Stream temperatures

Charlotte Wickham

```
library(tidyverse)
library(lubridate)
# knitr::opts_chunk$set(fig.height = 3, fig.width = 6)
```

Explore stream temperatures for all sites.

This is an example of where we might end up

Functions

```
# Calculate useful data coverage summaries
coverage_summary <- function(data){
  data %>%
  summarise(
    n_obs = n(),
    n_days = n_distinct(date),
    n_missing = sum(is.na(watertemp_mean_day))
  )
}
```

```
# Plot watertemp_day time series
# Note this is the two argument version
plot_temperature <- function(data, title){
  data %>%
    ggplot(aes(date, watertemp_mean_day)) +
    geom_line(aes(color = watertemp_method)) +
    scale_color_brewer("Method", palette = "Set2") +
    labs(x = "", y = "Water temperature (C)",
      title = title)
}
```

Data Import

```
streams <- read_csv("data/HT00441_v8.csv")
names(streams) <- tolower(names(streams))
```

Filter to time period of interest, and set observations with anything other than "Accepted" quality to missing:

```
streams <- streams %>%
  filter(year(date) > 2000, year(date) < 2020) %>%
  mutate(
    watertemp_mean_day = ifelse(watertemp_mean_flag != "A",
      NA, watertemp_mean_day)) %>%
  select(sitecode, date, watertemp_mean_day, watertemp_method)
```

Data coverage

`coverage_summary()` will work on grouped data:

```
streams %>%
  group_by(sitecode) %>%
  coverage_summary()
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

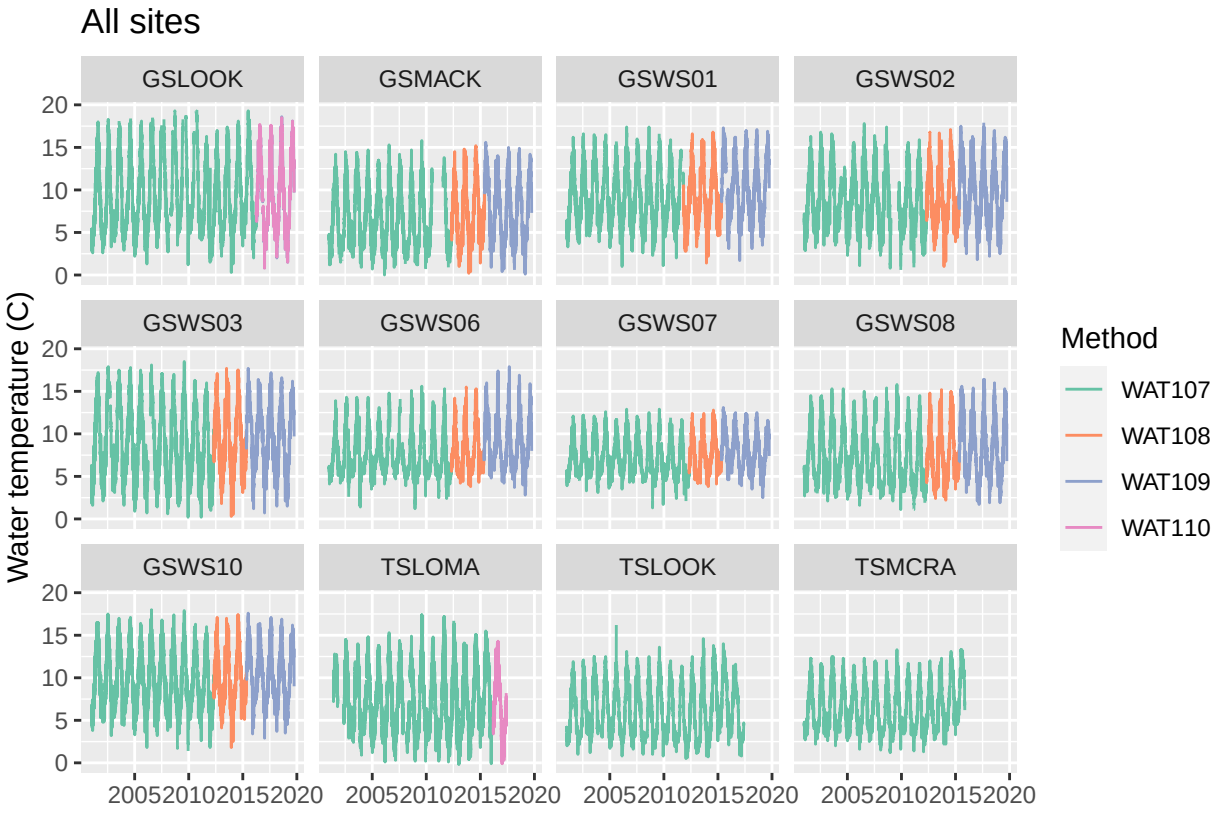
```
## # A tibble: 12 x 4
##    sitecode n_obs n_days n_missing
##    <chr>    <int>  <int>     <int>
##  1 GSLOOK    8120   6841       543
##  2 GSMACK    6847   6847       428
##  3 GSWS01    6847   6847       223
##  4 GSWS02    6847   6847       566
##  5 GSWS03    6847   6847       179
##  6 GSWS06    6847   6847       432
##  7 GSWS07    6847   6847        73
##  8 GSWS08    6847   6847       236
##  9 GSWS10    6847   6847        85
## 10 TSLOMA   11976   5988       938
## 11 TSLOOK    5989   5989       115
## 12 TSMCRA    5427   5427         0
```

Plots

One option is to apply the plot to all the data, then apply facetting to the resulting plot:

```
streams %>%
  plot_temperature("All sites") +
  facet_wrap(~ sitecode)
```

```
## Warning: Removed 10 row(s) containing missing values (geom_path).
```



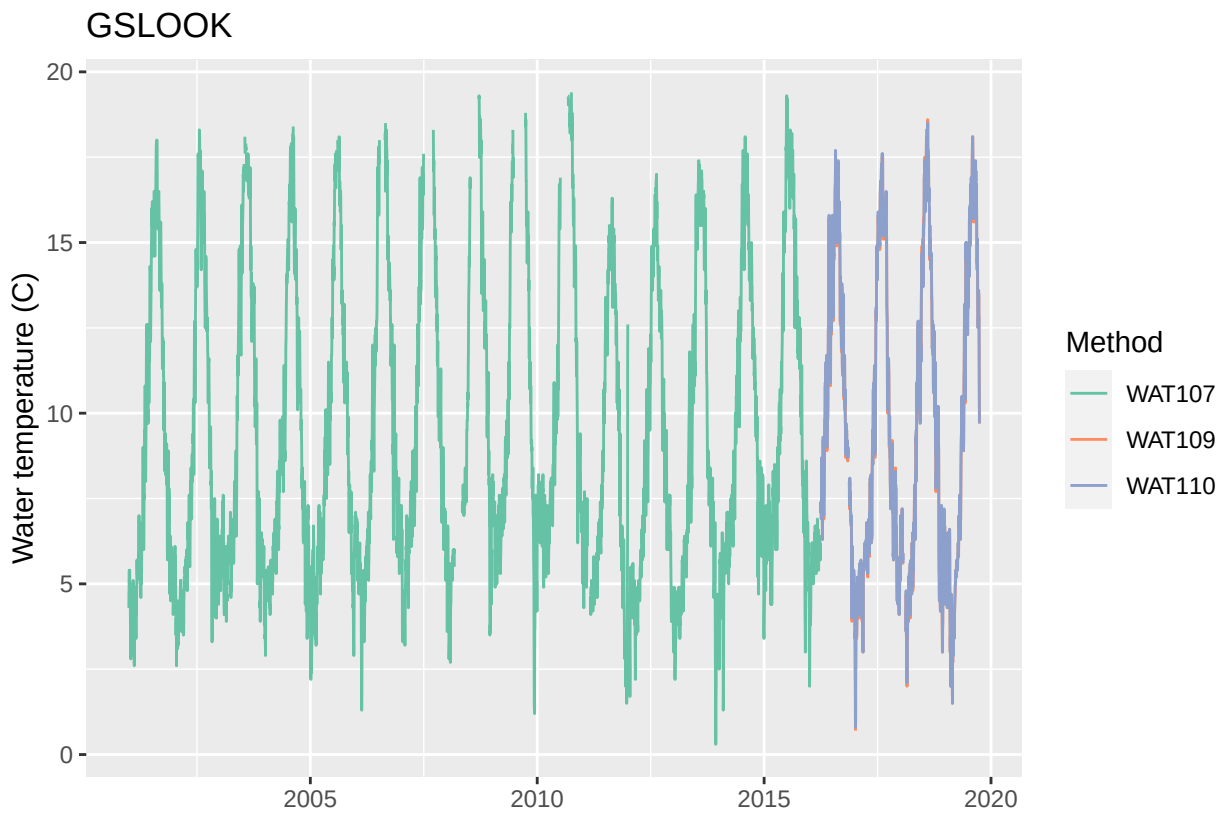
Alternatively, create and store all the plots with `rowwise()`:

```
streams_with_plots <- streams %>%
  nest_by(sitecode) %>%
  rowwise() %>%
  mutate(plots = list(plot_temperature(data, sitecode)))
```

Display one plot:

```
streams_with_plots$plots[[1]]
```

```
## Warning: Removed 5 row(s) containing missing values (geom_path).
```



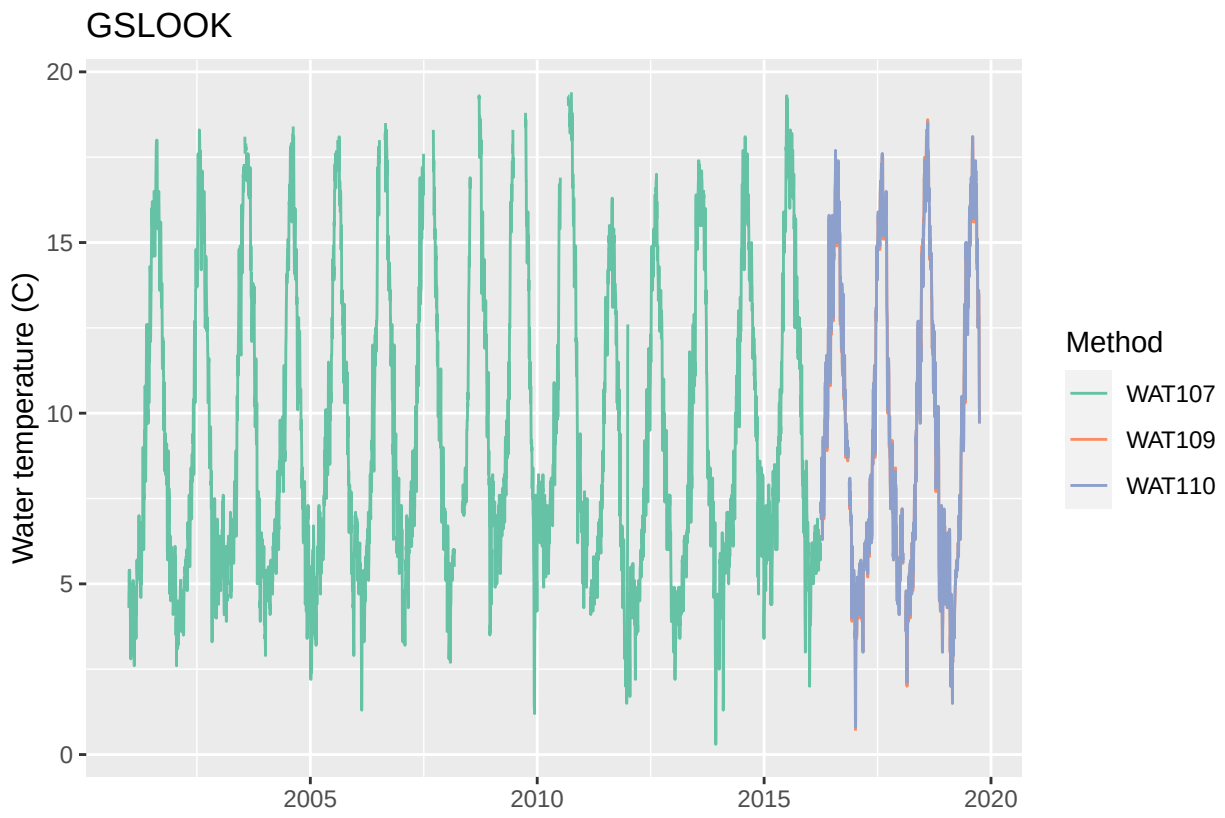
Different ways to actually see all the plots

(Also a survey of other iteration methods)

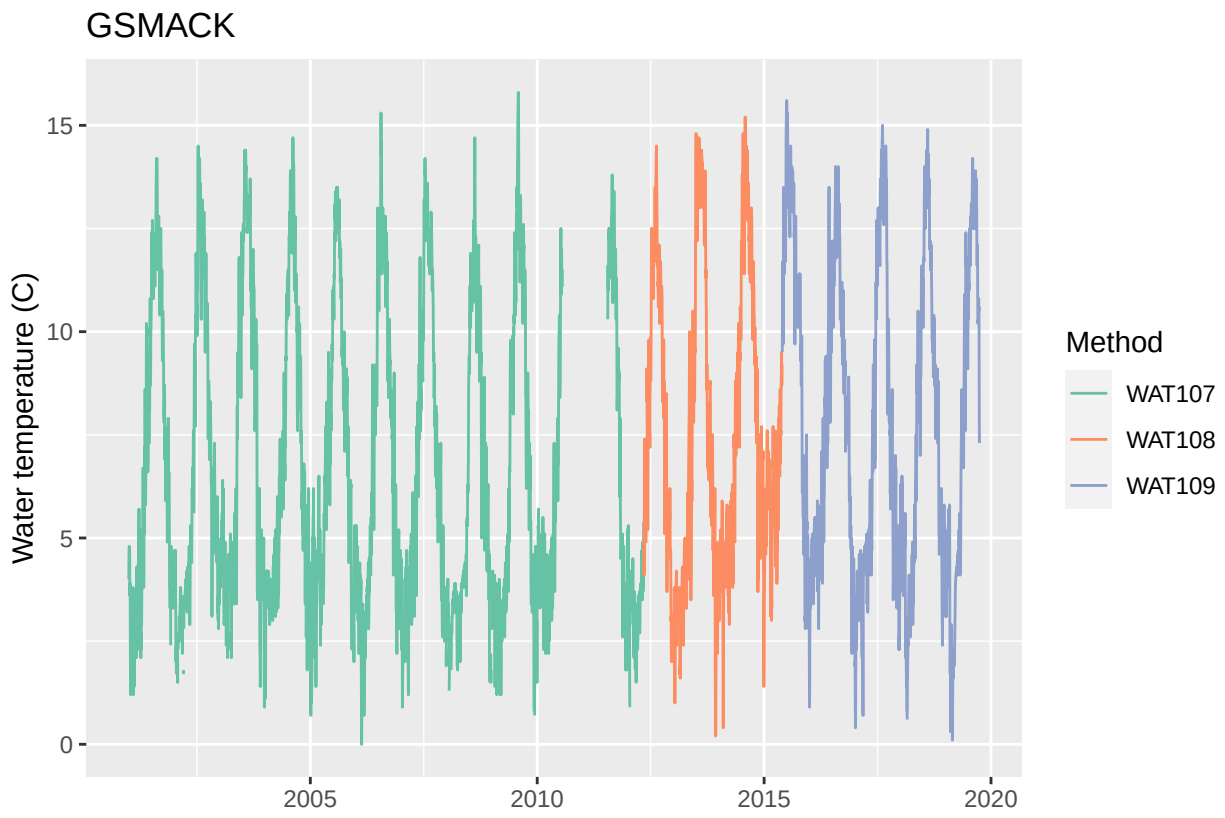
purrrs `walk()` function:

```
streams_with_plots %>% pull(plots) %>% walk(print)
```

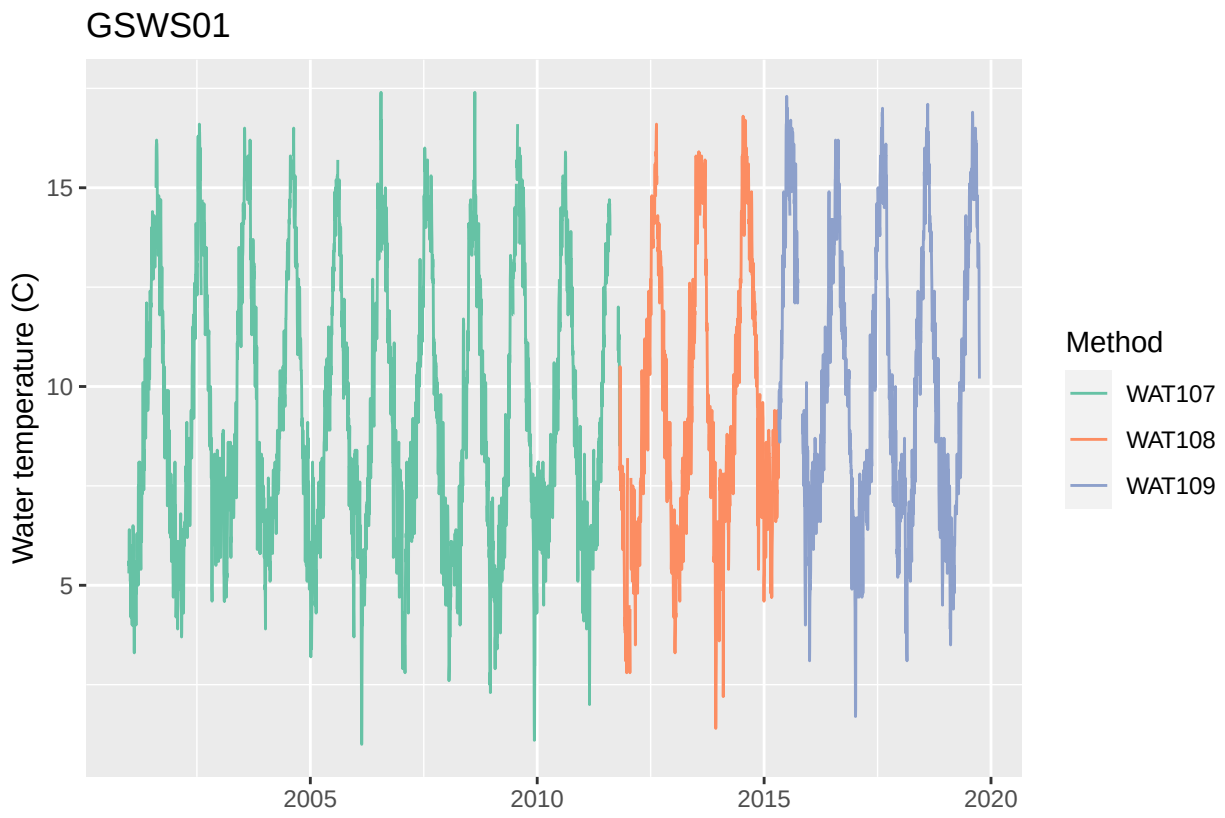
```
## Warning: Removed 5 row(s) containing missing values (geom_path).
```



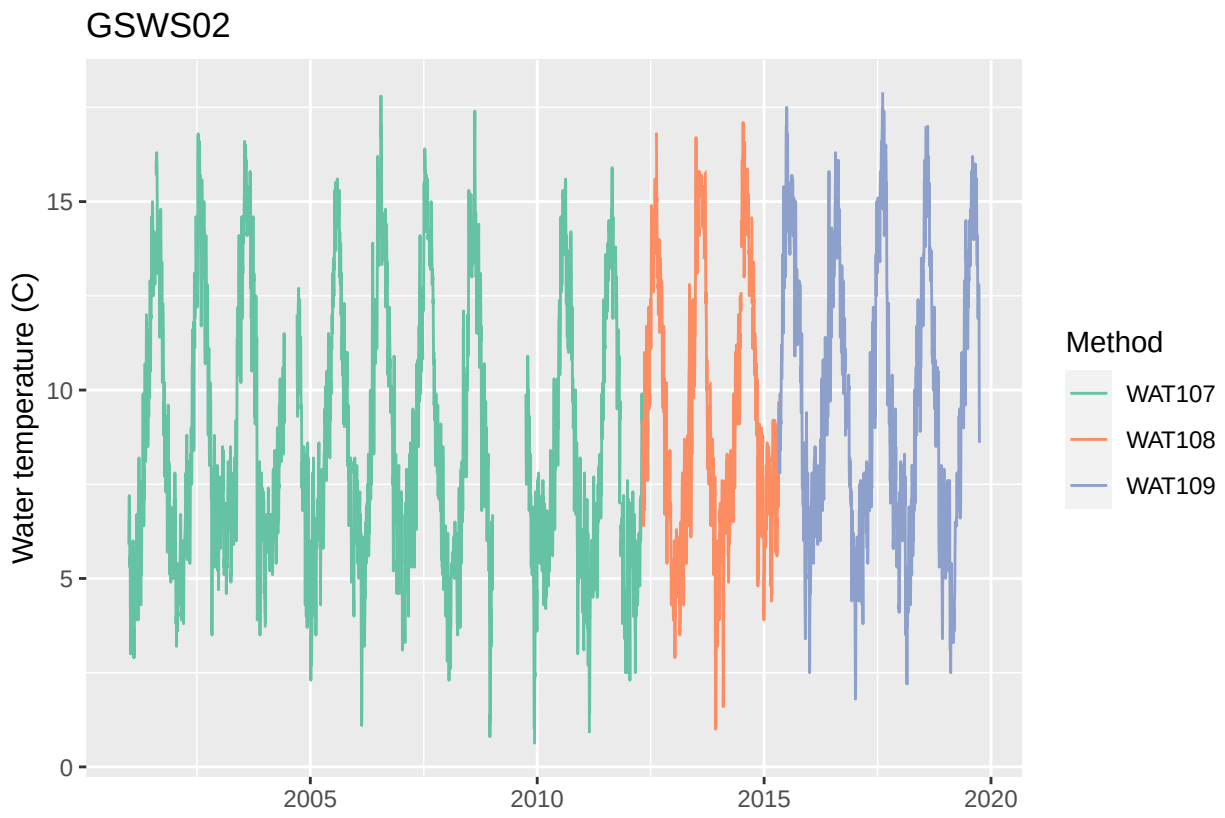

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```



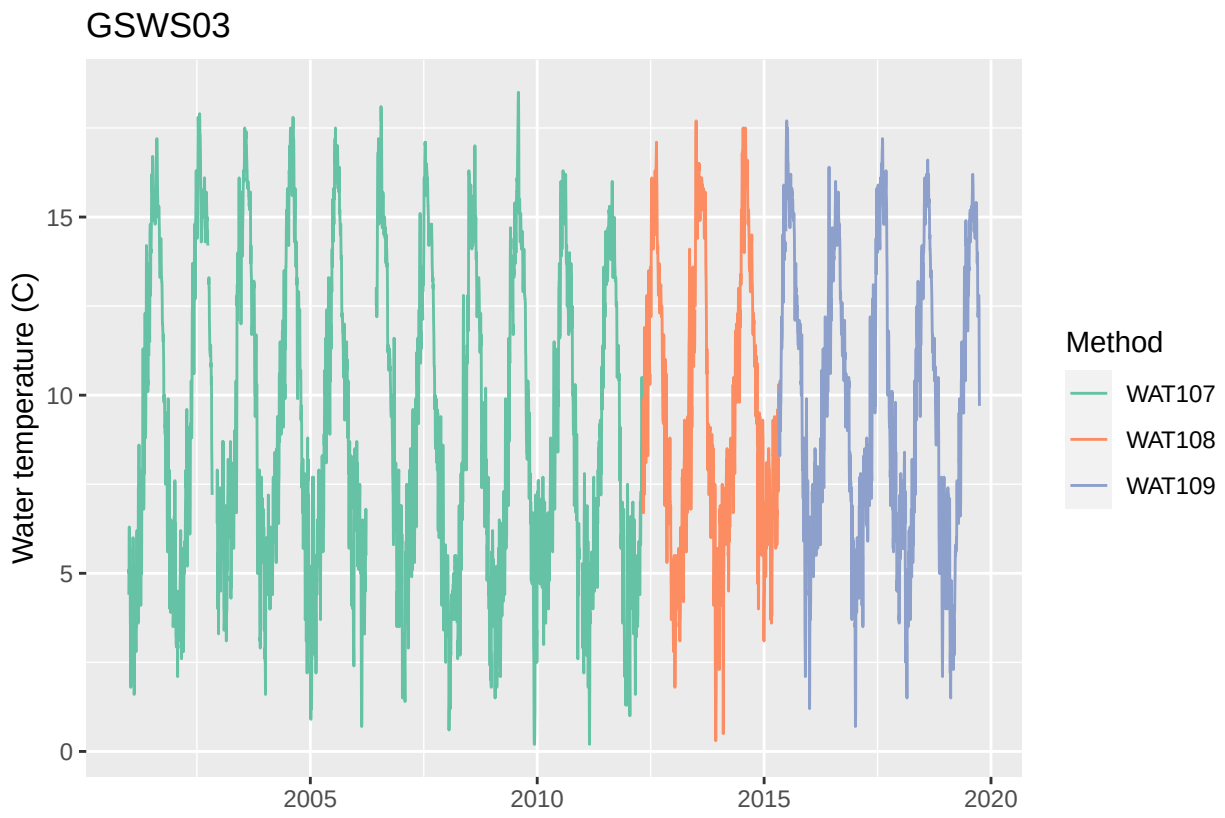

