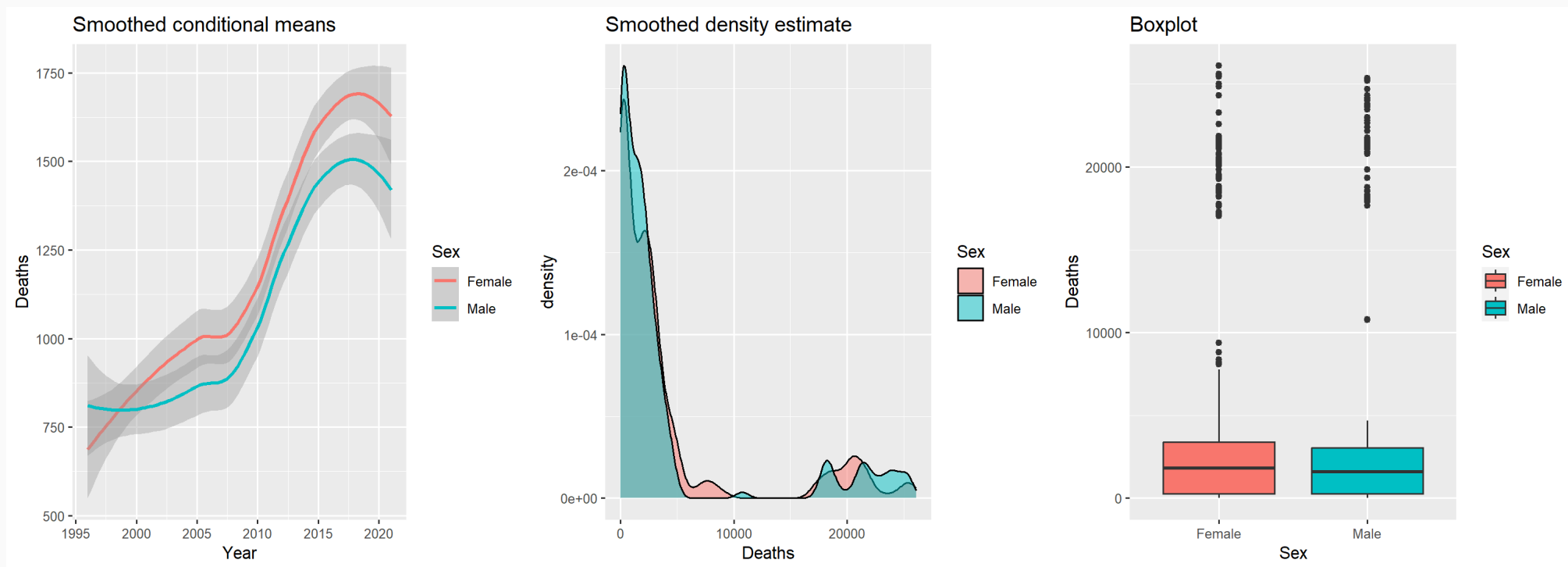
```
denp <- Death_in_NL %>%  
  filter(Age == "Total") %>%  
  ggplot(aes(x = Deaths, fill = Sex))+  
  geom_density(alpha = 0.5)+  
  ggtitle("Smoothed density estimate")
```

```
plot_grid(smoothp, denp, boxp, ncol = 3, label_size = 12)
```


Arrange multiple plots in a grid

You can use other packages to organize individual plots altogether. E.g `plot_grid` from `cowplot` package.

