

# Laboratory with R

Enrique Akira Tagikuchi e Itziar Salazar

2025-11-29

## Exercise 1

### Download data (1a, 1b)

```
source('CreateDF.R')
if (!file.exists('sea_data.RData')){
  download_data('sea_data.RData')
}
load('sea_data.RData')
```

### Check dimensionality (1c)

```
cat("Dimension df_yearly\n")

## Dimension df_yearly
dim_sea.deep <- dim(sea.deep)
cat("Rows (obs.):", dim_sea.deep[1], "\n")

## Rows (obs.): 936
cat("Columns (var.):", dim_sea.deep[2], "\n")

## Columns (var.): 4
print(str(sea.deep)) #Structure of the df, dimensions, variables and data types

## 'data.frame': 936 obs. of 4 variables:
## $ mes : chr "Gener" "Gener" "Gener" "Gener" ...
## $ fondària : num 0 -20 -50 -80 0 -20 -50 -80 0 -20 ...
## $ any : int 2000 2000 2000 2000 2000 2000 2000 2000 2000 ...
## $ temperatura: num 12.9 12.9 12.8 12.7 12.7 12.7 12.6 12.4 12.8 12.7 ...
## NULL

print(summary(sea.deep)) #Descriptive statistical summary: mean, quantiles, min, max, NAs...

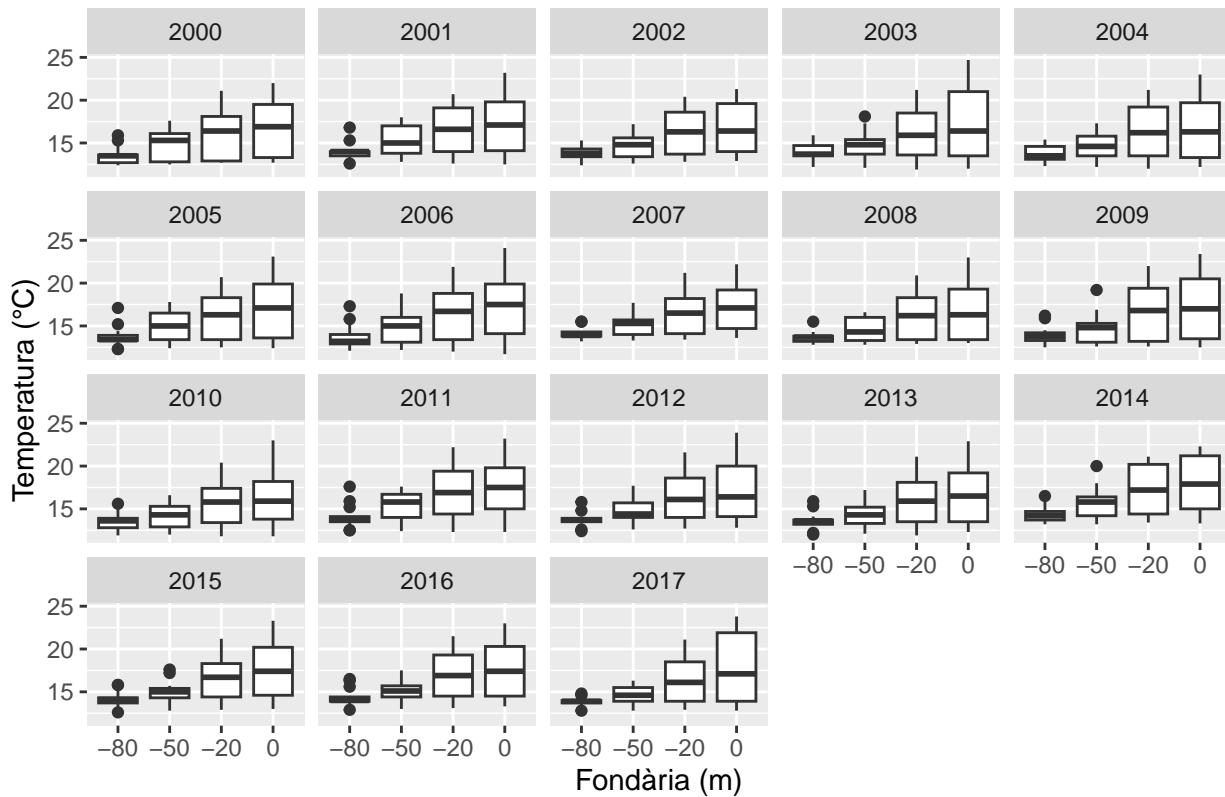
##      mes          fondària        any      temperatura
##  Length:936      Min.   :-80.0   Min.   :2000   Min.   :11.70
##  Class :character 1st Qu.:-57.5   1st Qu.:2004   1st Qu.:13.50
##  Mode  :character  Median :-35.0   Median :2008   Median :14.80
##                  Mean   :-37.5   Mean   :2008   Mean   :15.63
##                  3rd Qu.:-15.0   3rd Qu.:2013   3rd Qu.:17.10
##                  Max.    : 0.0    Max.   :2017   Max.   :24.70

print(sum(is.na(sea.deep))) #Just to make sure
```