```
## Warning: Removed 2 row(s) containing missing values (geom_path).
```

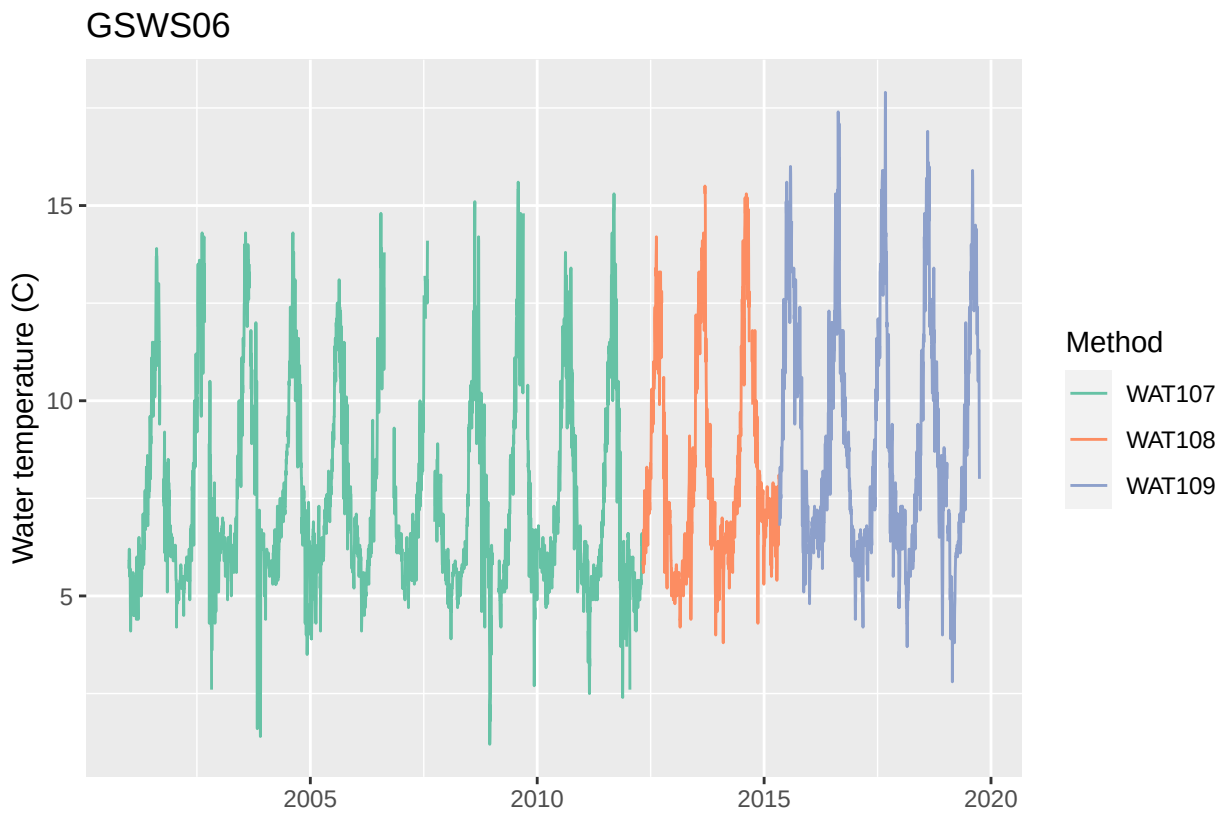


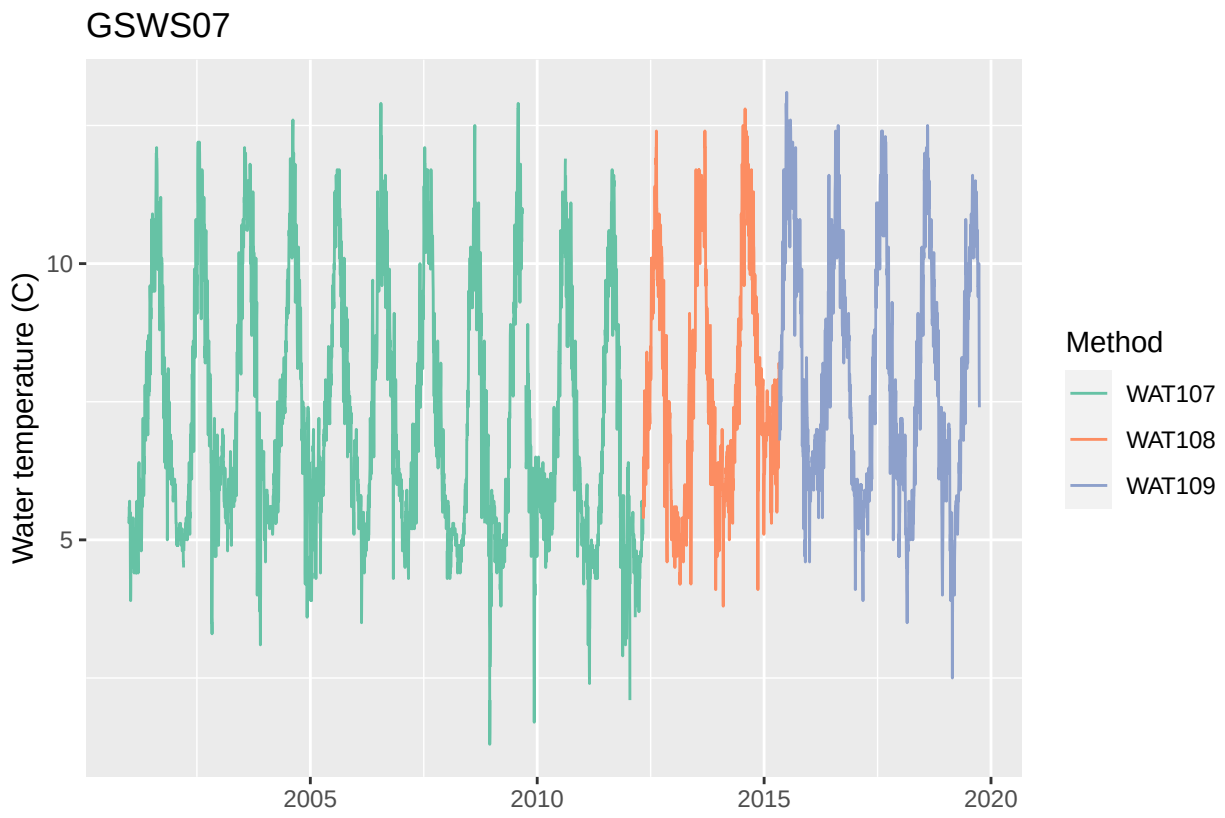