Code Output



Statistics layer

Two categories of functions:

- called from within a geom
- called independently

Statistics layer

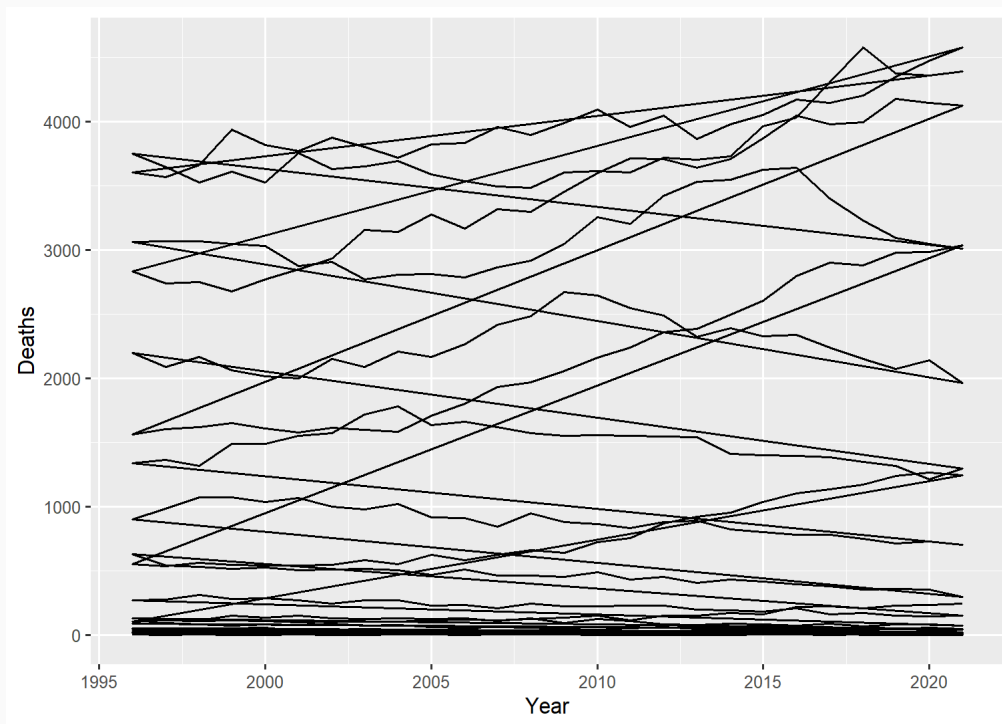
Two categories of functions:

- called from within a geom
- called independently

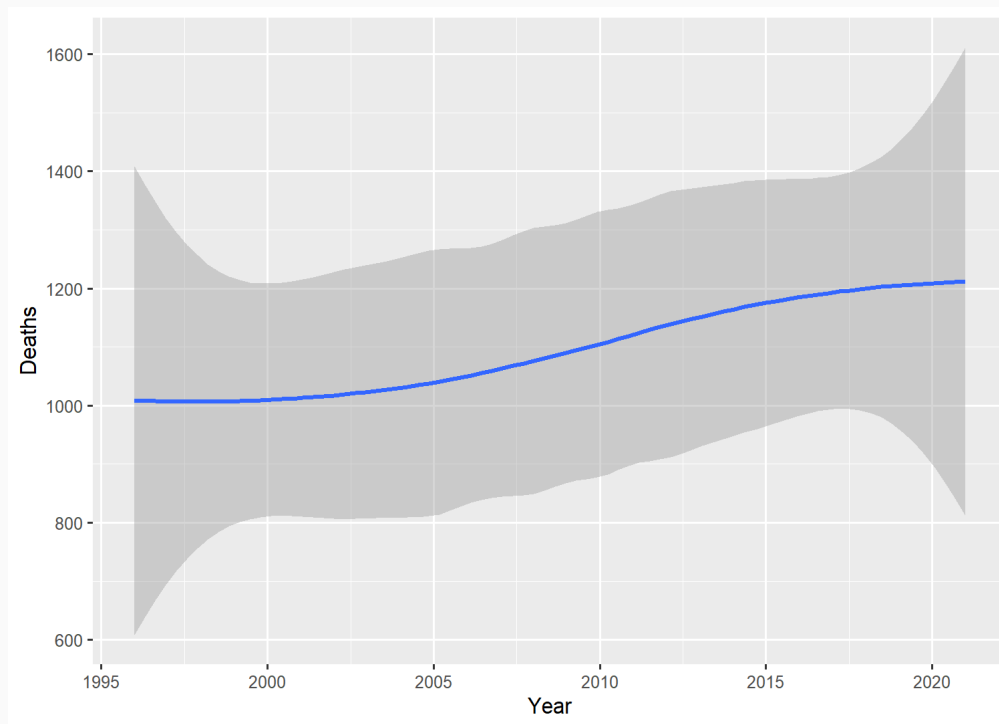
All the statistics functions called independently starts with *stat_*

