

# Strava Analysis 2020

Dubravko Dolic

30 12 2020

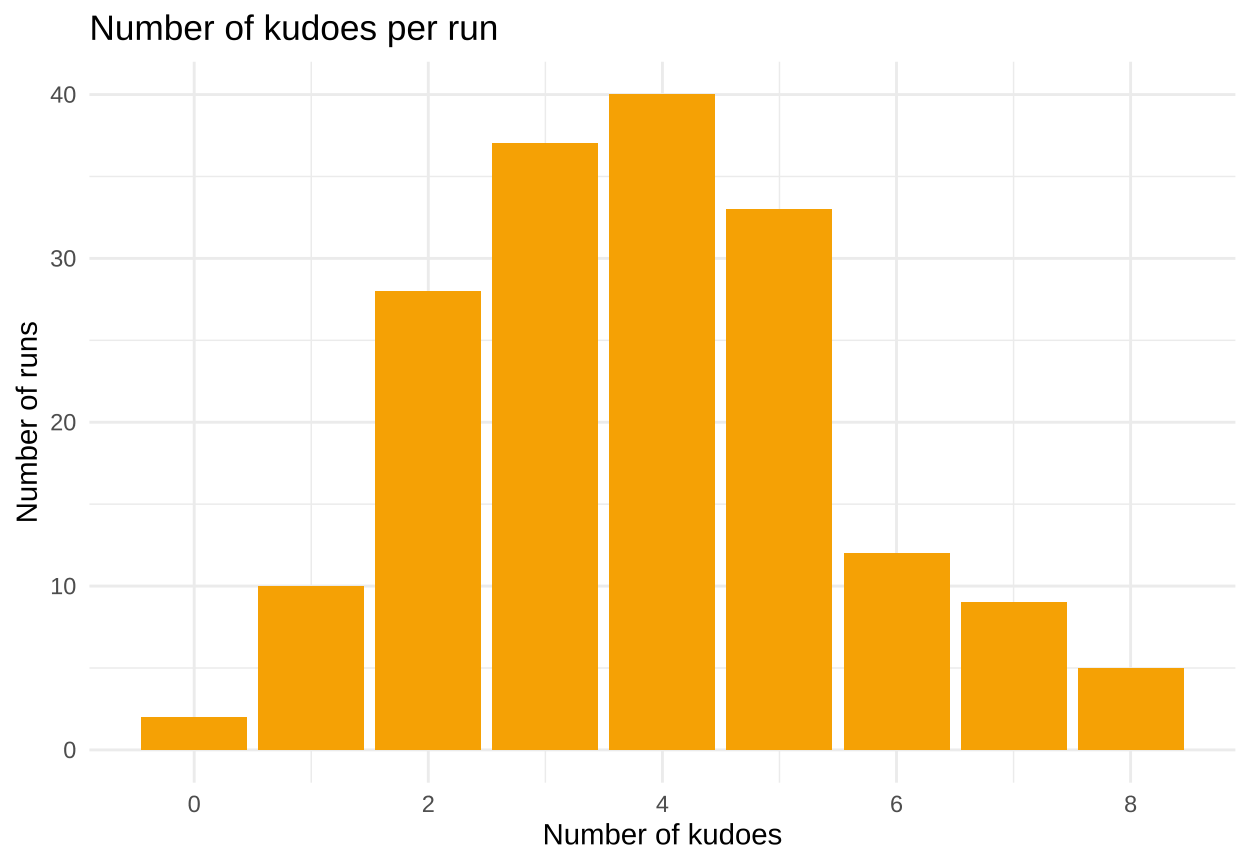
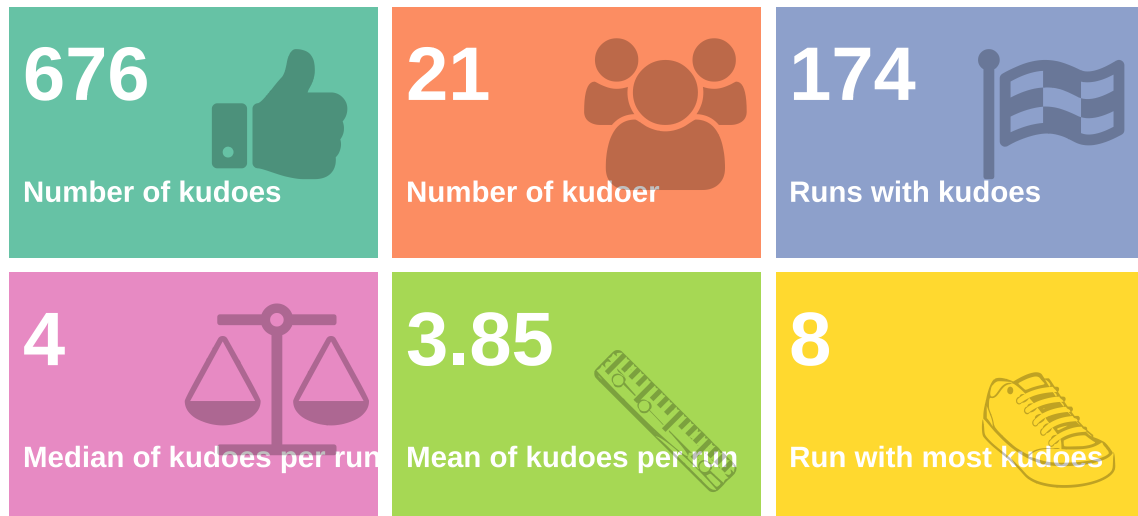
## Running 2020 - all runs summary

The following visualizations are done with data from Strava. Although Strava itself already allows a good deal of analyses I was curious what else could be done. So I checked out the API and here we are.

