

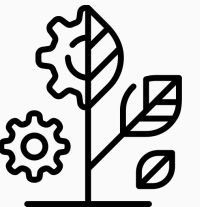
Tutorial 3

Data Visualization in R.

ggplot2: themes and styles.

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.

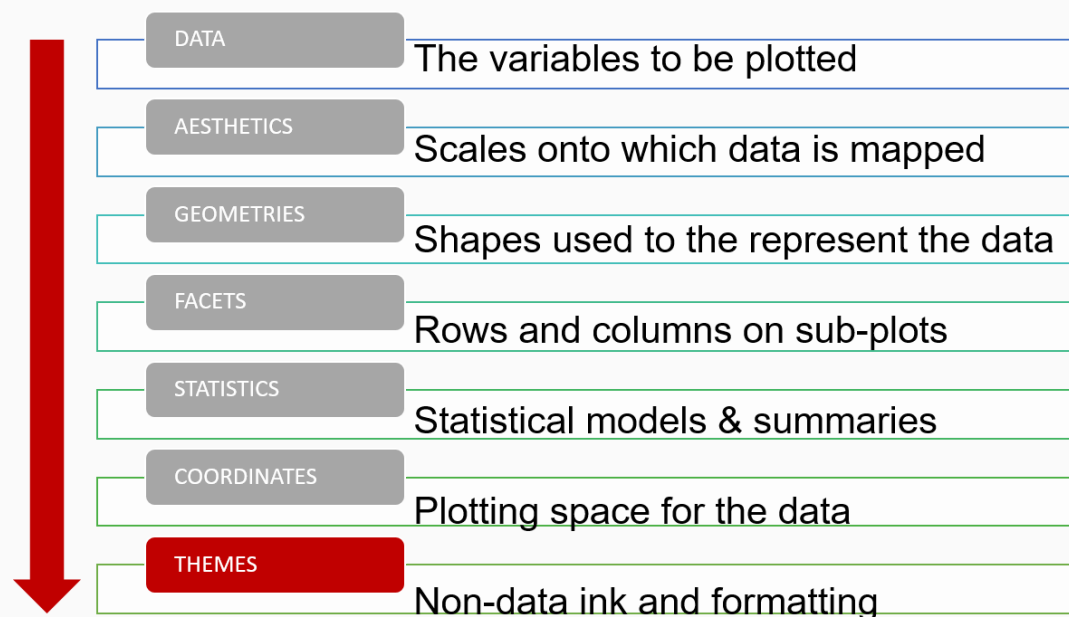
Practice 4. Project.

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

The **Themes Layer** refers to all non-data ink.

- You can change the labels of x or y axis,
- add a plot title,
- modify a legend title and position,
- add text anywhere on the plot,
- change the background color, axis lines, plot lines,
- etc.

Plan of the tutorial

- Themes principles
- Build-in themes
- Data annotation
- Customized design 1
- Customized design 2

?Theme()

There are three types of elements within the themes Layer: text, line, and rectangle.

Any text element can be modified with `element_text()`

Any line element can be modified with `element_line()`

Any rect element can be modified with `element_rect()`

One can also use `element_blank()` function to drop the element.

Build-in themes

For the most part you can probably avoid the `theme()` function by using built-in themes, unless there is a specific element you want to modify.

Code	Output
------	--------

```
p <- Death_in_NL %>%
  filter(Age = "Total", CausesOfDeath = "Infections")%>%
  ggplot(mapping = aes(x = Year, y = Deaths, linetype = Sex))+
  geom_path(show.legend = FALSE)

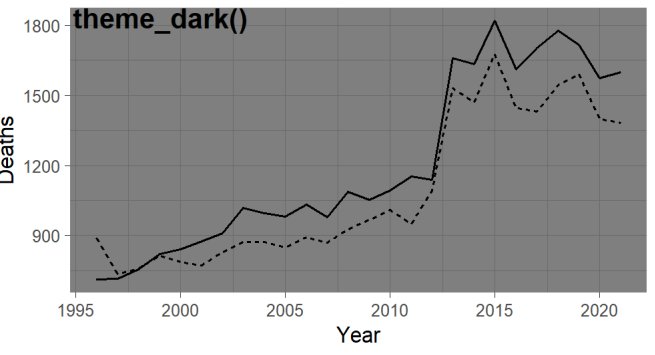
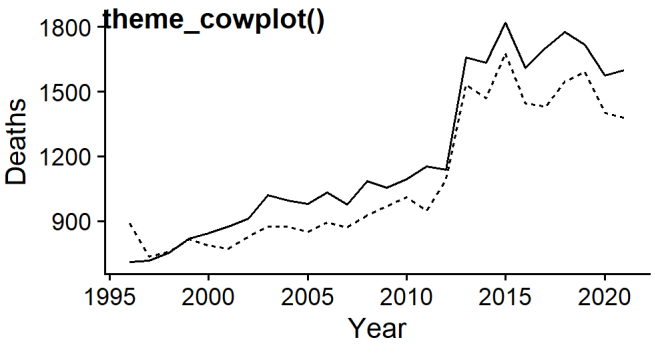
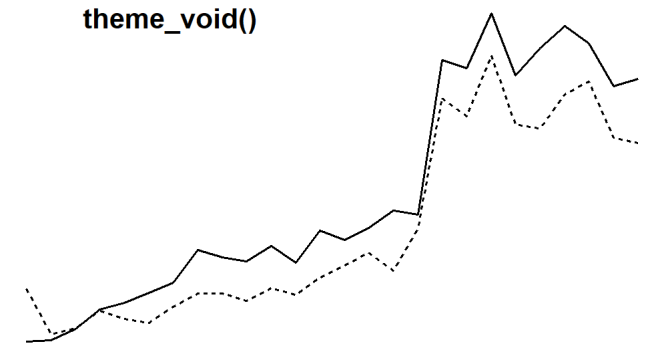
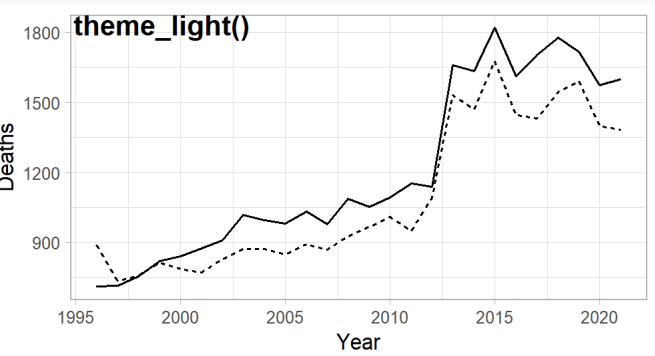
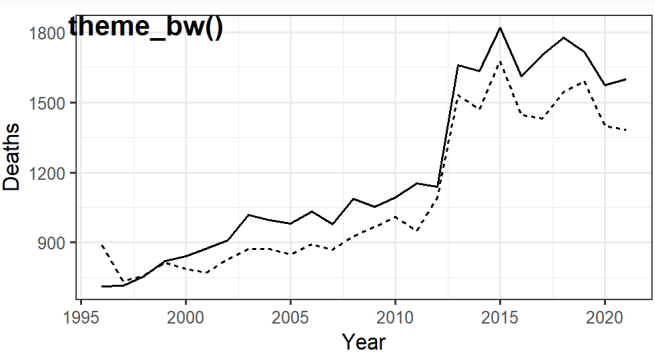
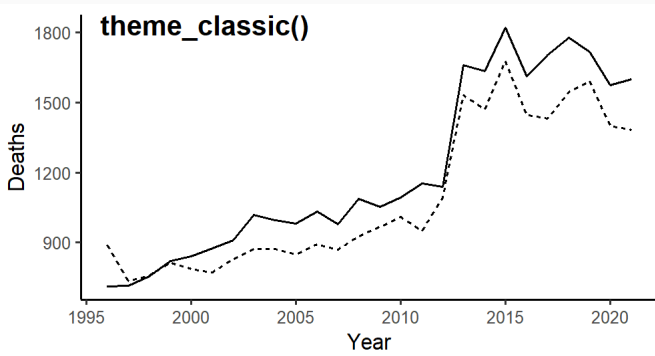
p1 <- p + theme_classic()
p2 <- p + theme_bw()
p3 <- p + theme_light()
p4 <- p + theme_void()
p5 <- p + theme_cowplot()
p6 <- p + theme_dark()

plot_grid(p1, p2, p3, p4, p5, p6, ncol = 3, labels = c("theme_classic()", "theme_bw()", "theme_light()", "theme_void(
```

Build-in themes

For the most part you can probably avoid the `theme()` function by using built-in themes, unless there is a specific element you want to modify.