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

#### Boxplot representation (1d)

Boxplot de la temperatura per fondària (2000–2017)



#### Statistical analysis (1e)

```
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

stats1 <- sea.deep %>%
  group_by(fondària, any) %>%
  summarise(
    Mitjana = mean(temperatura),
    Mediana = median(temperatura),
    SD = sd(temperatura),
    IQR = IQR(temperatura),
    Min = min(temperatura), #Other statistics of interest
    Max = max(temperatura),
```

```

Range = Max - Min, #Difference between max and min values
CV = SD / Mitjana, #Coefficient of Variation (CV)
  .groups = 'drop'
)

print(stats1)

## # A tibble: 72 x 10
##   fondària any Mitjana Mediana     SD    IQR    Min    Max Range     CV
##   <dbl> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     -80  2000  13.6  13.5  1.12  0.9   12.4  15.9  3.5  0.0823
## 2     -80  2001  14.0  14.   1.05  0.5   12.6  16.8  4.2  0.0752
## 3     -80  2002  13.8  13.8  0.783  0.9   12.4  15.3  2.9  0.0566
## 4     -80  2003  13.9  13.7  1.09  1.2   12.2  15.9  3.7  0.0785
## 5     -80  2004  13.7  13.5  1.03  1.5   12.3  15.4  3.1  0.0750
## 6     -80  2005  13.7  13.4  1.28  0.600 12.3  17.1  4.8  0.0930
## 7     -80  2006  13.8  13.2  1.53  1.1   12.1  17.3  5.2  0.111
## 8     -80  2007  14.1  13.9  0.716  0.5   13.2  15.5  2.3  0.0509
## 9     -80  2008  13.7  13.7  0.707  0.600 12.8  15.5  2.7  0.0518
## 10    -80  2009  13.9  13.8  1.12  0.900 12.5  16.2  3.7  0.0804
## # i 62 more rows

View(stats1) #Average temperature per depth per year

temp_dy <- sea.deep %>%
  group_by(fondària, any) %>% #Variables of interest
  summarise(
    temp_mitja = mean(temperatura),
    .groups = 'drop'
  ) #New variable average temperature per depth per year
print(temp_dy)

## # A tibble: 72 x 3
##   fondària any temp_mitja
##   <dbl> <int>     <dbl>
## 1     -80  2000     13.6
## 2     -80  2001     14.0
## 3     -80  2002     13.8
## 4     -80  2003     13.9
## 5     -80  2004     13.7
## 6     -80  2005     13.7
## 7     -80  2006     13.8
## 8     -80  2007     14.1
## 9     -80  2008     13.7
## 10    -80  2009     13.9
## # i 62 more rows

stats2 <- temp_dy %>%
  group_by(fondària) %>%
  summarise(
    Mitjana = mean(temp_mitja),
    Mediana = median(temp_mitja),
    SD = sd(temp_mitja),
    IQR = IQR(temp_mitja),
    Min = min(temp_mitja), #Other statistics of interest