### Kudoers

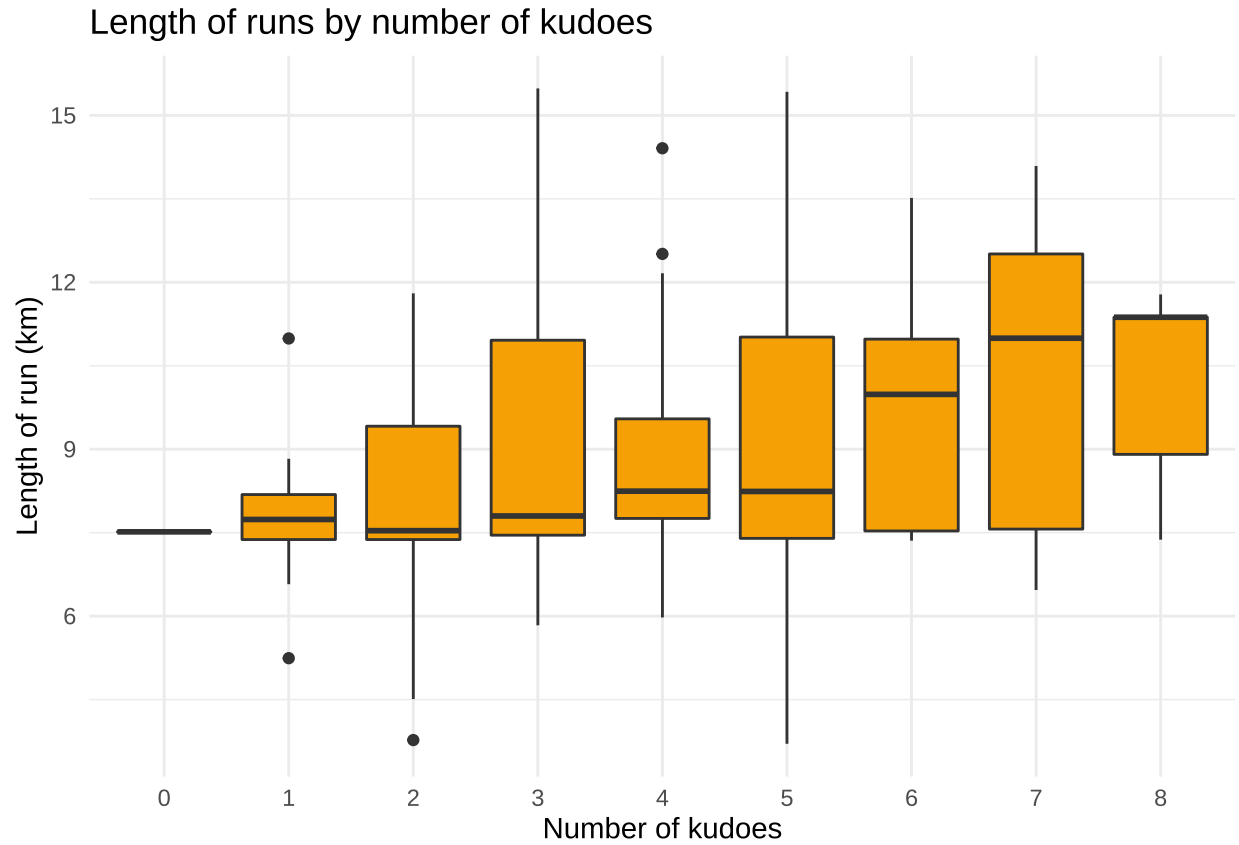
Let's have a look at the most exiting category: *Kudos*. These little somethings light up our days after a run. It can be a second reward for a run - apart from the run itself. So let's have a closer look.

## Number of kudoes and kudoers

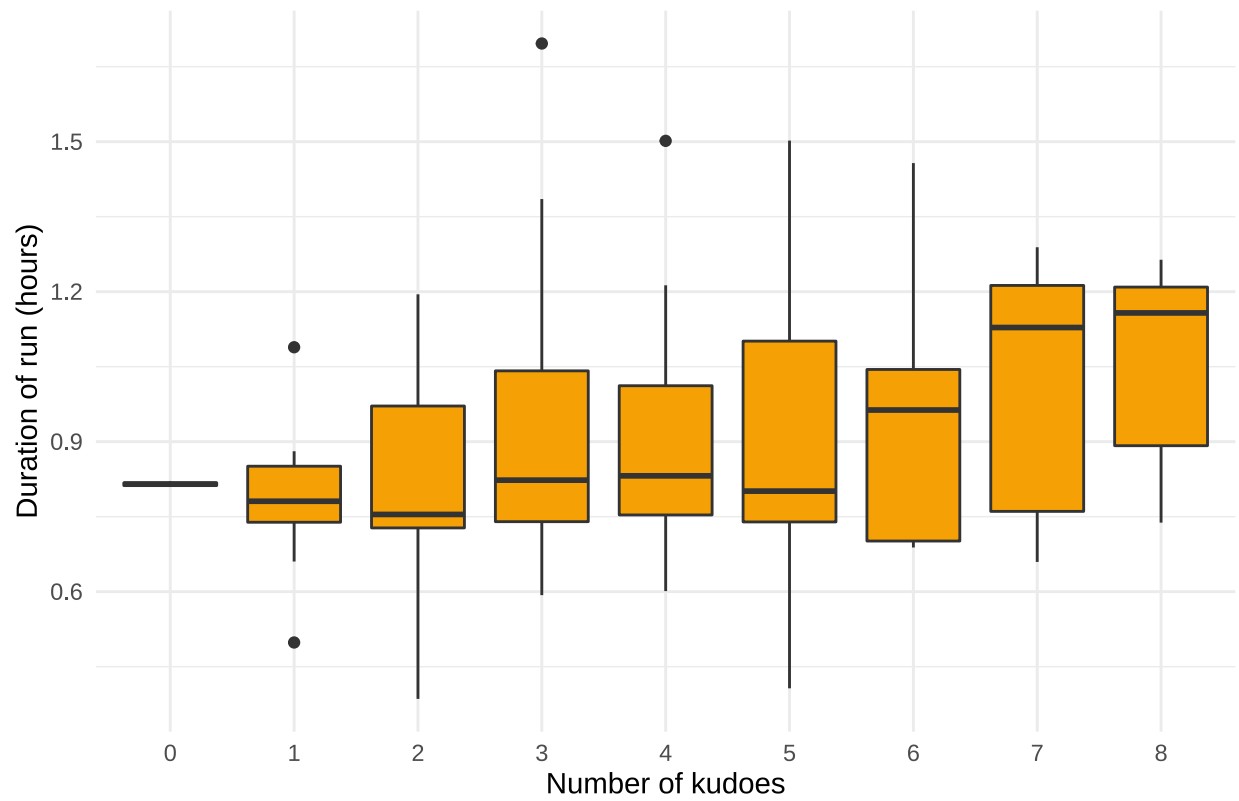


So, not a vast amount of kudoers but a very nice number of kudoes recieved already. Happy about that and thankful for this huge amount of motivation.