Some geoms have built-in statistical methods

```
Death_in_NL%>%  
  filter(CausesOfDeath == "Neoplasms", Sex == "Male", /  
  ggplot(aes(x = Year, y = Deaths))+  
    geom_path()
```

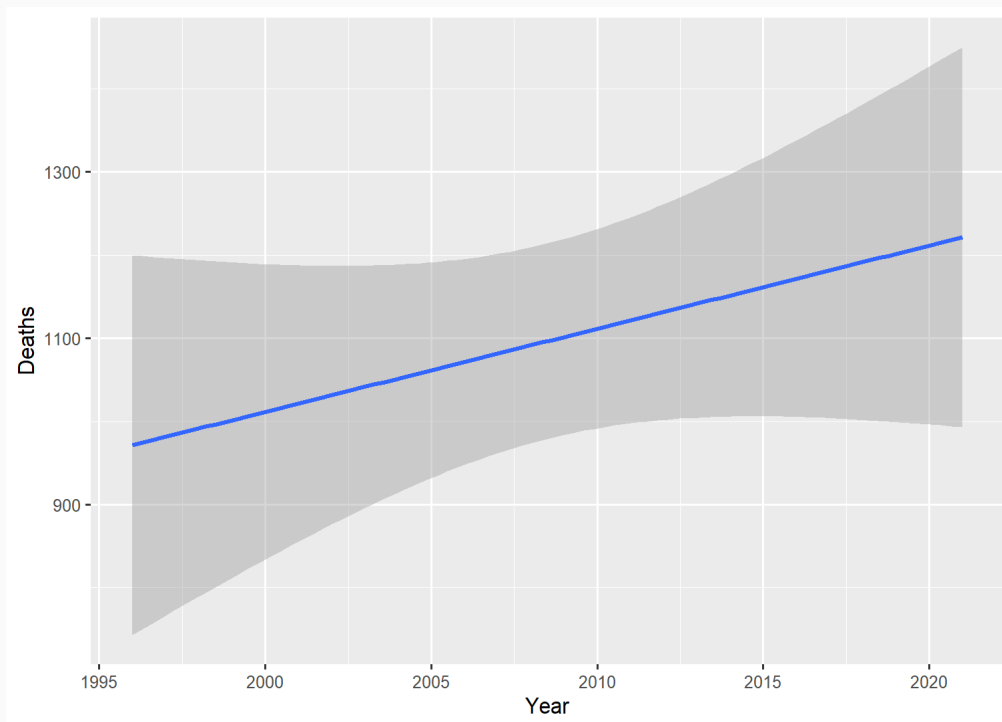


```
Death_in_NL%>%  
  filter(CausesOfDeath == "Neoplasms", Sex == "Male", /  
  ggplot(aes(x = Year, y = Deaths))+  
    geom_smooth()
```