```

```

Max = max(temp_mitja),
Range = Max - Min, #Difference between max and min values
CV = SD / Mitjana, #Coefficient of Variation (CV)
.groups = 'drop'
)

print(stats2)

## # A tibble: 4 x 9
##   fondària Mitjana Mediana   SD   IQR   Min   Max Range     CV
##       <dbl>    <dbl>    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1      -80    13.9    13.8 0.258 0.354 13.6 14.5  0.9  0.0186
## 2      -50    14.9    14.8 0.363 0.429 14.3 15.8  1.44 0.0243
## 3      -20    16.5    16.5 0.352 0.433 15.8 17.2  1.42 0.0213
## 4        0    17.2    17.1 0.374 0.575 16.5 17.9  1.35 0.0217
View(stats2) #Average temperature per depth across 2000-2017

```

### Annual variations (1f)

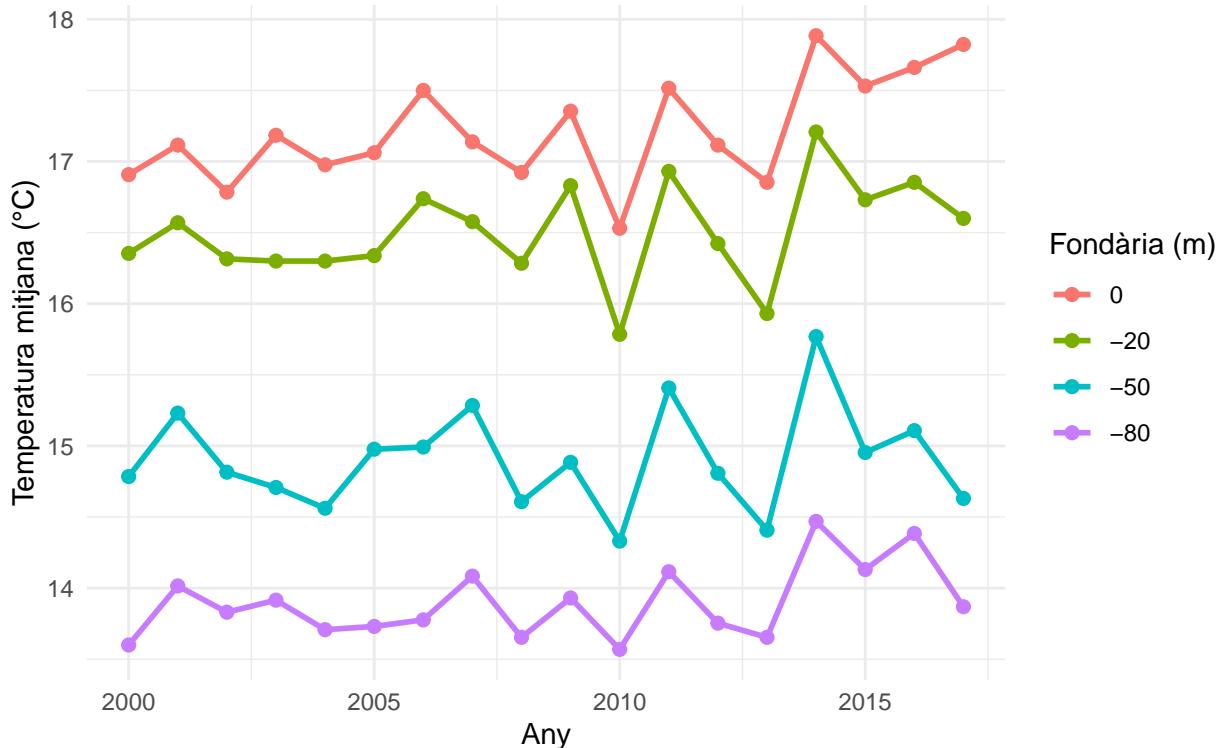
```

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