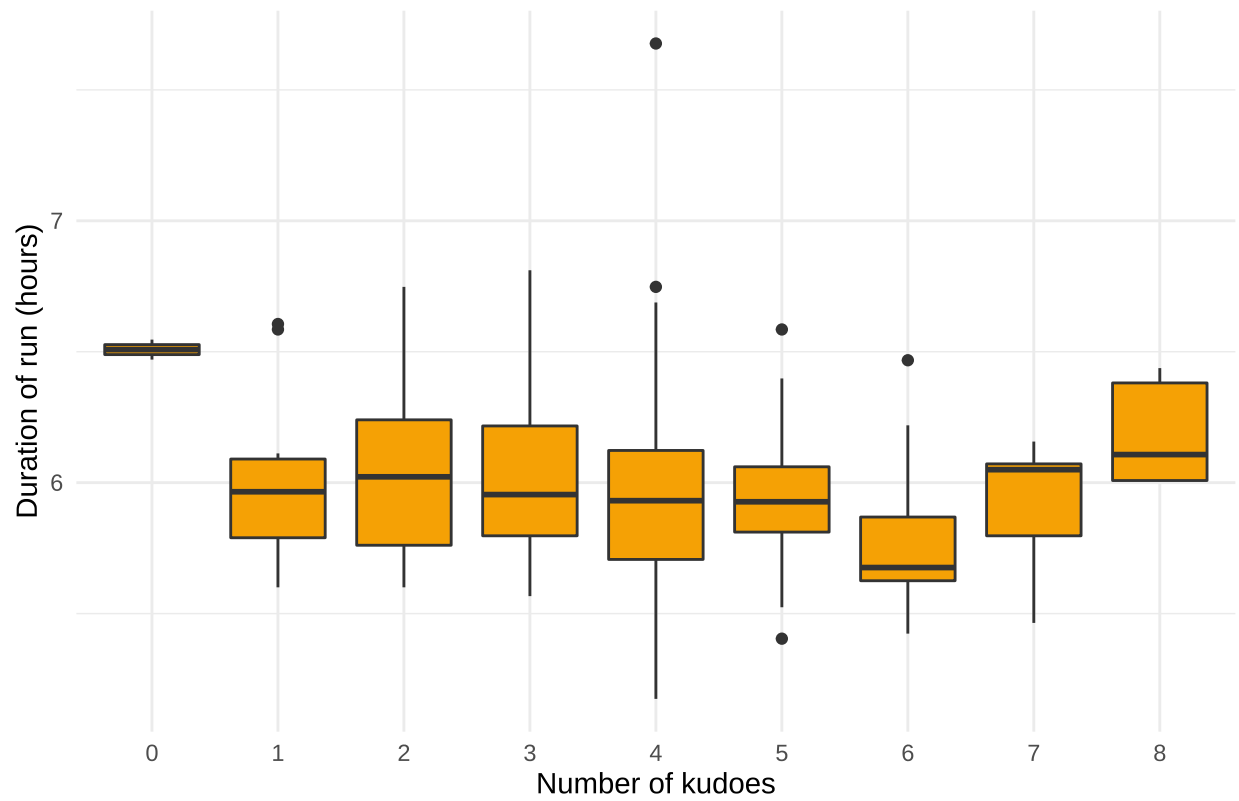
Now let's see what I could do to receive more kudos next year. What is it that drives the number of kudoes? Is it the distance I do? Maybe the duration? Or do I have to be real fast?