```
## Warning: Removed 2 row(s) containing missing values (geom_path).
```

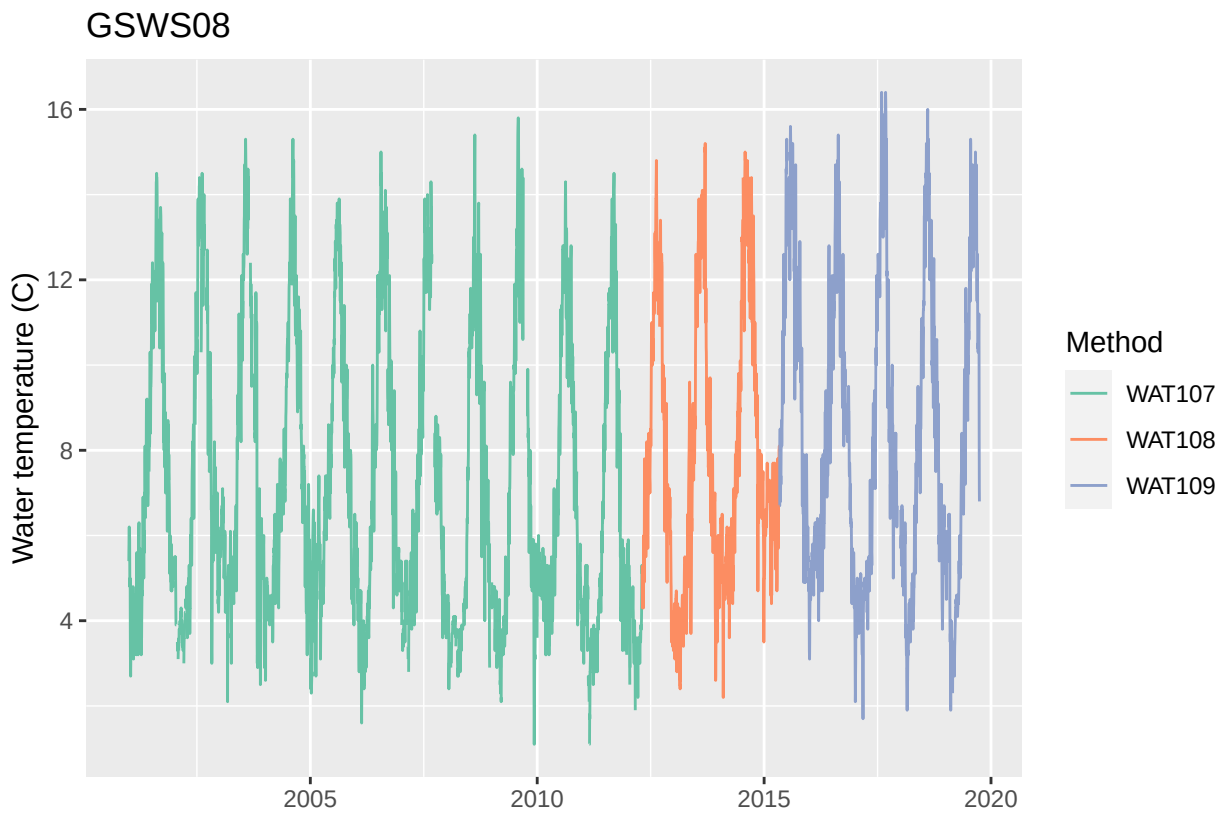