```

### Variació anual de la temperatura mitjana per fondària

(2000–2017) Mar Mediterrani, punt d'observació: 42° 03' N, 3° 15' E



## Save to Excel (1g)

```
library(openxlsx)

new_var <- createWorkbook()
#Average temperature per depth per year
addWorksheet(new_var, "Stats_per_any")
writeData(new_var, "Stats_per_any", stats1)
#Average temperature per depth across 2000-2017
addWorksheet(new_var, "Stats_globals")
writeData(new_var, "Stats_globals", stats2)
#Save on a new excel file
saveWorkbook(new_var, "NUEVO.xlsx", overwrite = TRUE)
```

## Exercise 2

### Plot1 - Heatmap

```
## Warning: package 'gplots' was built under R version 4.5.2
##
## -----
## gplots 3.3.0 loaded:
##   * Use citation('gplots') for citation info.
##   * Homepage: https://talgalili.github.io/gplots/
##   * Report issues: https://github.com/talgalili/gplots/issues
##   * Ask questions: https://stackoverflow.com/questions/tagged/gplots
##   * Suppress this message with: suppressPackageStartupMessages(library(gplots))
## -----

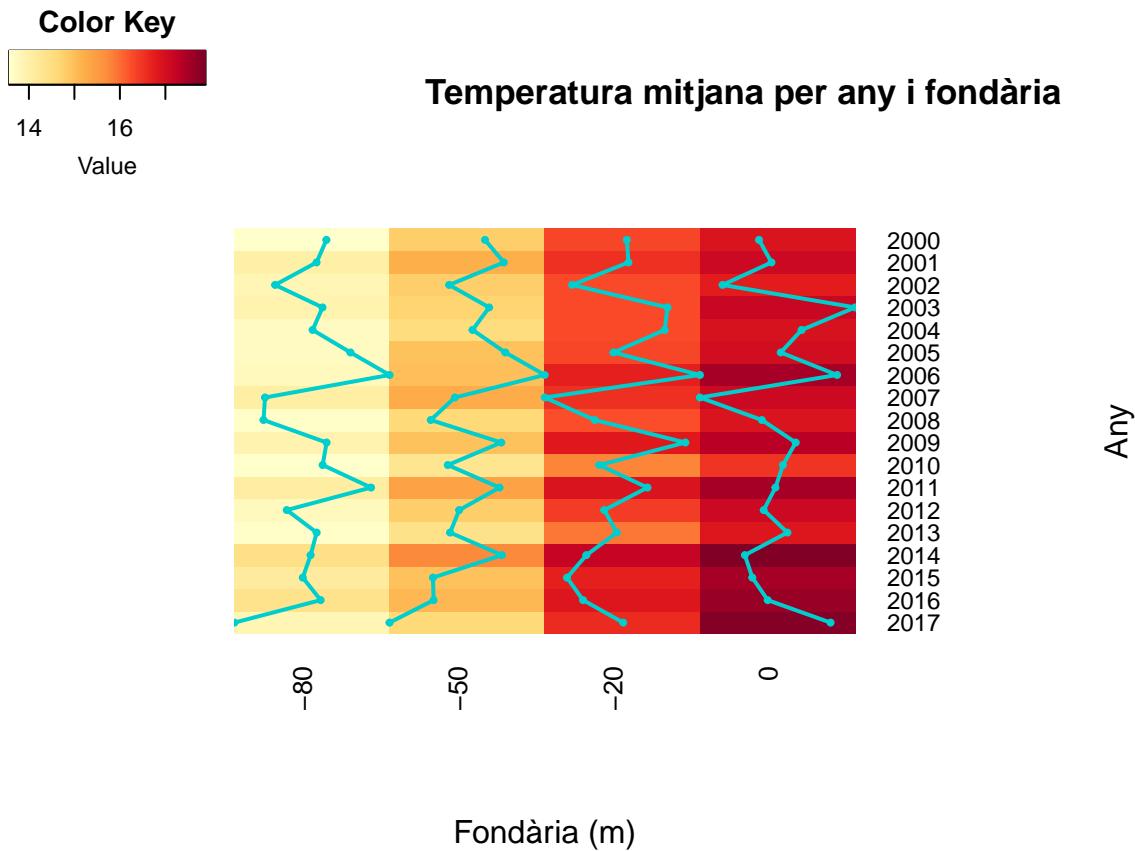
##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##     lowess

## `summarise()` has grouped output by 'any'. You can override using the `groups` argument.

## Warning: Setting row names on a tibble is deprecated.

## Warning: Setting row names on a tibble is deprecated.
```