Code	Output
<pre>theme_classic()</pre>	
<pre>theme_bw()</pre>	
<pre>theme_light()</pre>	
<pre>theme_void()</pre>	
<pre>theme_cowplot()</pre>	
<pre>theme_dark()</pre>	

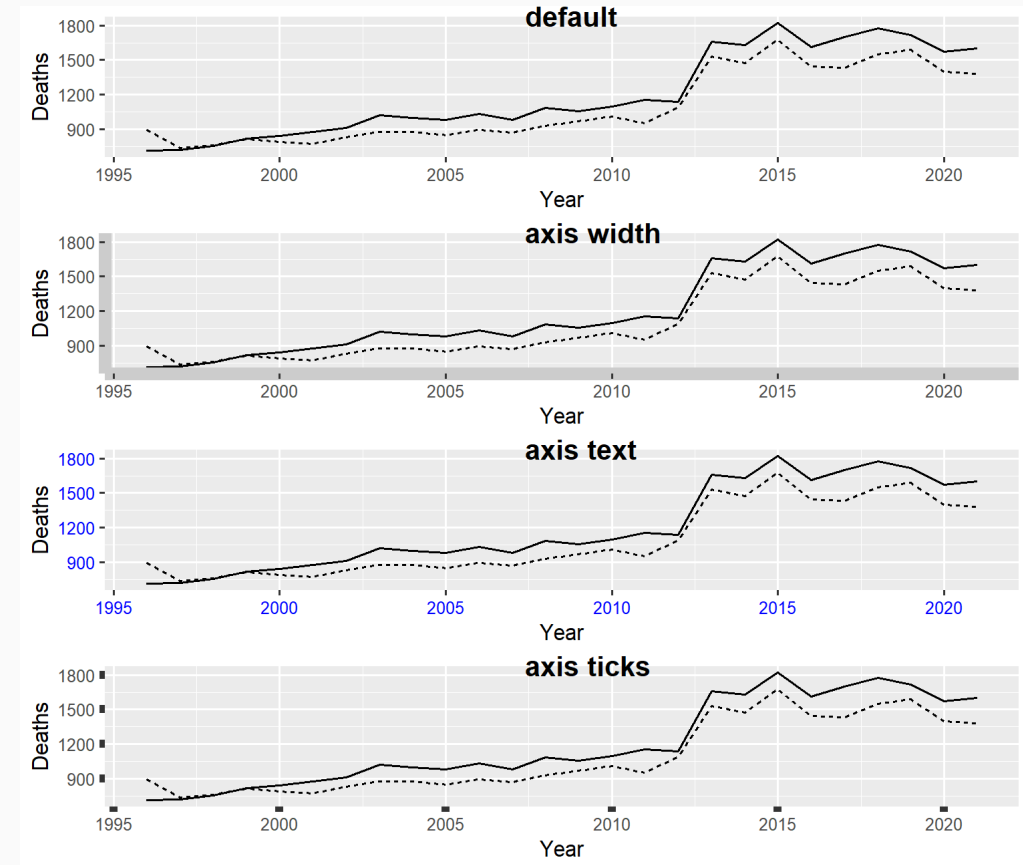


Customised theme: change styles of axes texts and

```
p <- Death_in_NL %>%  
  filter(Age == "Total", CausesOfDeath == "Infections")  
ggplot(mapping = aes(x = Year, y = Deaths, linetype =  
  geom_path(show.legend = FALSE)
```

```
p1 <- p + theme(axis.line = element_line(linewidth = 3,  
p2 <- p + theme(axis.text = element_text(colour = "blue"  
p3 <- p + theme(axis.ticks = element_line(linewidth = 2
```

```
plot_grid(p, p1, p2, p3, ncol = 1, labels = c("default"  
  label_x = 0.5, label_y = 1, hjust = 0, vjust
```

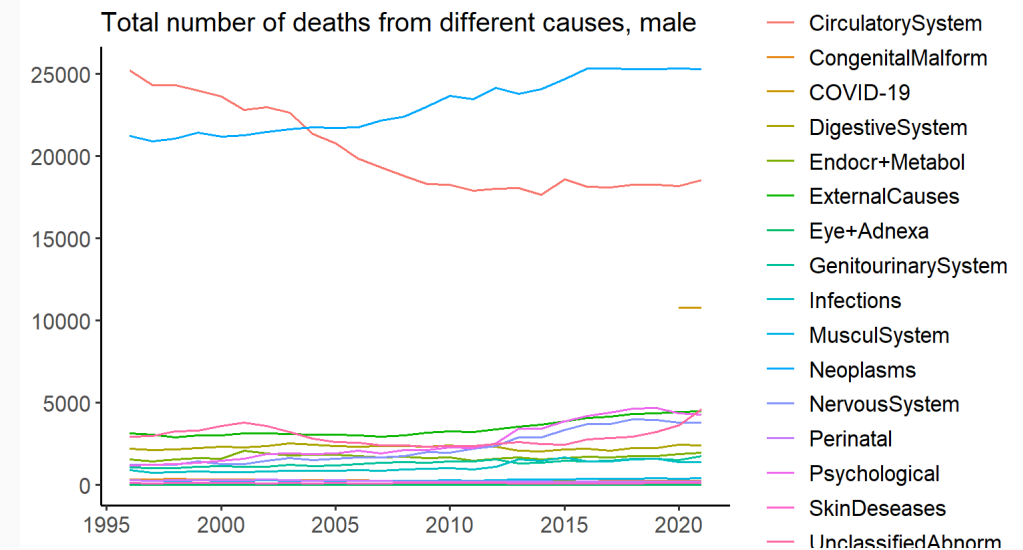
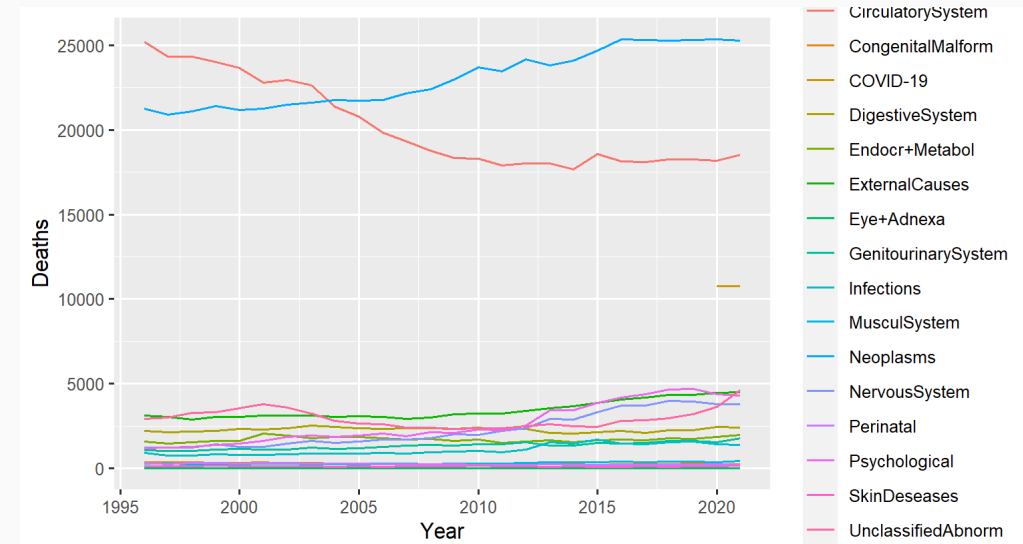