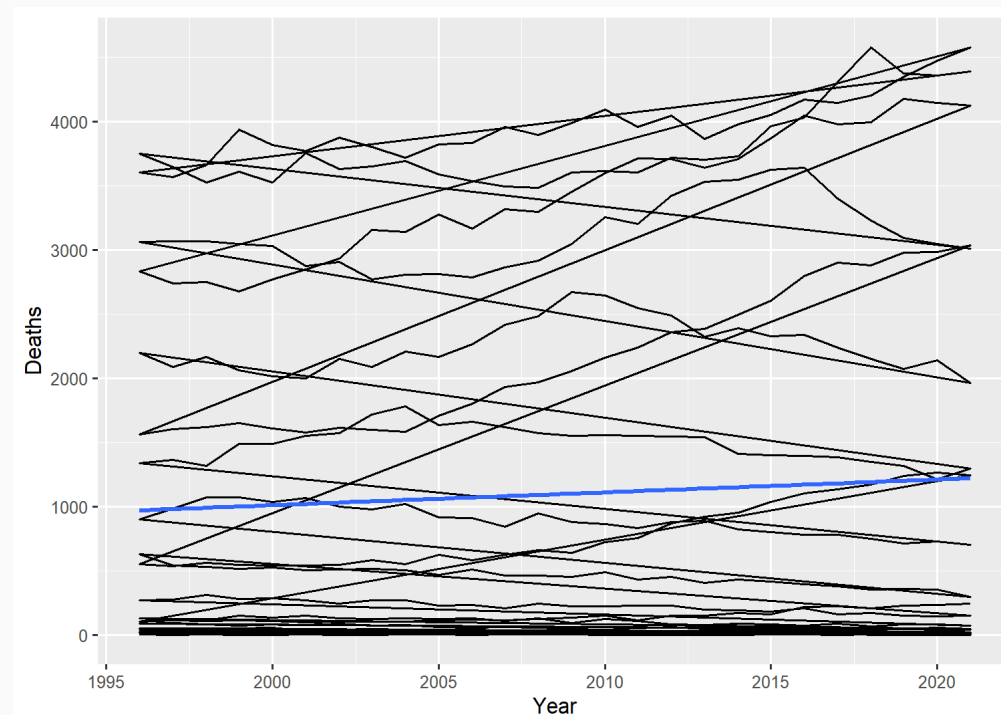


Adjusting statistical methods in geom

```
Death_in_NL%>%  
  filter(CausesOfDeath == "Neoplasms", Sex == "Male", /  
  ggplot(aes(x = Year, y = Deaths))+  
    geom_smooth(method = "lm",  
               formula = y ~ x)
```



```
Death_in_NL%>%  
  filter(CausesOfDeath == "Neoplasms", Sex == "Male", /  
  ggplot(aes(x = Year, y = Deaths))+  
    geom_path()+  
    geom_smooth(method = "lm", se = FALSE)
```



Showing statistical estimates on a plot

Lets draw a bar plot with error bars using `geom_errorbar()` function. Lets use the annual data as independent observations.

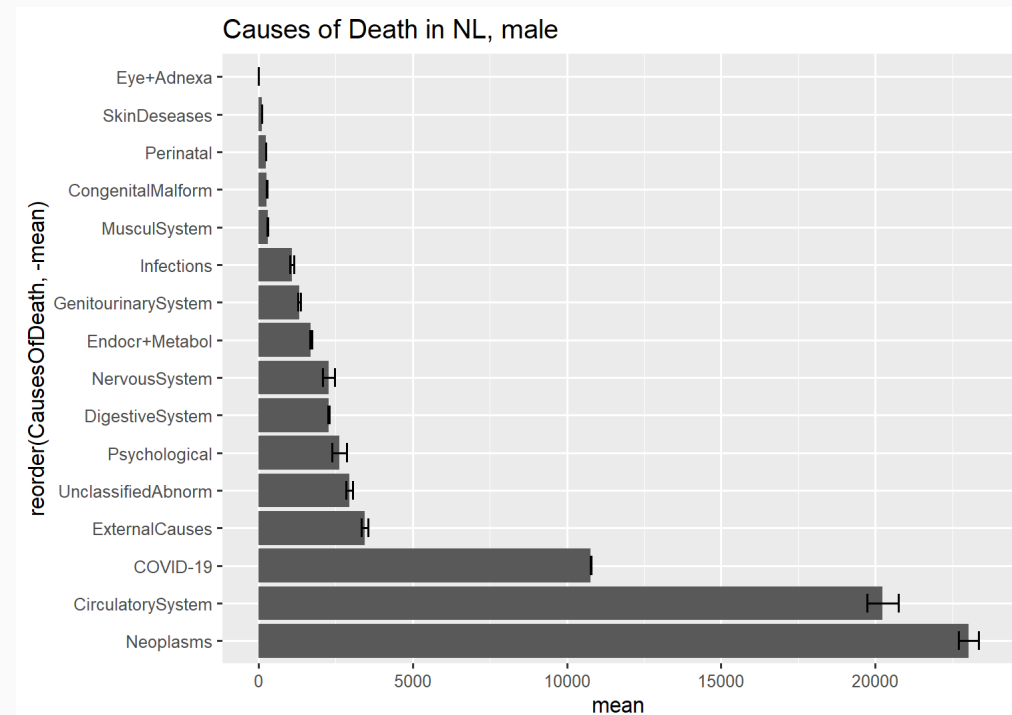
Note: the lower and upper limits of your error bars must be computed before building the chart, and available in a column of the input data.

```
data_stat <- Death_in_NL%>%  
  filter(Sex == "Male", Age == "Total")%>%  
  group_by(CausesOfDeath)%>%  
  summarize(mean = mean(Deaths),  
            se = sd(Deaths)/sqrt(n()),  
            max = mean + se,  
            min = mean - se,  
            N = n())
```