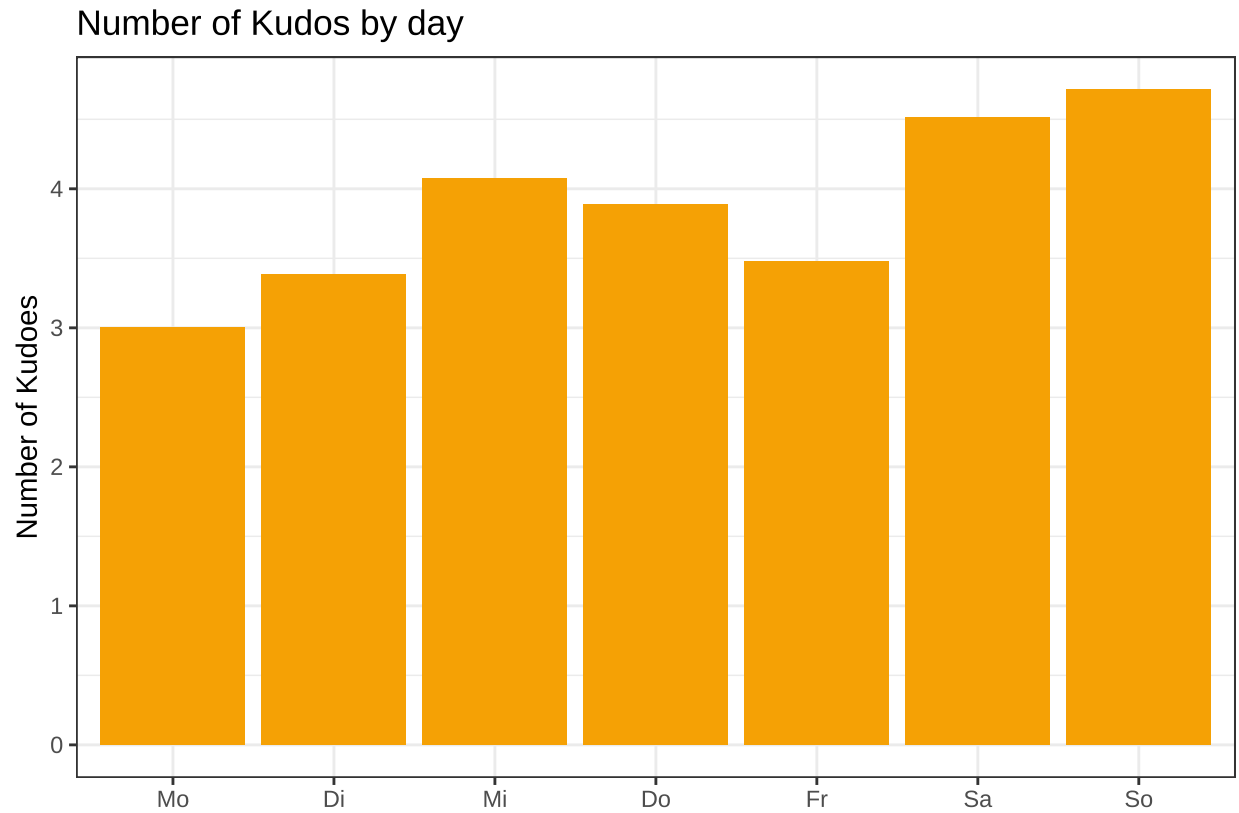
Duration of runs by number of kudoes



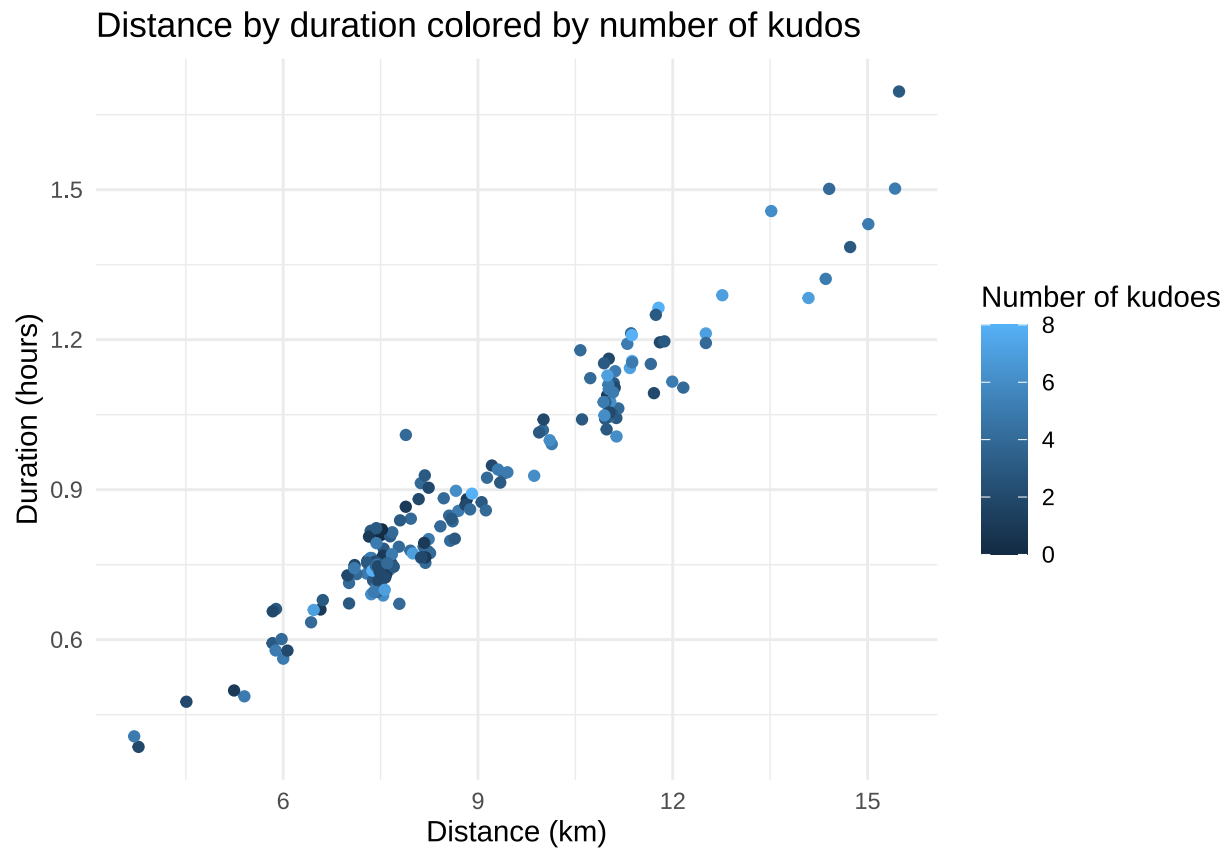
Pace by runs by number of kudoes



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



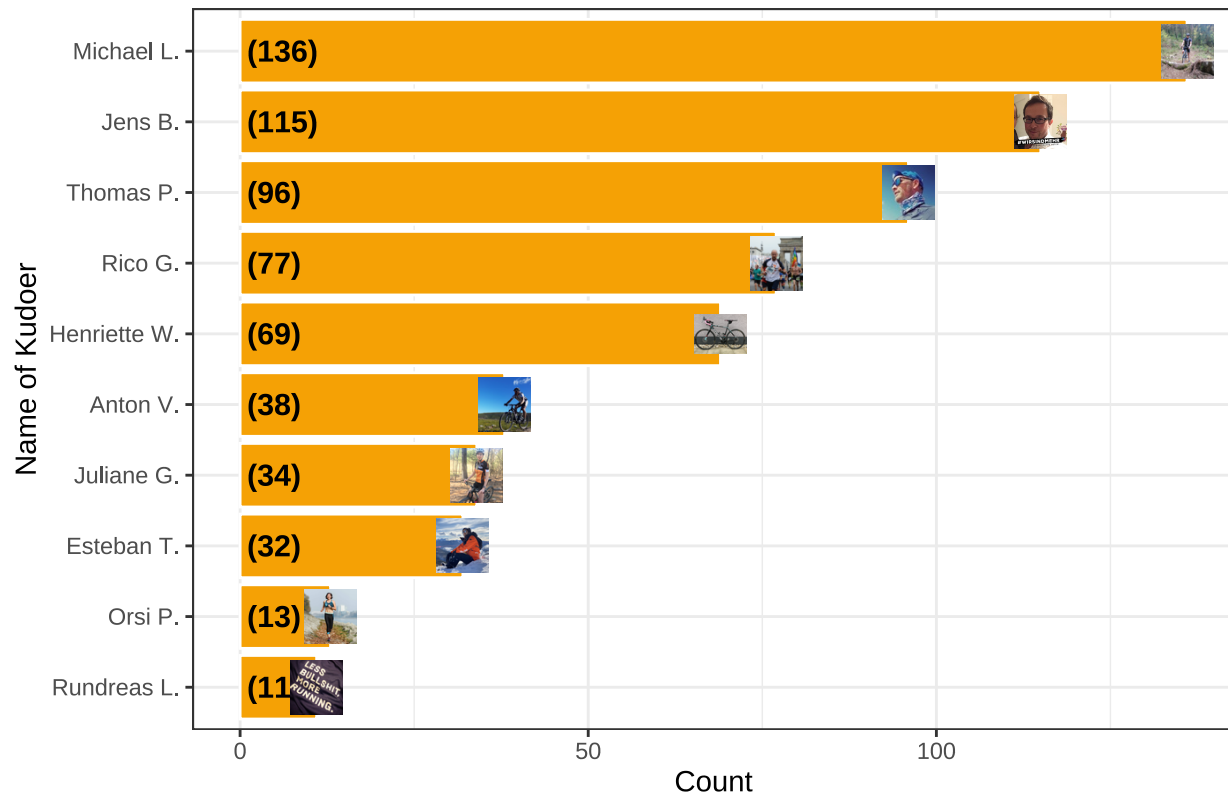
While Distance and duration seem to trigger my followers pace doesn't. Also the day seems to be without any influence although on Saturdays everyone seems to have more time for leaving a thumbs-up. Overall it looks like this:



Now here is the most exiting chart: who is doing the best job as motivator? The answer is...