##### The heatmap shows how the annual mean water temperature changes at different depths (-80, -50, -20 and 0 meters) between 2000 and 2017. The colour distribution shows that surface waters (0 and -20 meters) are consistently the warmest and exhibit the greatest interannual variability. This variability is evident in the blue lines representing the standard deviation (SD), which are imposed on the heatmap. The deeper layers (-50 and -80 meters) display a colder and more stable thermal profile, with significantly reduced standard deviation, indicating lower influence from seasonal cycles. The period between 2014 and 2016 stands out as particularly intense warming conditions are observed at the surface, potentially driven by large-scale oceanographic or climatic dynamics.

## Plot2 -

```
library(plotrix)

## Warning: package 'plotrix' was built under R version 4.5.2
##
## Attaching package: 'plotrix'

## The following object is masked from 'package:gplots':
##
##     plotCI

mes_1 <- c("Gener", "Febrer", "Març", "Abril", "Maig", "Juny",
          "Juliol", "Agost", "Setembre", "Octubre", "Novembre", "Desembre")

radial_data <- sea.deep %>%
  filter(mes != 'Mitjana anual') %>%
```

```

    mutate(mes = factor(mes, levels = mes_1)) %>%
    group_by(mes, fondària) %>%
    summarise(avg_temp = mean(temperatura, na.rm = TRUE), .groups = 'drop')

plot_matrix <- radial_data %>%
  pivot_wider(names_from = fondària, values_from = avg_temp) %>%
  arrange(mes) %>%
  select(-mes) %>%
  as.matrix()

plot_matrix = rbind(plot_matrix, plot_matrix[1, 1:4])

depth_levels <- colnames(plot_matrix)
num_depths <- length(depth_levels)

col_set = c("blue", "cyan", "green", "coral")

angles <- c((0:(length(mes_1) - 1)) * (2 * pi / length(mes_1)), 0)

radial.plot(
  t(plot_matrix), radial.pos = t(replicate(4, angles)), rp.type = "l",
  lty = 4, lwd = 2,
  labels = mes_1, label.pos = angles, line.col = col_set,
  radial.lim = c(0, 30),
  main = "Average Monthly Temperature by Depth (All Years)",
  clockwise = TRUE, start = pi/2
)
)

## Warning in x - lwidths * xpad: longer object length is not a multiple of
## shorter object length

## Warning in x + lwidths * xpad: longer object length is not a multiple of
## shorter object length

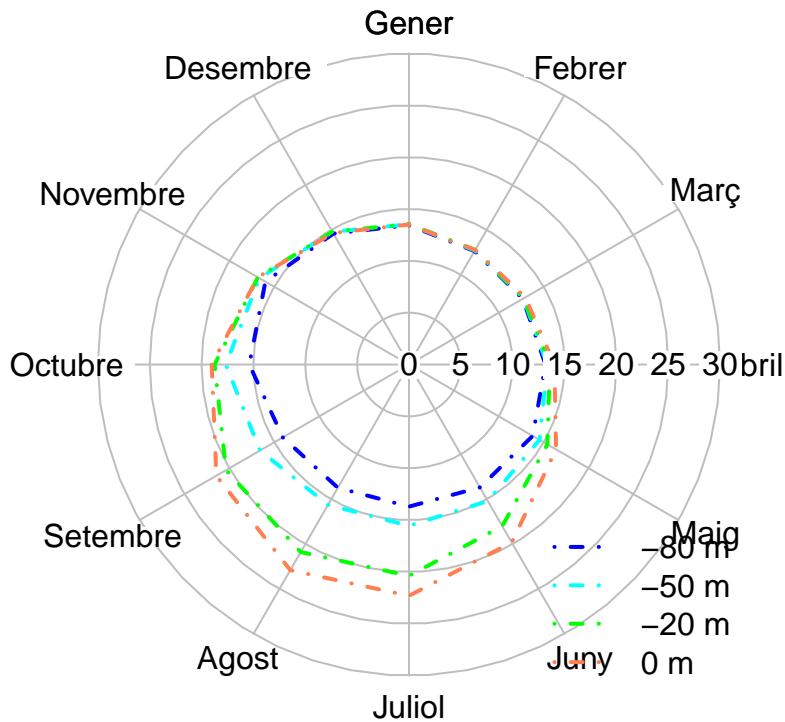
## Warning in y - bheights * ypad: longer object length is not a multiple of
## shorter object length

## Warning in y + theights * ypad: longer object length is not a multiple of
## shorter object length

legend(
  "bottomright", legend = paste(depth_levels, "m"),
  col=col_set, lty = 4, lwd = 2, bty = "n"
)