Customised theme: axes and legend text formatting

```
p <- Death_in_NL %>%
  filter(Age = "Total", Sex = "Male")%>%
  ggplot(aes(x = Year, y = Deaths, color = CausesOfDeat
  geom_path()+
  scale_color_discrete()

p1 <- p + theme_classic()+
  ggtitle("Total number of deaths from different causes
  labs(x = NULL, y = NULL)+
  theme(legend.title = element_blank(),
        axis.text = element_text(size = 11),
        legend.text = element_text(size = 11))

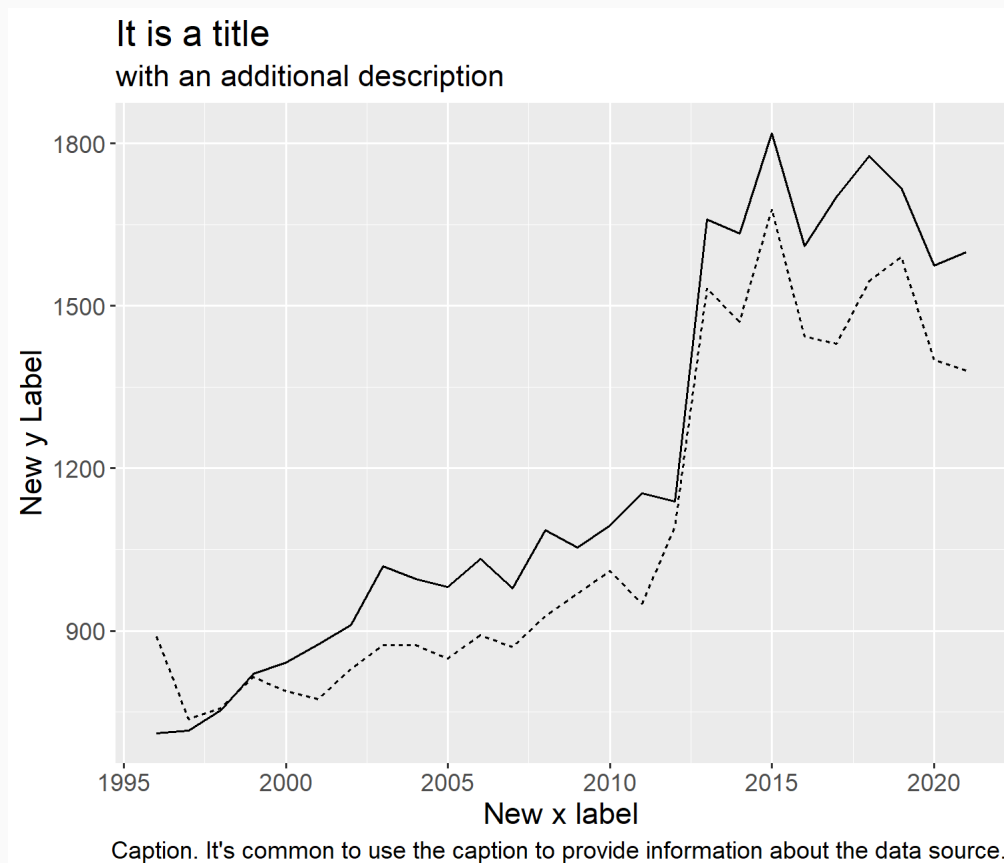
p
p1
```



labs: annotate the plot

Good labels are critical for making your plots accessible to a wider audience. Always ensure the axis and legend labels display the full variable name. Use the plot title and subtitle to explain the main findings.

```
Death_in_NL %>%  
  filter(Age = "Total", CausesOfDeath = "Infections")  
  ggplot(mapping = aes(x = Year, y = Deaths, linetype =  
    geom_path(show.legend = FALSE))+  
  labs(title = "It is a title",  
        subtitle = "with an additional description",  
        caption = "Caption. It's common to use the capti  
        x = "New x label",  
        y = "New y Label")+  
  theme(text = element_text(size = 15))
```



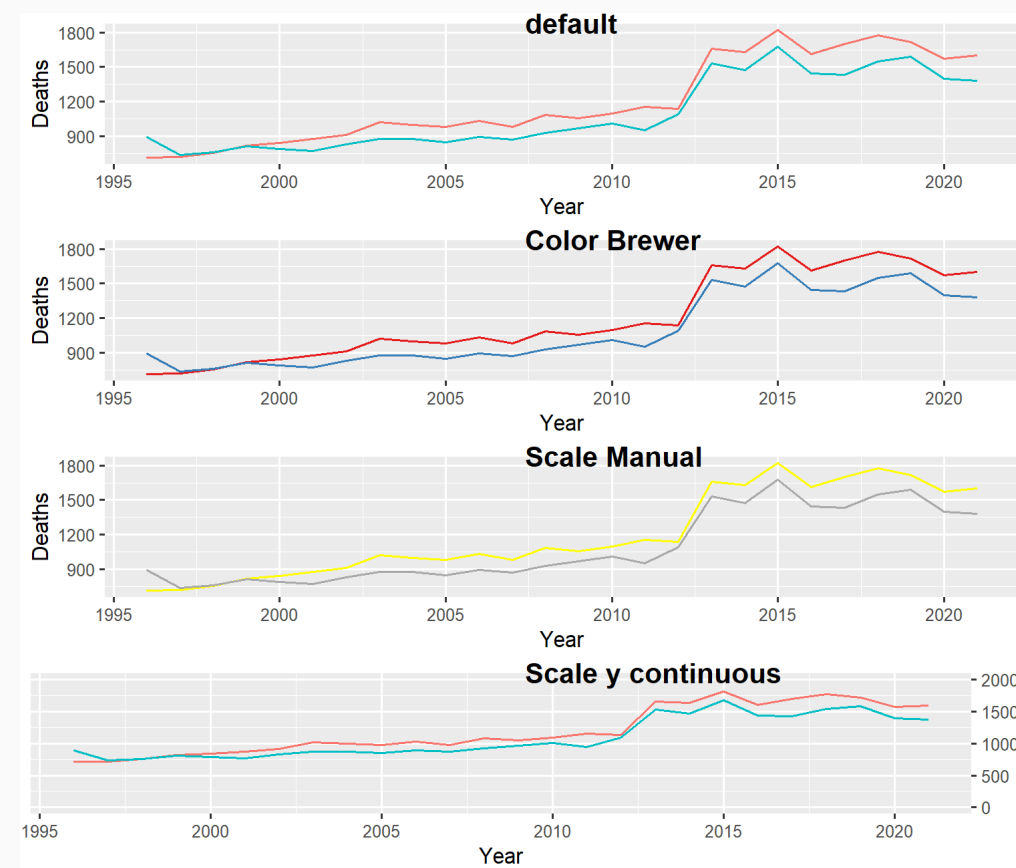
Scales formatting and transformations

`scale_` family of functions affect how a single observation raw data is 'interpreted' before going on the graph. There are two types of scales: `continuous` and `discrete`. `scale_` functions are used to set colors, limits, and transformations.

```
p <- Death_in_NL %>%  
  filter(Age == "Total", CausesOfDeath == "Infections")  
ggplot(mapping = aes(x = Year, y = Deaths, color = Sex)) +  
  geom_path(show.legend = FALSE)
```

```
p1 <- p + scale_color_brewer(palette = 'Set1')  
p2 <- p + scale_color_manual(values = c("yellow", "darkred"))  
p3 <- p + scale_y_continuous(position = "right", limits = c(0, 2000))
```

```
plot_grid(p, p1, p2, p3, ncol = 1, labels = c("default",  
  label_x = 0.5, label_y = 1, hjust = 0, vjust = 0))
```



Customised bar plot: example

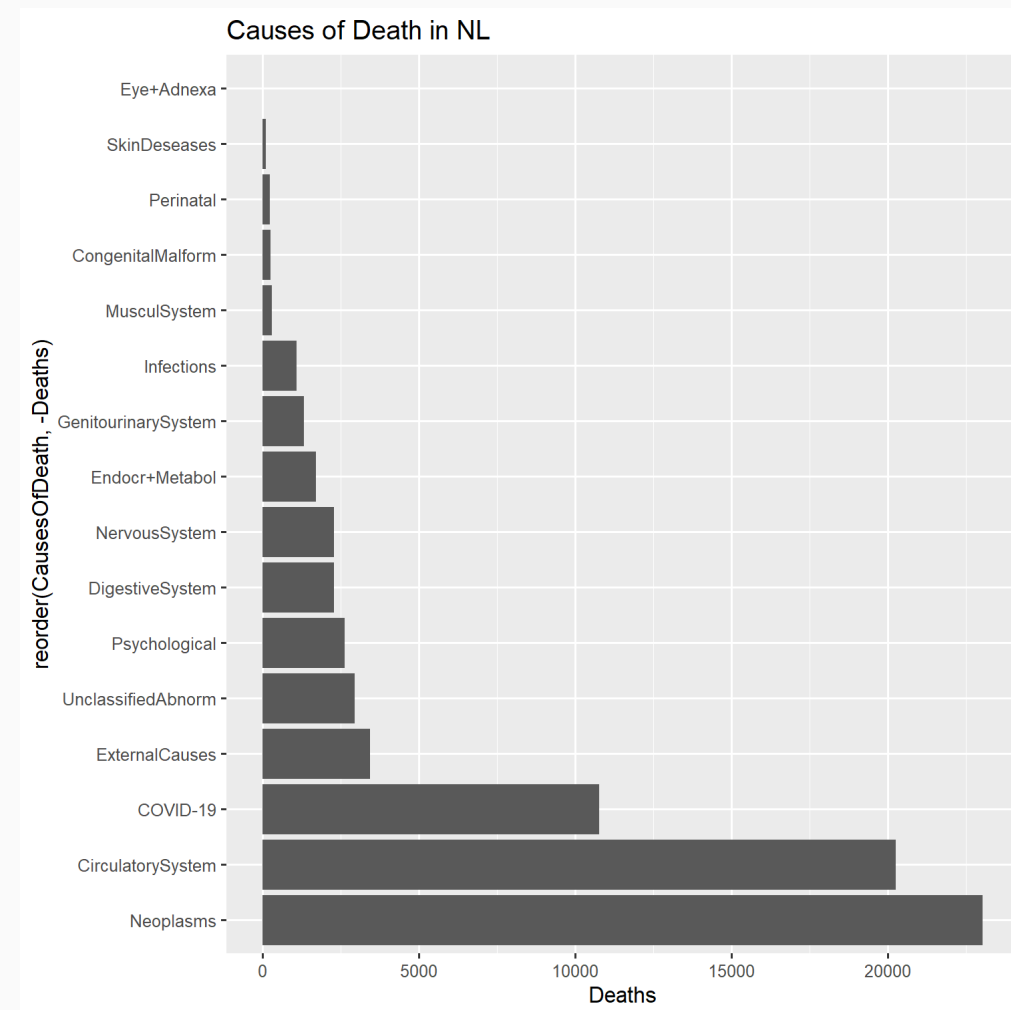
Lets customize the bar plot we built on Tutorial 2