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

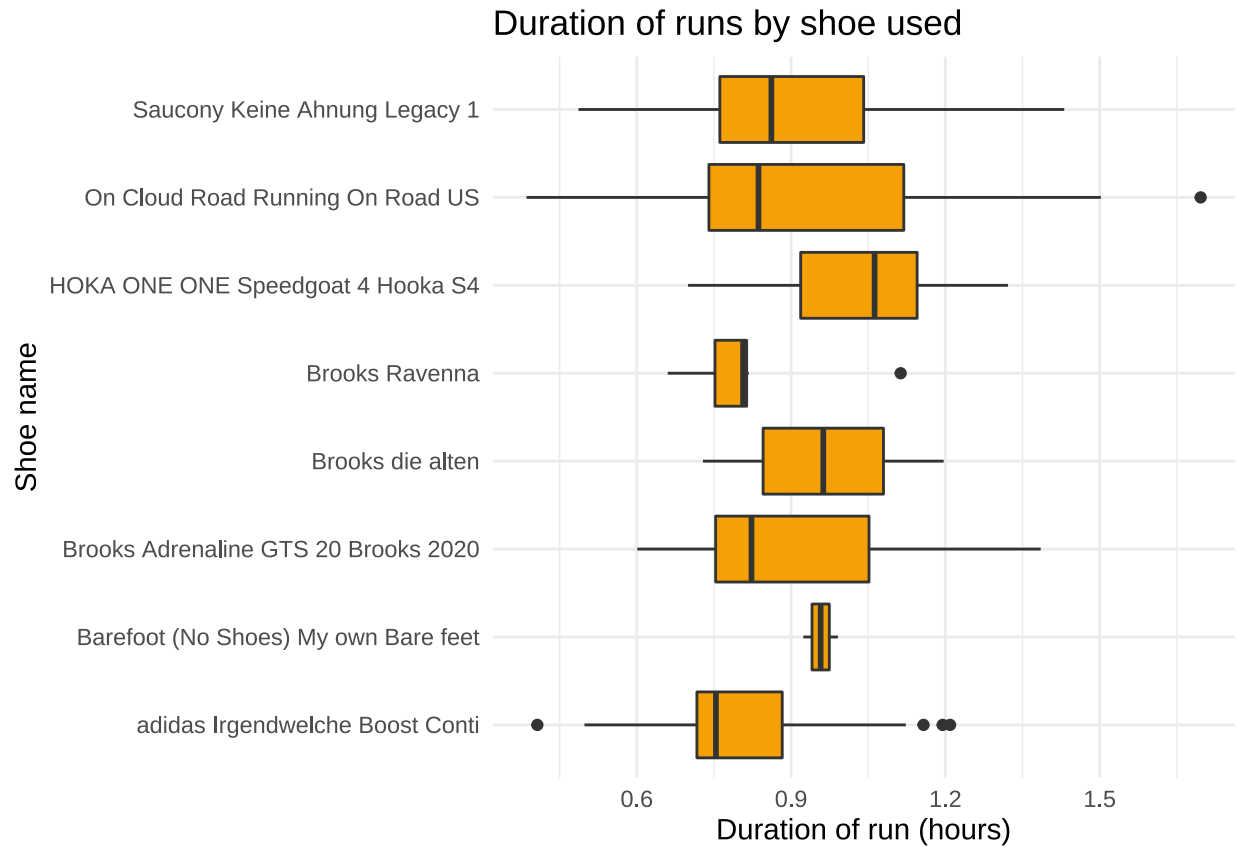
## My top Kudoers 2020



## My shoes

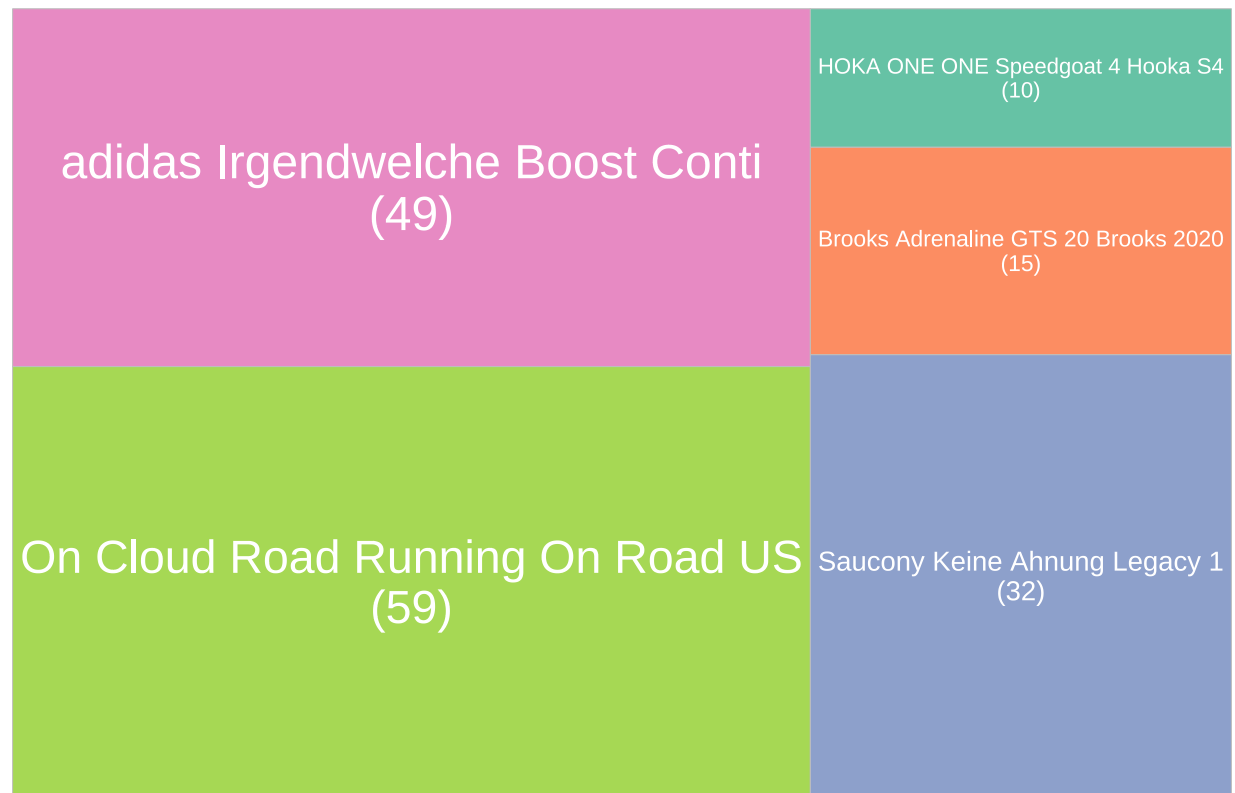
Clearly shoes are the most important gear for every runner. So I have a closer look on my gear. While Strava allows me to track the mileage of each shoe, I was interested on other parameters like speed, frequency of wear and duration.