```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```

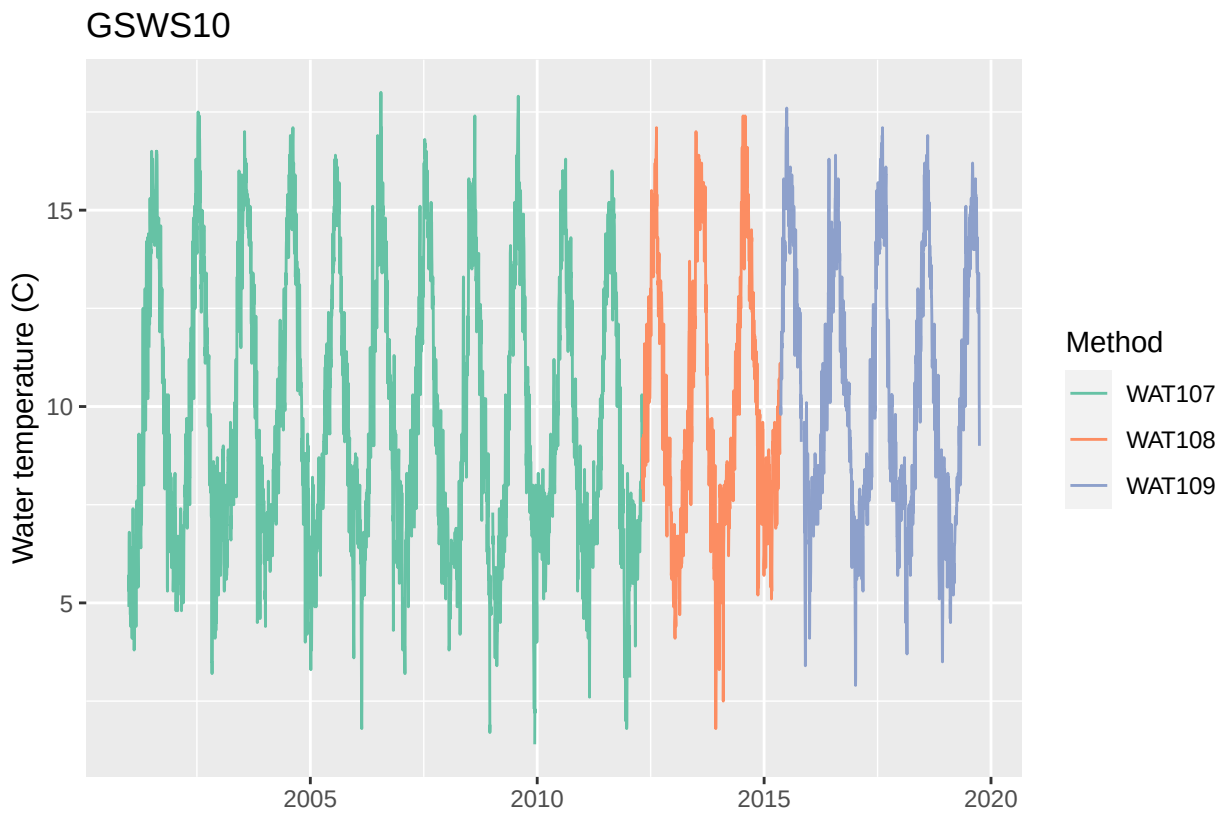

GSWS03
Water temperature (C)
0
5
10
15
2005
2010
2015
2020
Method
WAT107
WAT108
WAT109