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

What I don't like:

1. the y axis title is non-sense
2. the x axis title is unclear
3. the text size on the y axis is too small
4. the gray background is annoying
5. the columns color is not appealing

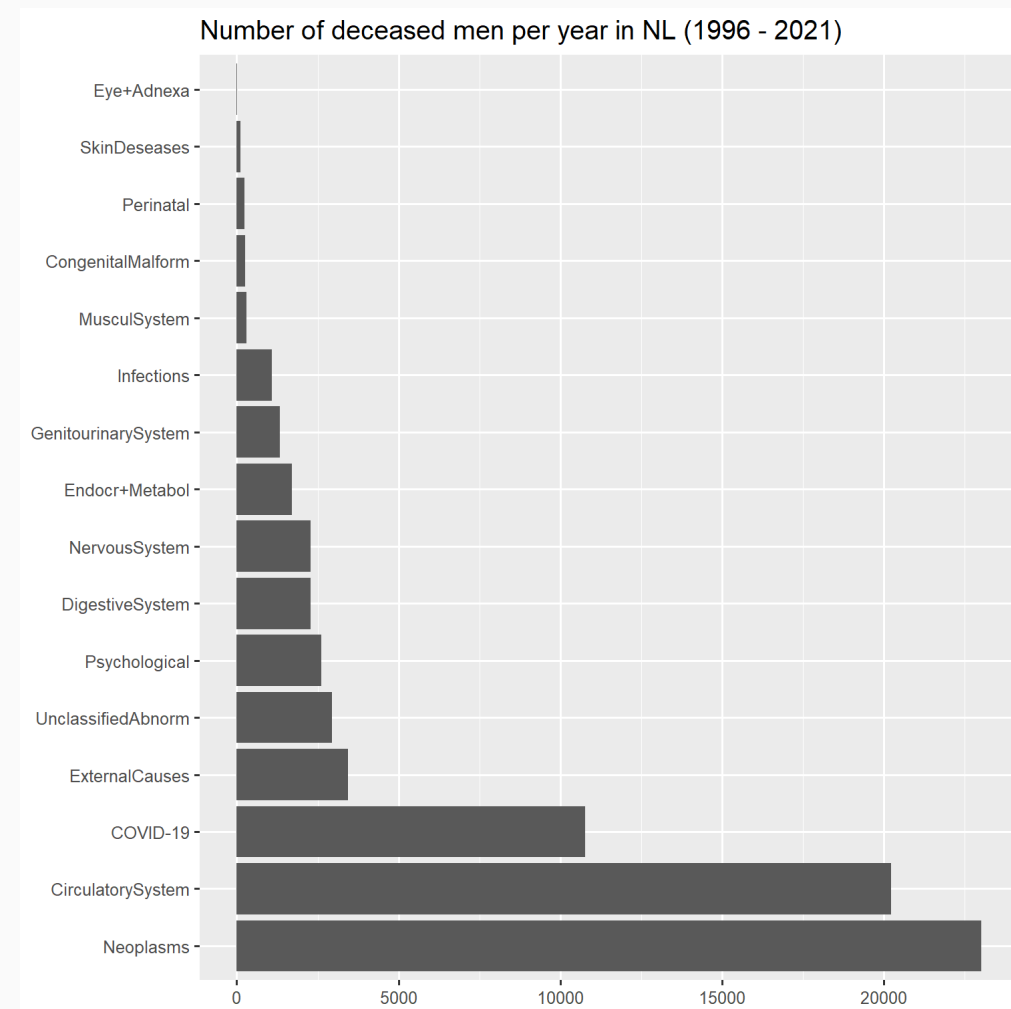
Lets address those issues!



Customised bar plot

1. the y axis title is non-sense
2. the x axis title is unclear
3. the text size on the y axis is too small
4. the gray background is annoying
5. the columns color is not appealing

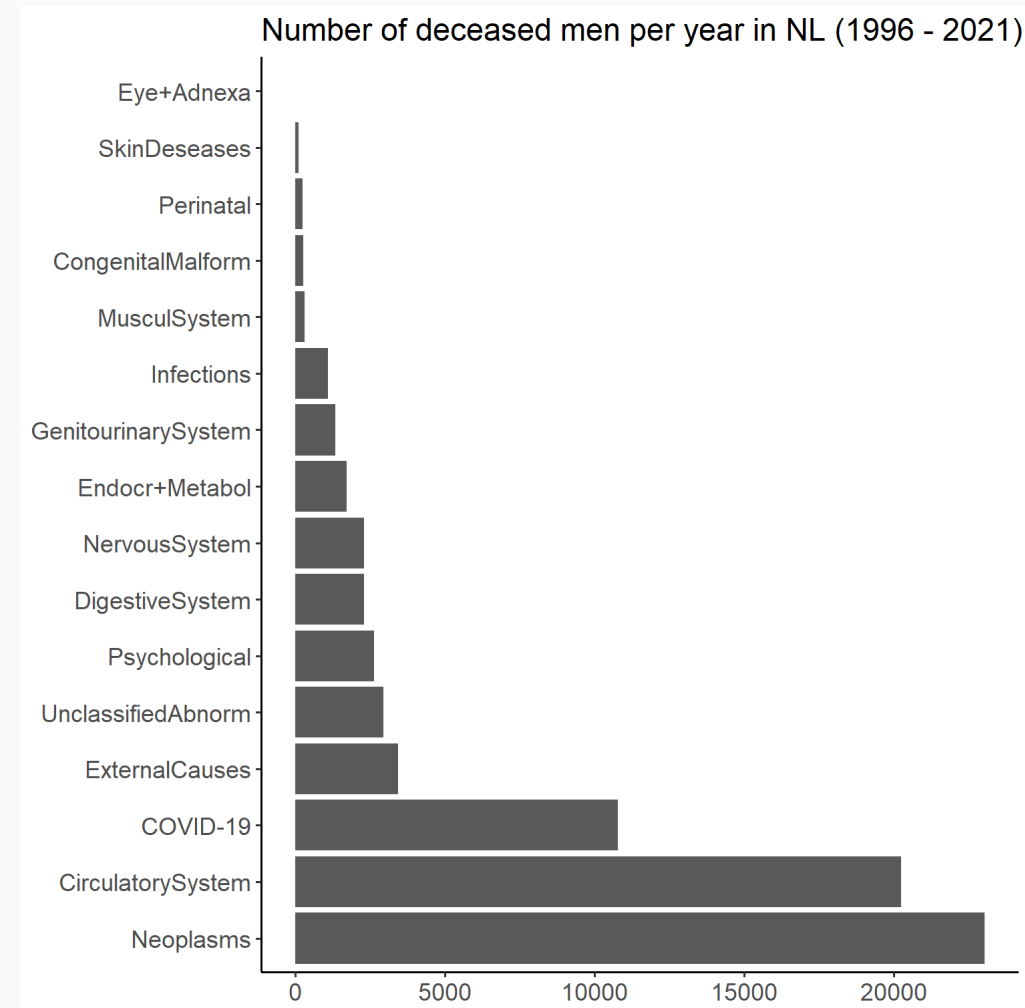
```
Death_in_NL%>%  
  filter(Age = "Total", Sex = "Male")%>%  
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [  
    stat_summary(fun.data = "mean_se", geom = "bar")+  
    coord_flip()+  
    labs(y = NULL, x = NULL)+  
    ggtitle("Number of deceased men per year in NL (1996
```



Customised bar plot

1. ~~the y axis title is non sense~~
2. ~~the x axis title is unclear~~
3. the text size on the y axis is too small
4. the gray background is annoying
5. the columns color is not appealing

```
Death_in_NL%>%
  filter(Age == "Total", Sex == "Male")%>%
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [
    stat_summary(fun.data = "mean_se", geom = "bar")+
    coord_flip()+
    labs(y = NULL, x = NULL)+
    theme_classic()+
    theme(axis.text = element_text(size = 12),
          title = element_text(size = 13))+
    ggtitle("Number of deceased men per year in NL (1996
```