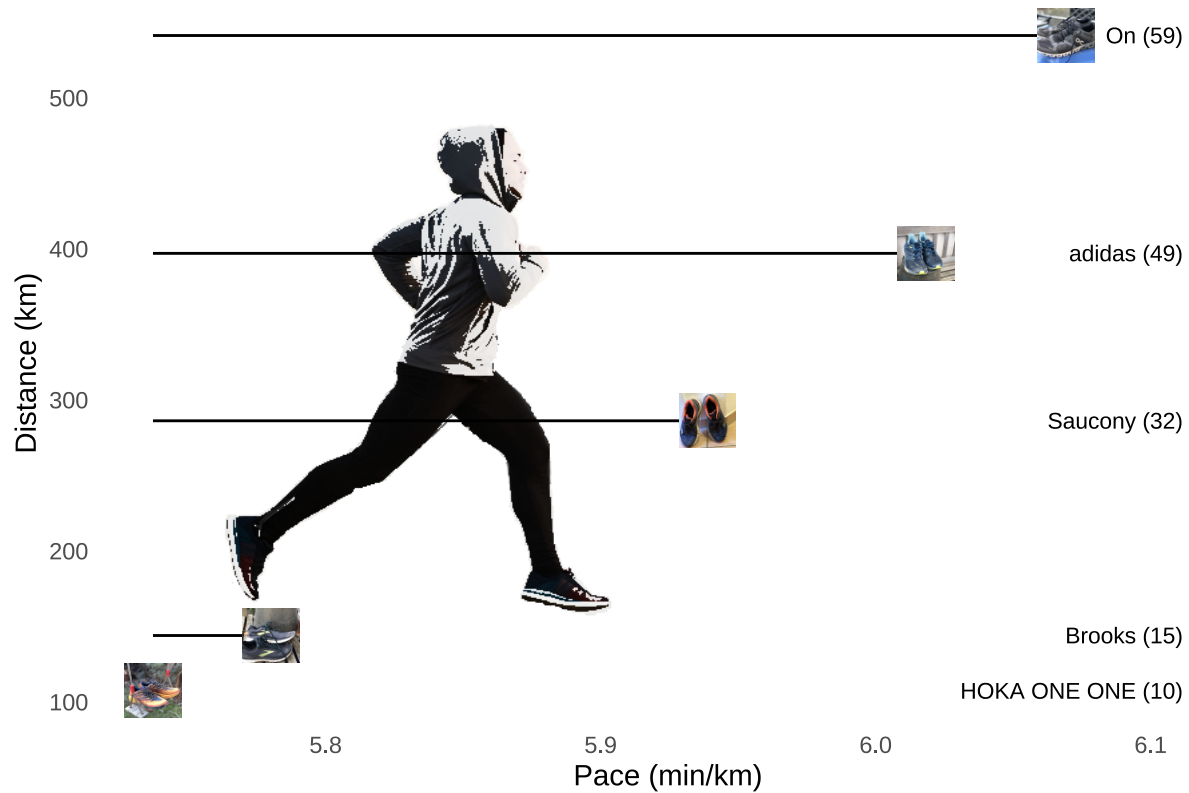


```
## `summarise()` regrouping output by 'shoe' (override with `.groups` argument)
```