```

## Average Monthly Temperature by Depth (All Years)



### Exercise 3

```
library(stringr)
source('Ex3.R')
plots = ex3(sea.deep, sea.pred)

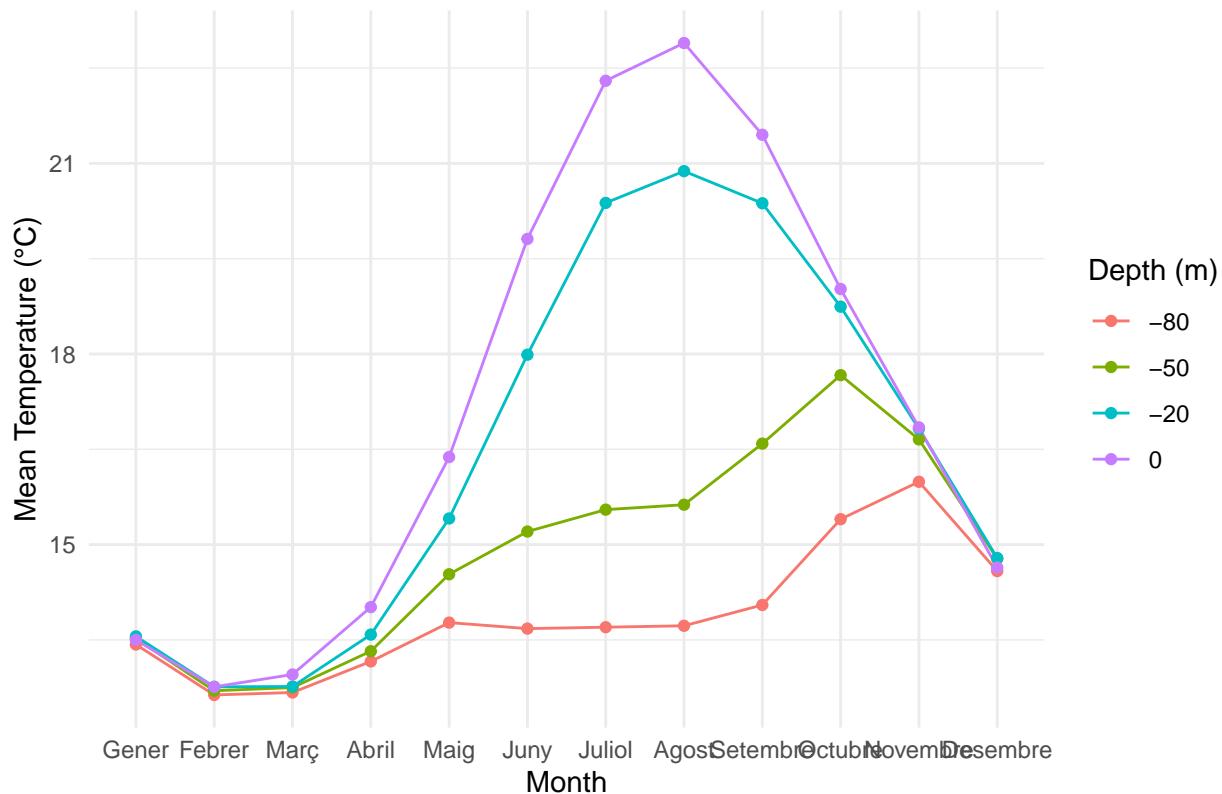
## Warning: Returning more (or less) than 1 row per `summarise()` group was deprecated in
## dplyr 1.1.0.
## i Please use `reframe()` instead.
## i When switching from `summarise()` to `reframe()`, remember that `reframe()`-
##   always returns an ungrouped data frame and adjust accordingly.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

## `summarise()` has grouped output by 'any', 'fondària'. You can override using
## the ` `.groups` argument.