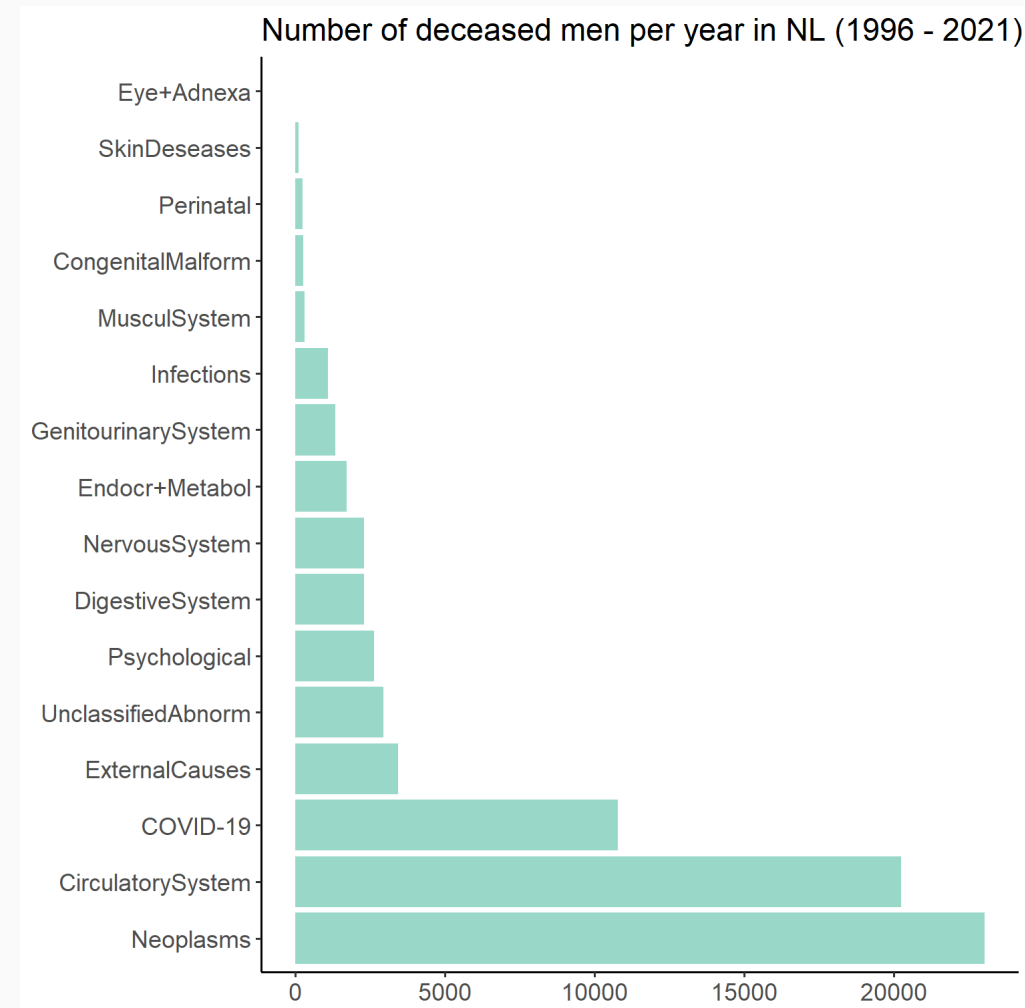


Customised bar plot

- ~~1. the y axis title is non sense~~
- ~~2. the x axis title is unclear~~
- ~~3. the text size on the y axis is too small~~
- ~~4. the gray background is annoying~~
5. the columns color is not appealing

```
Death_in_NL%>%
  filter(Age = "Total", Sex = "Male")%>%
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [
    stat_summary(fun.data = "mean_se", geom = "bar",
                  fill = "#99d8c9")+
    coord_flip()+
    labs(y = NULL, x = NULL)+
    theme_classic()+
    theme(axis.text = element_text(size = 12),
          title = element_text(size = 13))+
    ggtitle("Number of deceased men per year in NL (1996
```

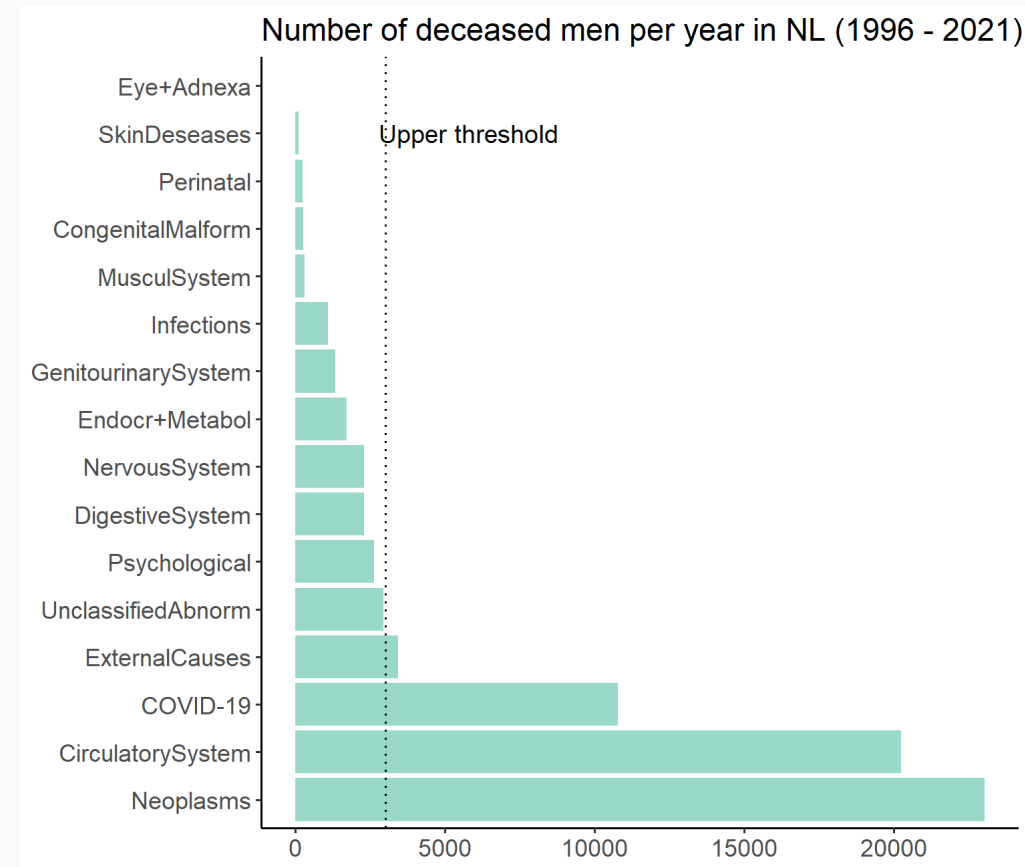
You can pick the color [here](#), or [here](#)