| CausesOfDeath | mean | se | max | min | N |
|---------------------|--------------|------------|-------------|--------------|----|
| CirculatorySystem | 20239.038462 | 502.475639 | 20741.51410 | 19736.562823 | 26 |
| CongenitalMalform | 266.538461 | 8.748139 | 275.28660 | 257.790322 | 26 |
| COVID-19 | 10772.500000 | 8.500000 | 10781.00000 | 10764.000000 | 2 |
| DigestiveSystem | 2284.076923 | 24.724570 | 2308.80149 | 2259.352353 | 26 |
| Endocr+Metabol | 1702.923077 | 29.283649 | 1732.20673 | 1673.639428 | 26 |
| ExternalCauses | 3440.961539 | 105.080445 | 3546.04198 | 3335.881093 | 26 |
| Eye+Adnexa | 2.538461 | 0.509089 | 3.04755 | 2.029373 | 26 |
| GenitourinarySystem | 1324.153846 | 39.444048 | 1363.59789 | 1284.709798 | 26 |
| Infections | 1090.884615 | 61.835970 | 1152.72059 | 1029.048646 | 26 |
| MusculSystem | 297.923077 | 11.546151 | 309.46923 | 286.376926 | 26 |
| Neoplasms | 23028.653846 | 325.132593 | 23353.78644 | 22703.521253 | 26 |
| NervousSystem | 2282.615385 | 197.962095 | 2480.57748 | 2084.653290 | 26 |
| Perinatal | 239.423077 | 8.353935 | 247.77701 | 231.069142 | 26 |
| Psychological | 2620.461539 | 235.839275 | 2856.30081 | 2384.622263 | 26 |
| SkinDeseases | 108.384615 | 3.596185 | 111.98080 | 104.788431 | 26 |
| UnclassifiedAbnorm | 2941.307692 | 110.288486 | 3051.59618 | 2831.019207 | 26 |