## Shoes: Frequency of wear

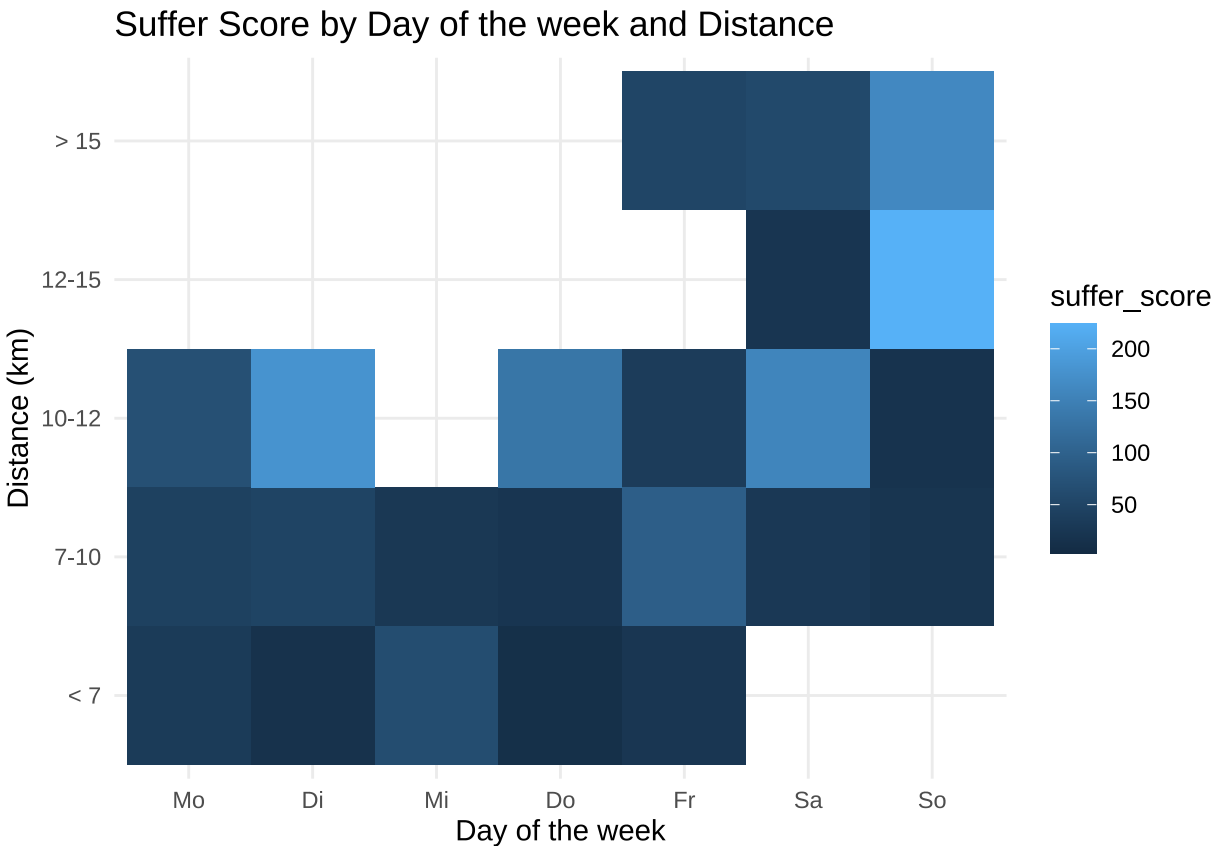
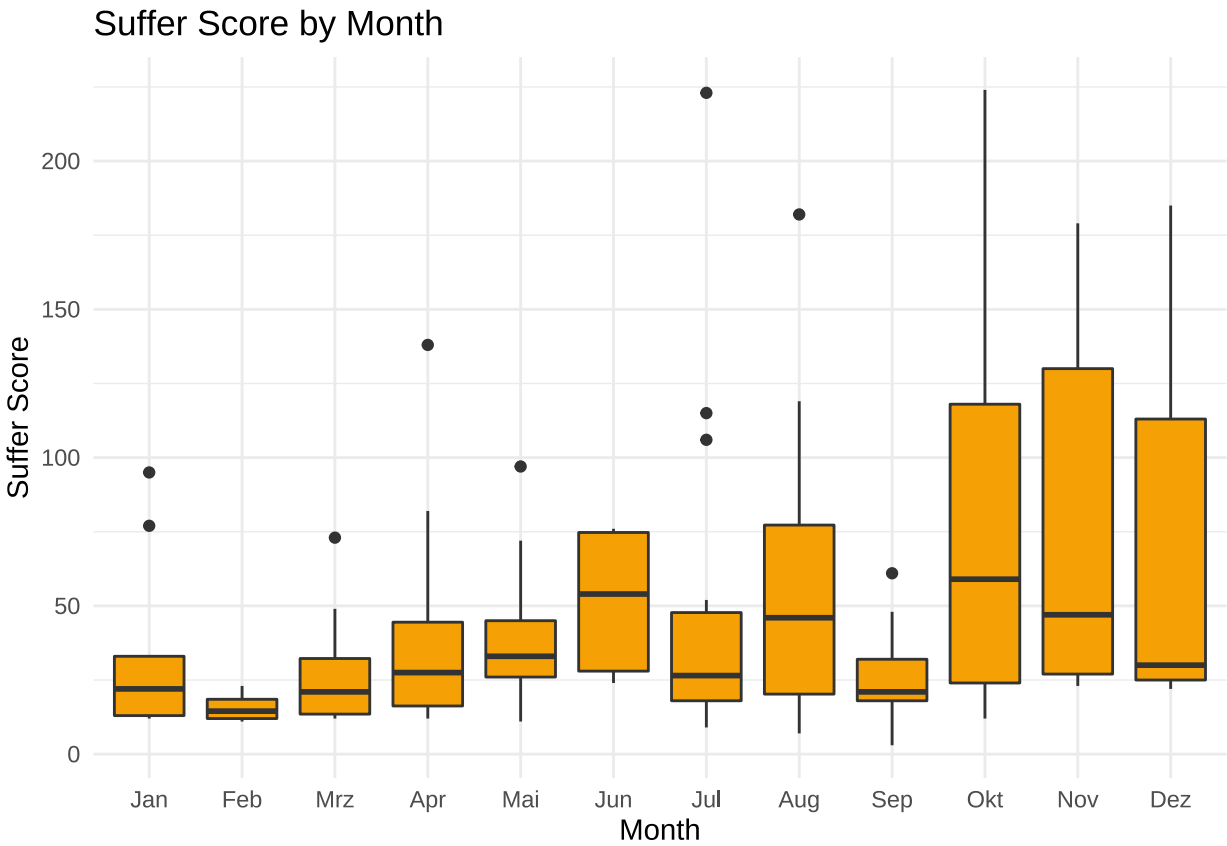


## Shoes by Distance and Pace



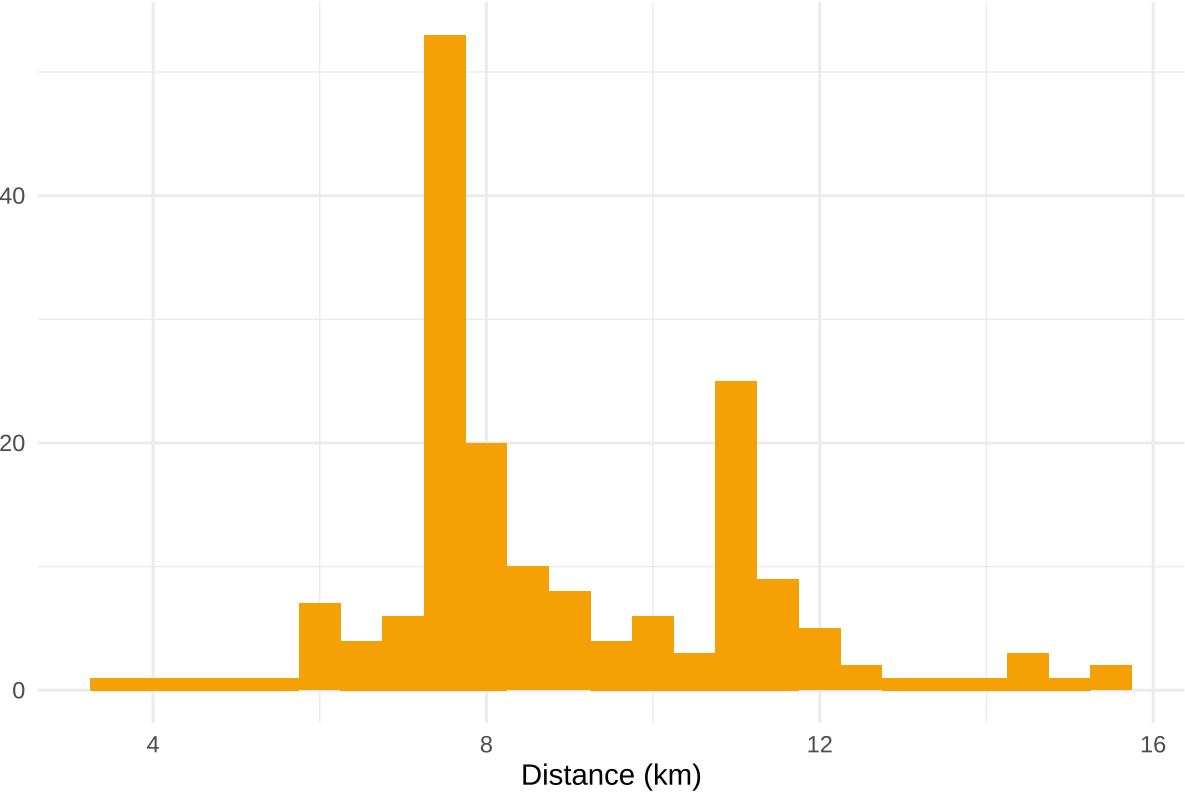
While my Hoka ONE ONE are giving me pace the ON shoes were my most frequently used one. Actually I was a little bit astonished because the usage was less than I expected.