Customised bar plot: add annotation

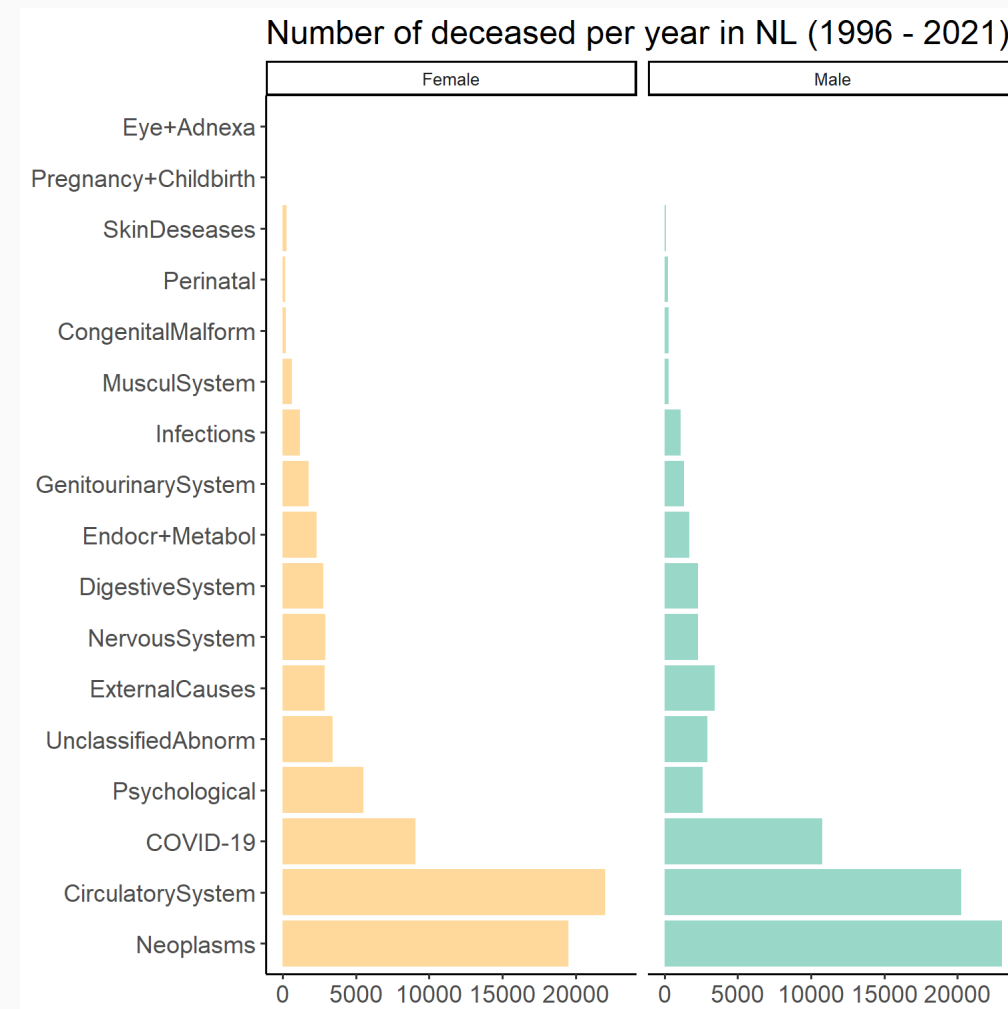
One can add additional elements and annotations

```
Death_in_NL%>%
  filter(Age = "Total", Sex = "Male")%>%
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [
    stat_summary(fun.data = "mean_se", geom = "bar", fill
    geom_hline(yintercept=3000, linetype="dotted")+
    annotate(geom="text", x=15, y=5800, label="Upper thre
  coord_flip()+
  labs(y = NULL, x = NULL)+
  theme_classic()+
  theme(axis.text = element_text(size = 12),
        title = element_text(size = 13))+
  ggtitle("Number of deceased men per year in NL (1996
```



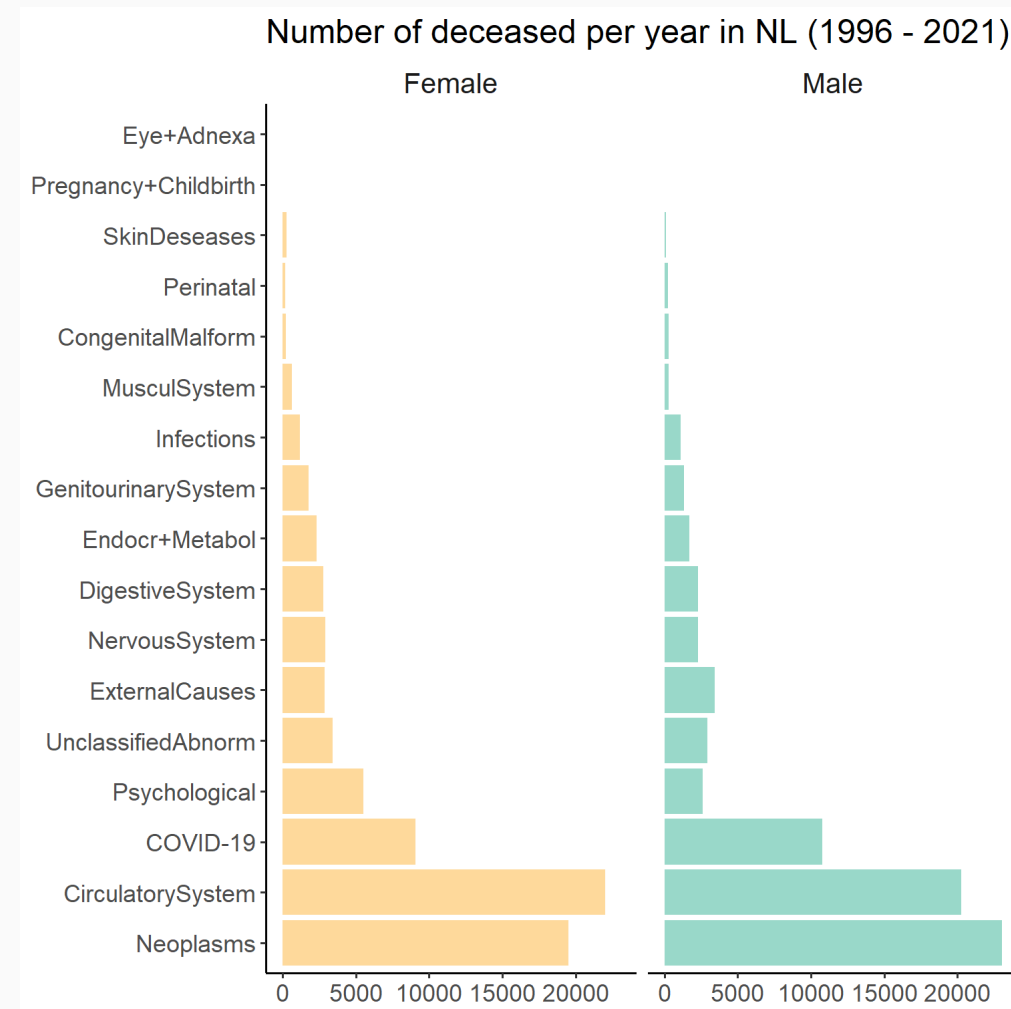
Customised bar plot: facets

```
Death_in_NL%>%  
  filter(Age == "Total")%>%  
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [  
    stat_summary(fun.data = "mean_se", geom = "bar", show  
    coord_flip()+  
    labs(y = NULL, x = NULL)+  
    scale_fill_manual(values = c("#fed99b", "#99d8c9"))+  
    theme_classic()+  
    theme(axis.text = element_text(size = 12),  
          title = element_text(size = 14))+  
    ggtitle("Number of deceased per year in NL (1996 - 2021)  
    facet_wrap(~Sex)
```



Customised bar plot: facets

```
Death_in_NL%>%
  filter(Age == "Total")%>%
  ggplot(aes(x = reorder(CausesOfDeath, -Deaths), y = [
    stat_summary(fun.data = "mean_se", geom = "bar", show
    coord_flip()+
    labs(y = NULL, x = NULL)+
    scale_fill_manual(values = c("#fed99b", "#99d8c9"))+
    theme_classic()+
    theme(axis.text = element_text(size = 12),
          title = element_text(size = 14),
          strip.background = element_blank(),
          strip.text = element_text(size = 14))+
    ggtitle("Number of deceased per year in NL (1996 - 2021)")
    facet_wrap(~Sex)
```