GSWS06
Water temperature (C)
15
10
5
2005
2010
2015
2020
Method
WAT107
WAT108
WAT109

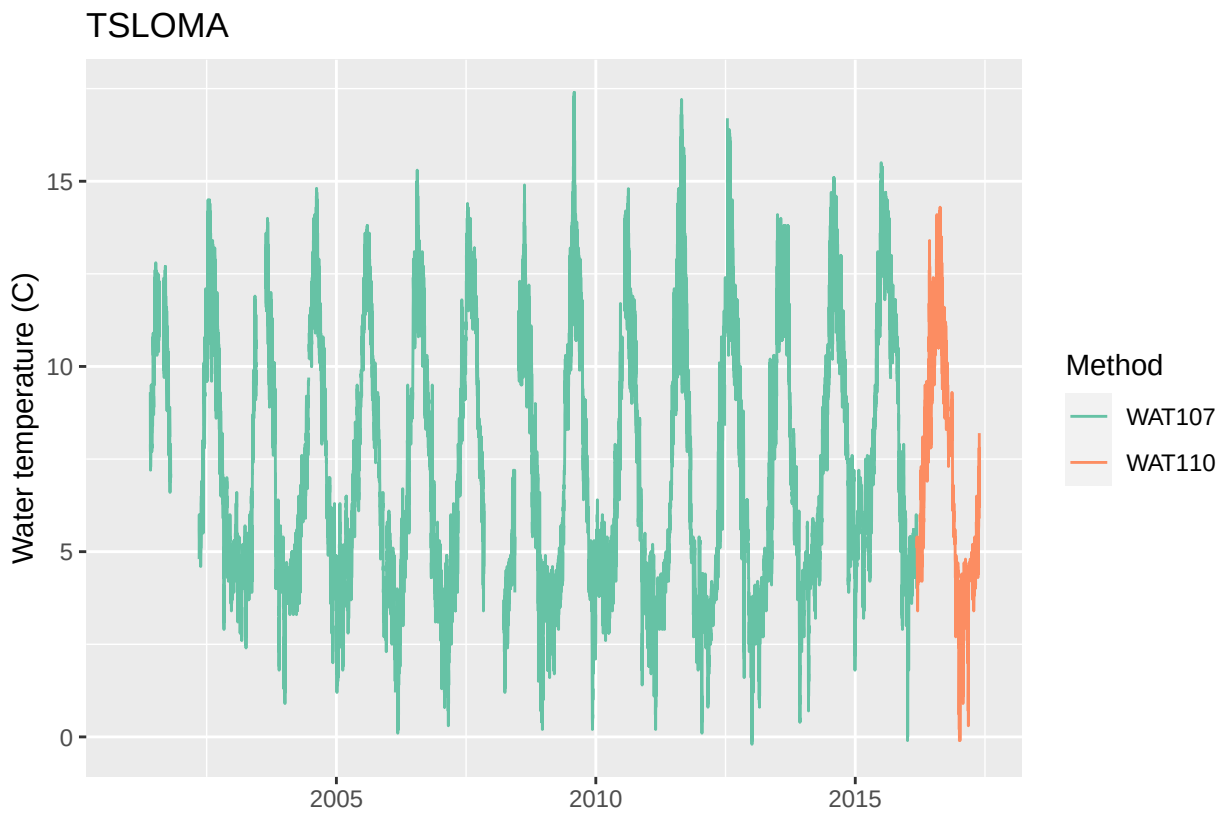

GSWS07
Water temperature (C)
10
5
2005
2010
2015
2020
Method
WAT107
WAT108
WAT109



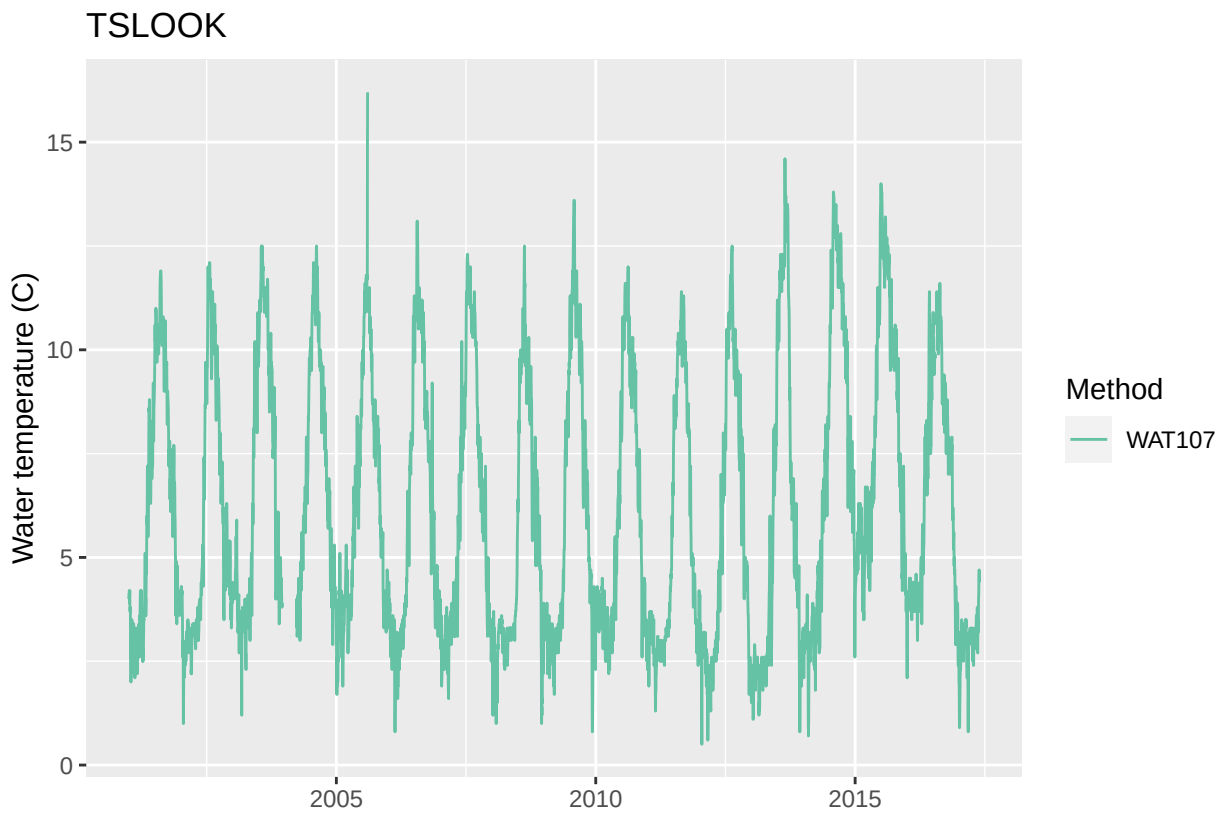

```
## Warning: Removed 4 row(s) containing missing values (geom_path).
```