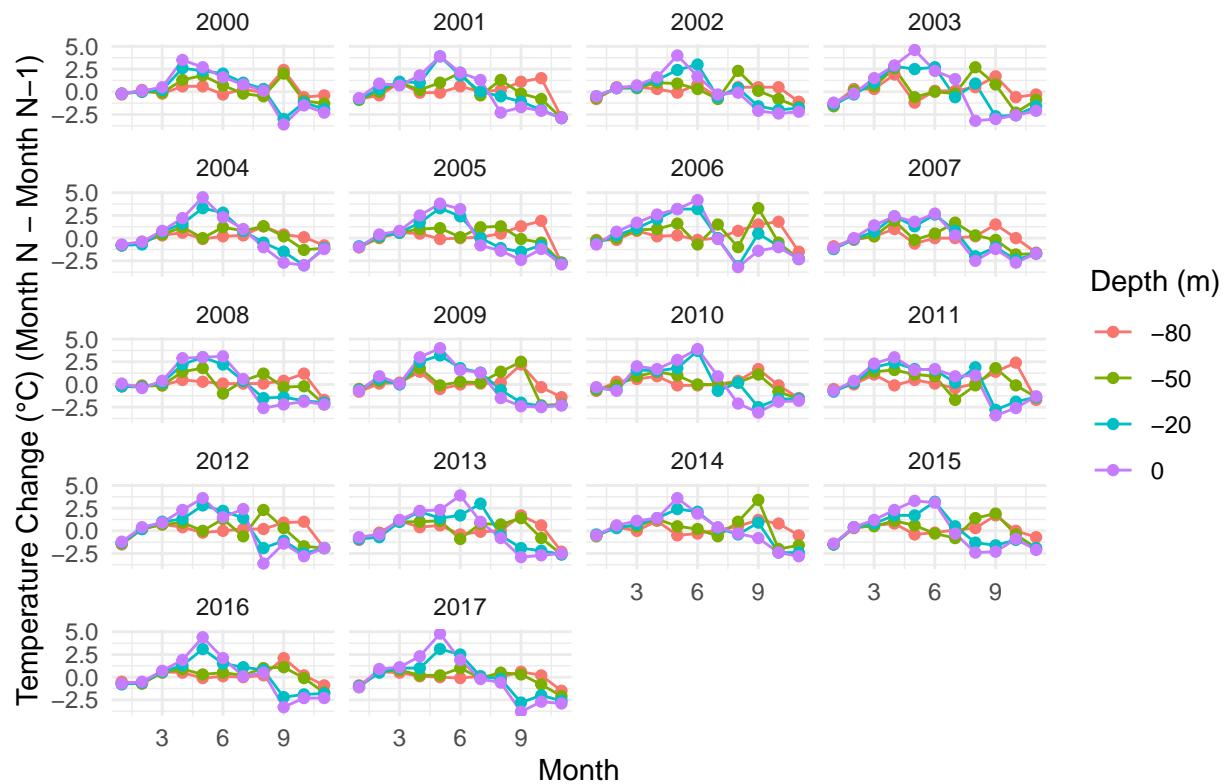
## Warning: Returning more (or less) than 1 row per `summarise()` group was deprecated in
## dplyr 1.1.0.
## i Please use `reframe()` instead.
## i When switching from `summarise()` to `reframe()`, remember that `reframe()`-
##   always returns an ungrouped data frame and adjust accordingly.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

## `summarise()` has grouped output by 'any', 'fondària'. You can override using
## the ` `.groups` argument.
```