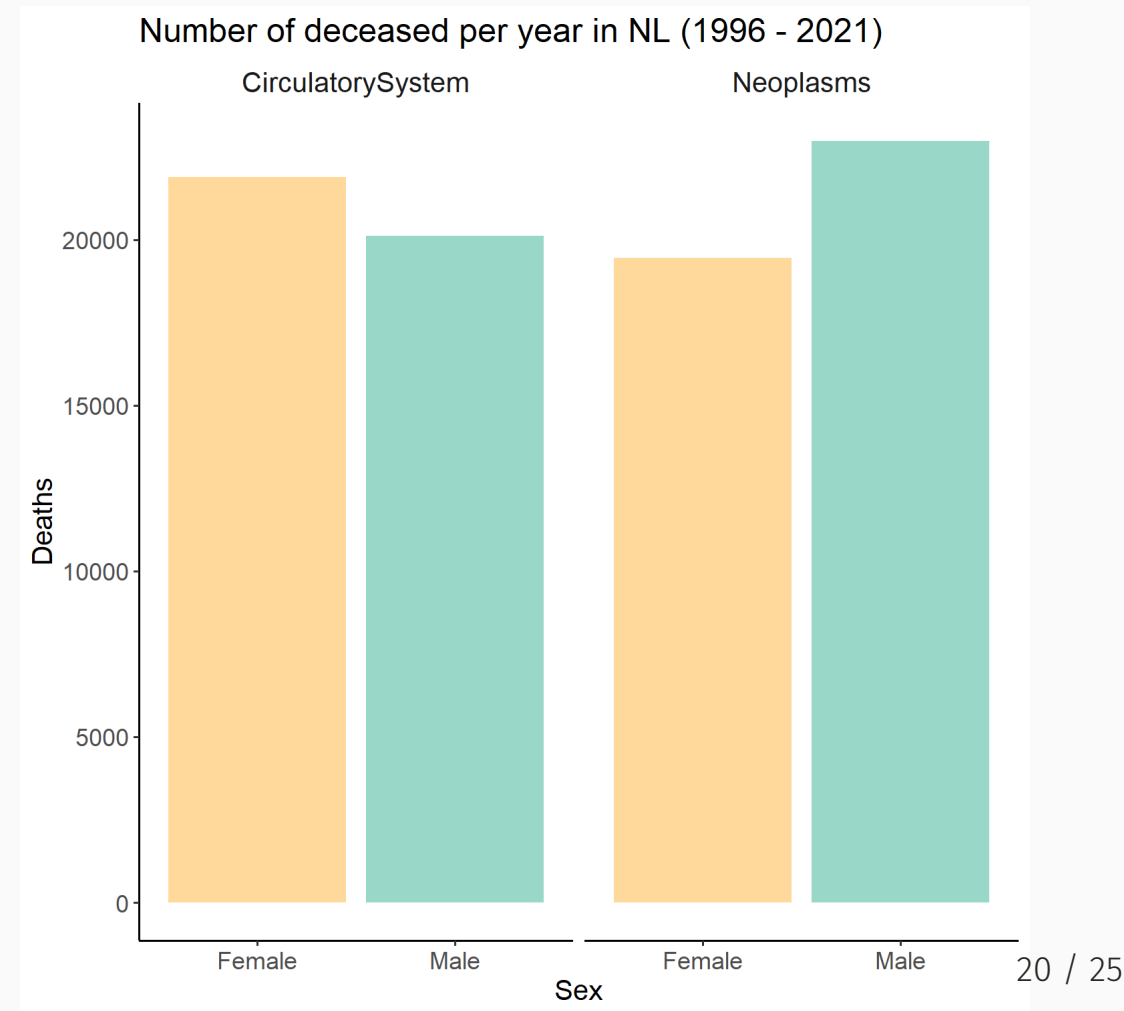


Customised strip chart

Vizualizations to explore the data

Lets zoom in to the top2 causes of death in NL and explore if the gender-related differences are significant or not

```
Death_in_NL%>%
  filter(Age == "Total", CausesOfDeath == c("Neoplasms",
  ggplot(aes(x = Sex, y = Deaths, fill = Sex))+
  stat_summary(fun.data = "mean_se", geom = "bar", show
  scale_fill_manual(values = c("#fed99b", "#99d8c9"))+
  theme_classic()+
  theme(axis.text = element_text(size = 12),
         title = element_text(size = 14),
         strip.background = element_blank(),
         strip.text = element_text(size = 14))+
  ggtitle("Number of deceased per year in NL (1996 - 20
  facet_wrap(~CausesOfDeath)
```

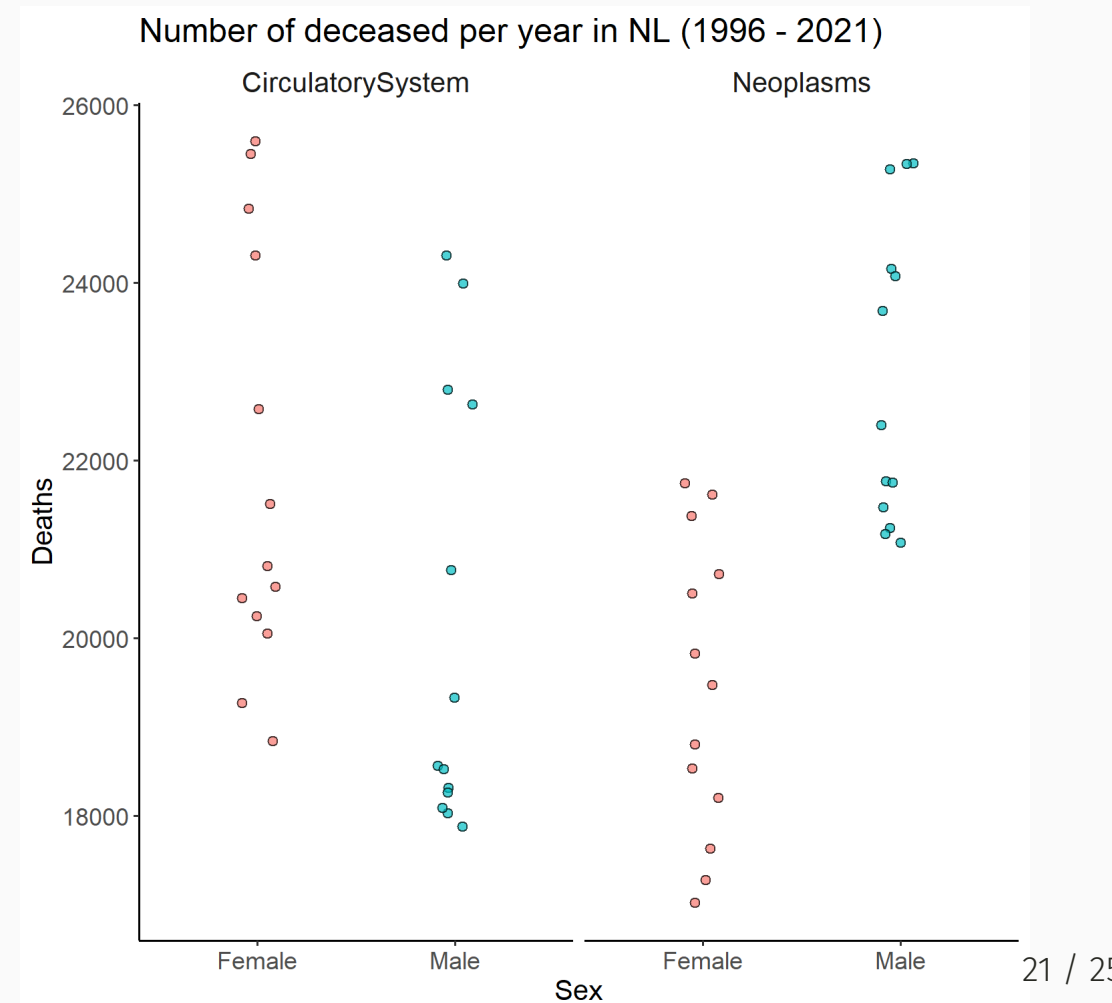


Showing individual observations on the plot

Lets plot individual observations instead of the bar plot using `geom_jitter`. This will requires some adjustments of the code.

```
Death_in_NL%>%
  filter(Age == "Total", CausesOfDeath == c("Neoplasms",
  ggplot(aes(x = Sex, y = Deaths, fill = Sex))+
  geom_jitter(show.legend = FALSE, alpha = 0.7, shape =
  scale_color_manual(values = c("#ef8a62", "#67a9cf"))+
  theme_classic()+
  theme(axis.text = element_text(size = 12),
        title = element_text(size = 14),
        strip.background = element_blank(),
        strip.text = element_text(size = 14))+
  ggtitle("Number of deceased per year in NL (1996 - 2021)")
  facet_wrap(~CausesOfDeath)
```

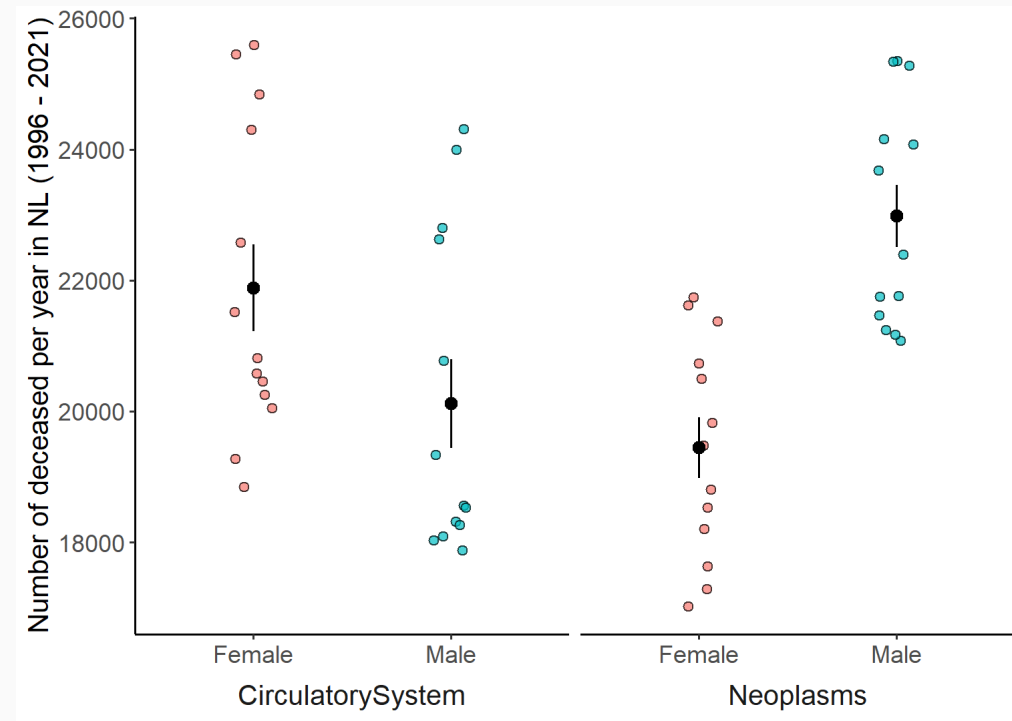
Are the differences significant?