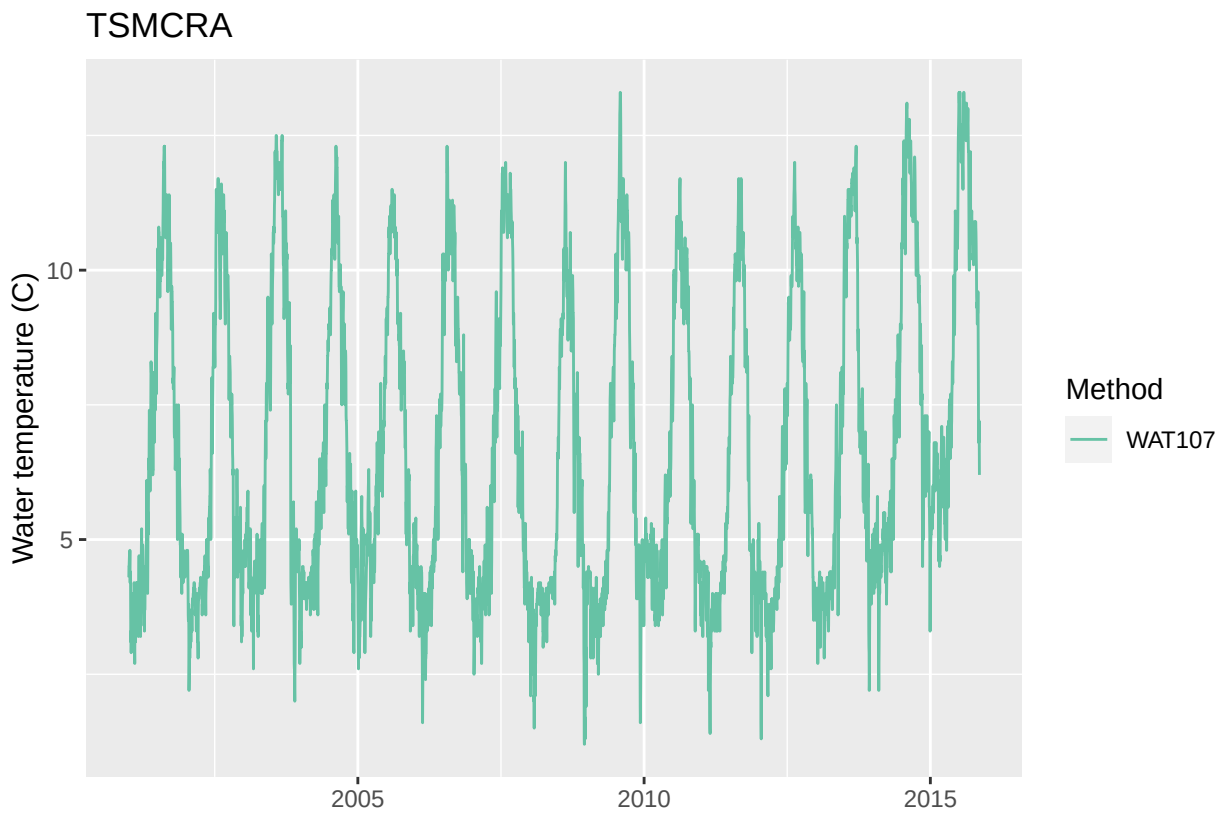



```
## Warning: Removed 4 row(s) containing missing values (geom_path).
```




```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```


TSLOOK
Water temperature (C)
15
10
5
0
2005
2010
2015
Method
WAT107



dplyrs `group_walk()` function (this both creates and displays the plots):

```
streams %>%
  group_by(sitecode) %>%
  group_walk(~ print(plot_temperature(.x, .y)))
```

A `for` loop:

```
sitecodes <- unique(streams$sitecode)
for (site in sitecodes){
  site_data <- filter(streams,
    sitecode == site)
  site_data %>%
    plot_temperature(site) %>%
    print()
}
```

Or alternatively save plots as PNG files. With purrrs `walk2()`

```
streams_with_plots %>%
  mutate(
    plot_files = paste0("plots/", sitecode, ".png")
  ) %>%
  with(walk2(plot_files, plots, ggsave))
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 5 row(s) containing missing values (geom_path).

## Saving 6.5 x 4.5 in image

## Warning: Removed 1 row(s) containing missing values (geom_path).

## Saving 6.5 x 4.5 in image

## Warning: Removed 2 row(s) containing missing values (geom_path).

## Saving 6.5 x 4.5 in image

## Warning: Removed 2 row(s) containing missing values (geom_path).

## Saving 6.5 x 4.5 in image

## Warning: Removed 1 row(s) containing missing values (geom_path).

## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image
## Saving 6.5 x 4.5 in image

## Warning: Removed 4 row(s) containing missing values (geom_path).

## Saving 6.5 x 4.5 in image

## Warning: Removed 4 row(s) containing missing values (geom_path).

## Saving 6.5 x 4.5 in image

## Warning: Removed 1 row(s) containing missing values (geom_path).

## Saving 6.5 x 4.5 in image
```