## Average monthly changes

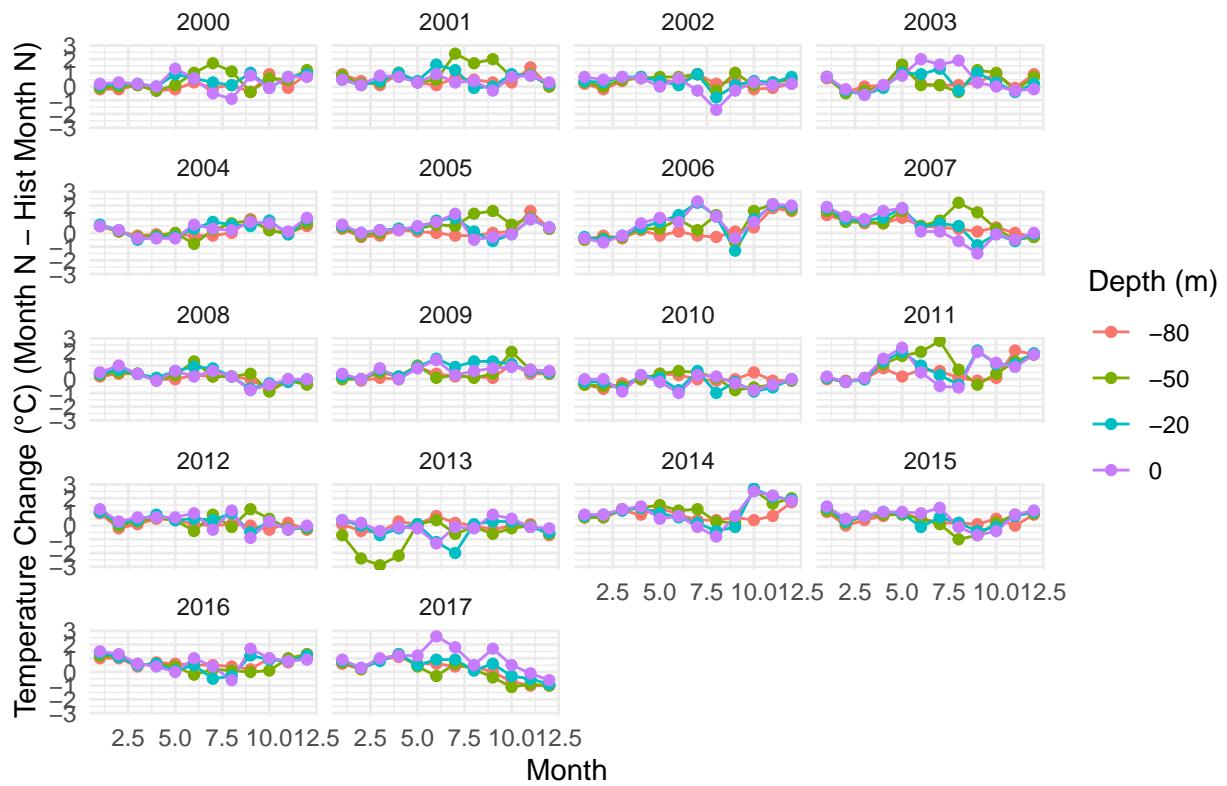


## Month-to-Month Temperature Change



```
print(plots$b1)
```

## Current year to previous 30 years



```
print(plots$b2)
```

## Average monthly changes