Showing individual observations on the plot

Would be better having some statistical estimates on the top of the individual observations.

```
Death_in_NL%>%
  filter(Age == "Total", CausesOfDeath == c("Neoplasms",
  ggplot(aes(x = Sex, y = Deaths, fill = Sex))+
  geom_jitter(show.legend = FALSE, alpha = 0.7, shape =
  stat_summary(fun.data = "mean_se", show.legend = FALSE,
  scale_color_manual(values = c("#ef8a62", "#67a9cf"))+
  theme_classic()+
  labs(x = NULL, y = "Number of deceased per year in NL")
  theme(axis.text = element_text(size = 12),
        title = element_text(size = 14),
        strip.background = element_blank(),
        strip.text = element_text(size = 14),
        strip.placement = "outside")+
  facet_wrap(~CausesOfDeath, strip.position = "bottom")
```

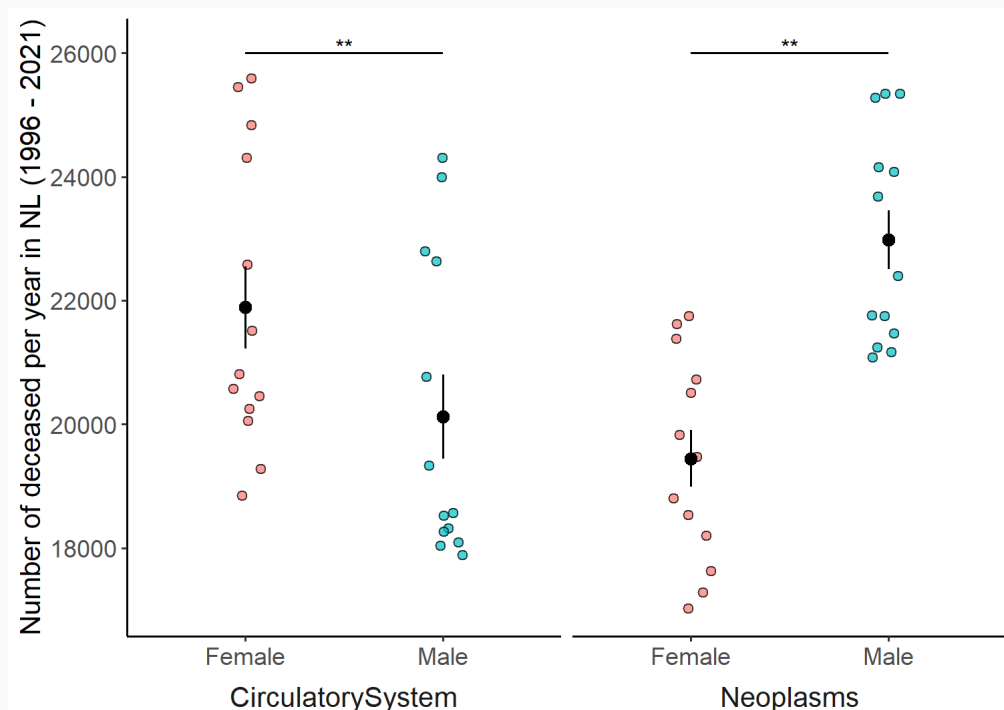


Are the differences significant?

```
##           Female
## Male 2.130499e-09
```


Showing significance levels

```
Death_in_NL%>%
  filter(Age = "Total", CausesOfDeath = c("Neoplasms",
  ggplot(aes(x = Sex, y = Deaths, fill = Sex))+
  geom_jitter(show.legend = FALSE, alpha = 0.7, shape =
  stat_summary(fun.data = "mean_se", show.legend = FALSE,
  scale_color_manual(values = c("#ef8a62", "#67a9cf"))+
  theme_classic()+
  labs(x = NULL, y = "Number of deceased per year in NL")
  theme(axis.text = element_text(size = 12),
        title = element_text(size = 14),
        strip.background = element_blank(),
        strip.text = element_text(size = 14),
        strip.placement = "outside")+
  facet_wrap(~CausesOfDeath, strip.position = "bottom")
  geom_line(data = tibble(x = c(1,2), y= c(26000, 26000)),
            aes(x=x, y = y), inherit.aes = FALSE)+
  geom_text(data = tibble(x = c(1.5), y= c(26100)),
            aes(x=x, y = y, label = "**"), inherit.aes = FALSE)
```



Vizualizing statistical data using ggpubr package

The `ggpubr` package provides some easy-to-use functions for creating and customizing 'ggplot2'- based publication ready plots.

```
library(ggpubr)
```

```
Death_in_NL%>%
```

```
  filter(Age = "Total", CausesOfDeath = c("Neoplasms"
```

```
  ggboxplot(x = "Sex", y = "Deaths",
```

```
           color = "Sex", palette = c("#00AFBB", "#F781BF",
```

```
           add = "jitter", shape = "Sex"))+
```

```
  facet_wrap(~CausesOfDeath)+
```

```
  stat_compare_means(label = "p.format", label.x = 1.5)
```

```
  labs(x = NULL, y = "Number of deceased men per year i
```

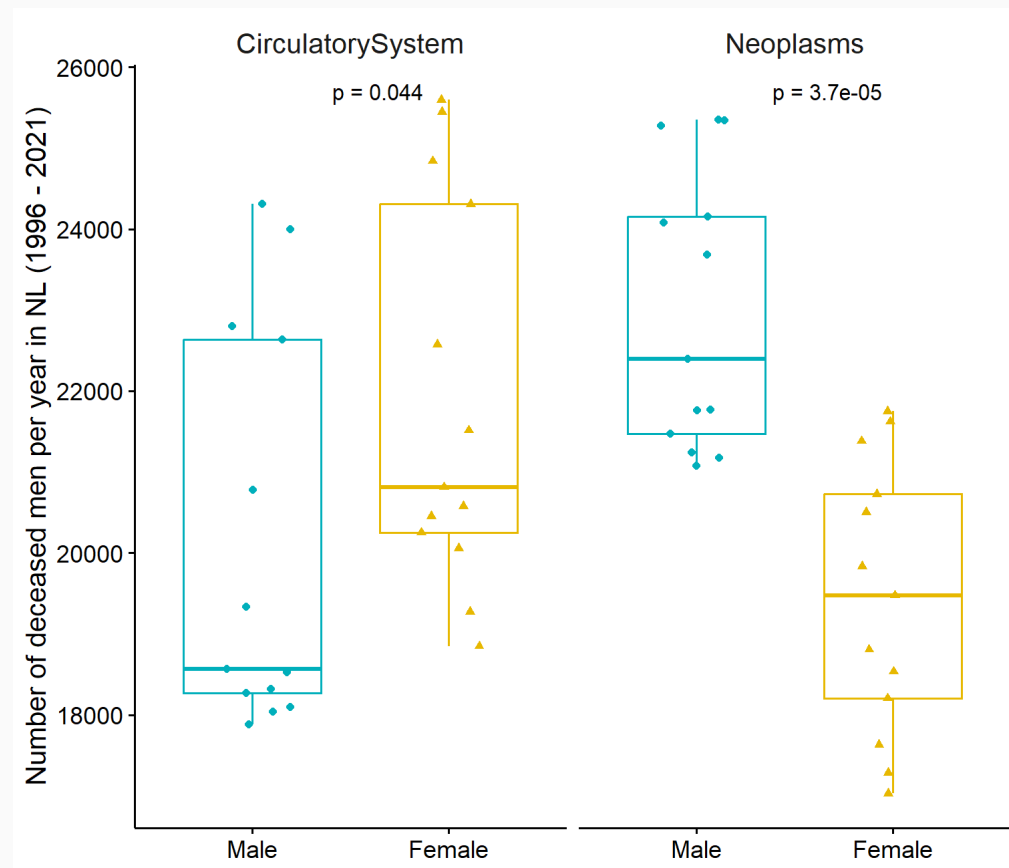
```
  theme(axis.text = element_text(size = 12),
```

```
        title = element_text(size = 14),
```

```
        strip.background = element_blank(),
```

```
        strip.text = element_text(size = 14))+
```

```
  rremove("legend")
```



Your turn

Part1.

Using Death_in_NL dataset build the following two vizualizations:

- What is an age profile of the number of deaths from external causes (not a disease) in NL? Is there any differences between man and women?
- How the number of perinatal deaths in NL changed over time?

You will need these data to answer the quiz in the brightspace.

Part 2.

- Adjust the style and design of the plot making it visually appealing.

You will need these skills to finalize your project next week.