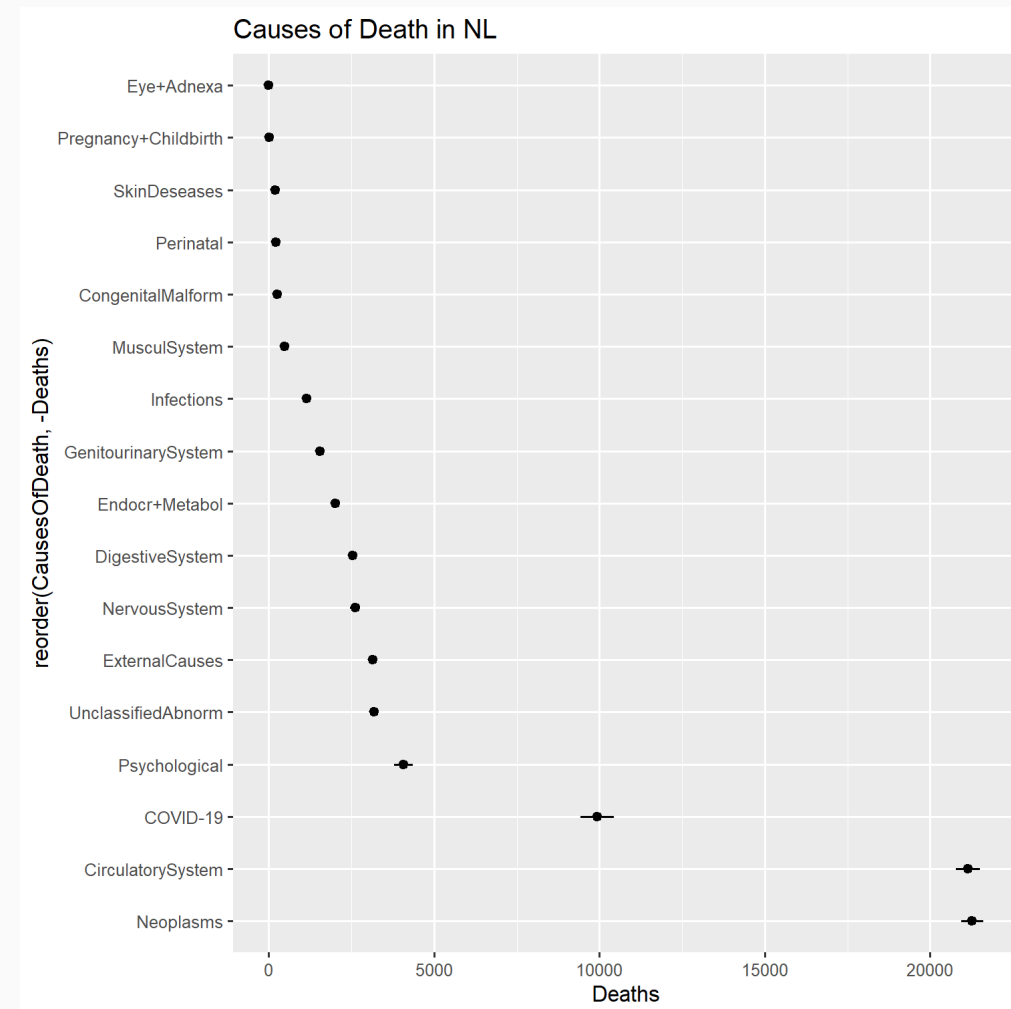
Bar Plot with error bars

```
data_stat%>%  
  ggplot(aes(x = reorder(CausesOfDeath, -mean), y = mean)) +  
  geom_col() +  
  coord_flip() +  
  geom_errorbar(width = 0.5) +  
  ggtitle("Causes of Death in NL, male")
```



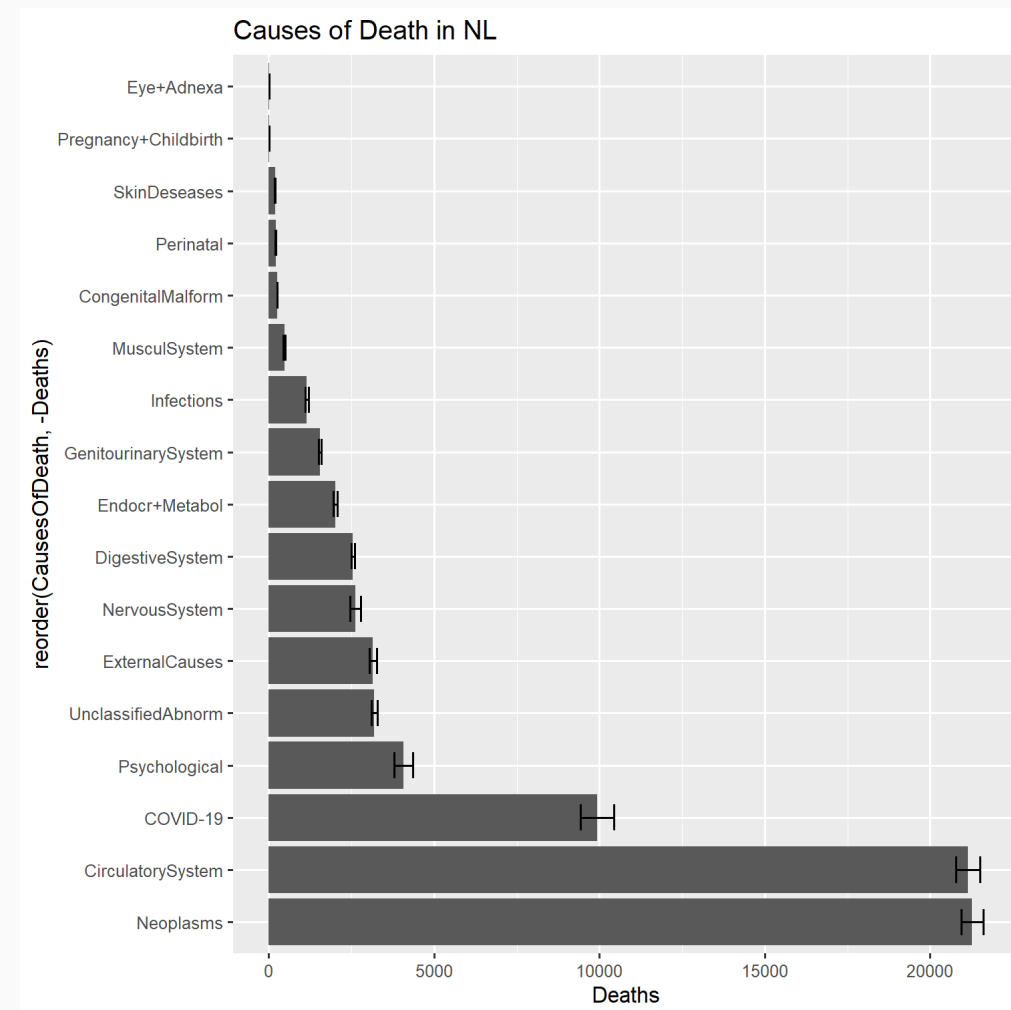
Use of statistical functions in ggplot2

```
Death_in_NL%>%  
  filter(Age == "Total")%>%  
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [  
    stat_summary(size = 0.3)+  
    coord_flip()+  
    ggtitle("Causes of Death in NL")
```



Use of `stat_summary` in `ggplot2`

```
Death_in_NL%>%  
  filter(Age == "Total")%>%  
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [  
    stat_summary(fun.data = "mean_se", geom = "bar")+  
    stat_summary(fun.data = "mean_se", geom = "errorbar",  
    coord_flip()+  
    ggtitle("Causes of Death in NL")
```



Saving the plot

You can use `ggsave()` to save the visualization in the format and resolution you wish.

```
?ggsave
```

```
ggsave(  
  filename,  
  plot = last_plot(),  
  device = NULL,  
  path = NULL,  
  scale = 1,  
  width = NA,  
  height = NA,  
  units = c("in", "cm", "mm", "px"),  
  dpi = 300,  
  limitsize = TRUE,  
  bg = NULL,  
  ...  
)
```

```
ggsave(filename= "figs/BarPlotSD.png", device = png, width = 10, height = 10, units = "cm")
```

Your turn

Using the `Death_in_NL` dataset, build the following two visualizations:

- 1) What is the age profile of the number of deaths from external causes (not a disease) in NL? Is there any difference between men and women?
- 2) How has the number of perinatal deaths in the Netherlands changed over time?

You will need this information to complete the quiz in the brightspace (to be done after Tutorial 3).