Suffer Score

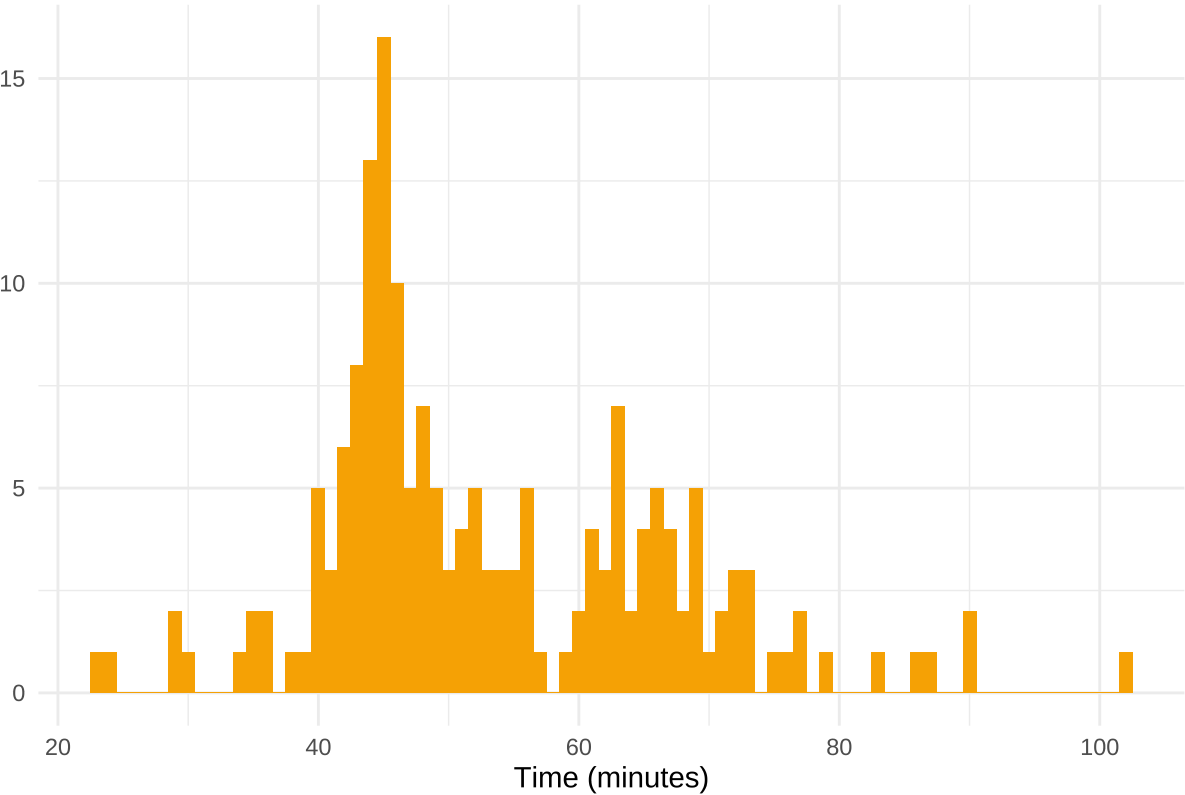


Runs

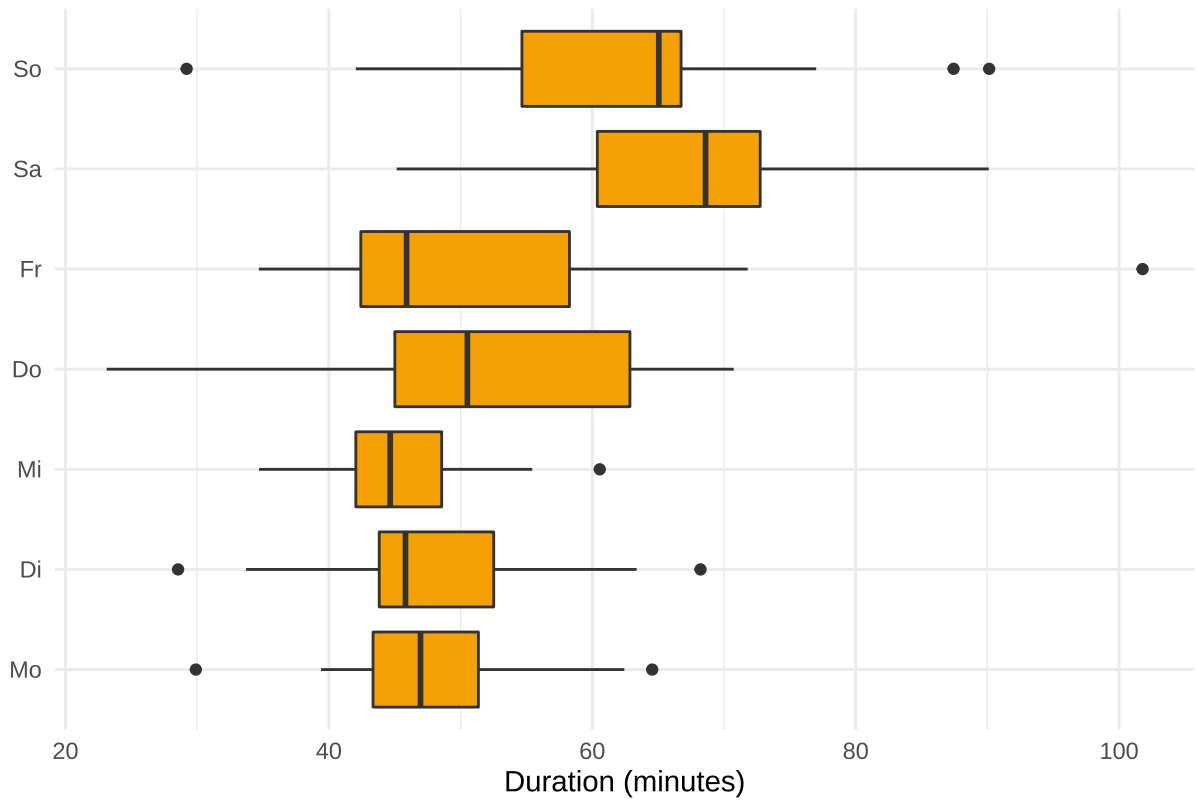
Distribution of length

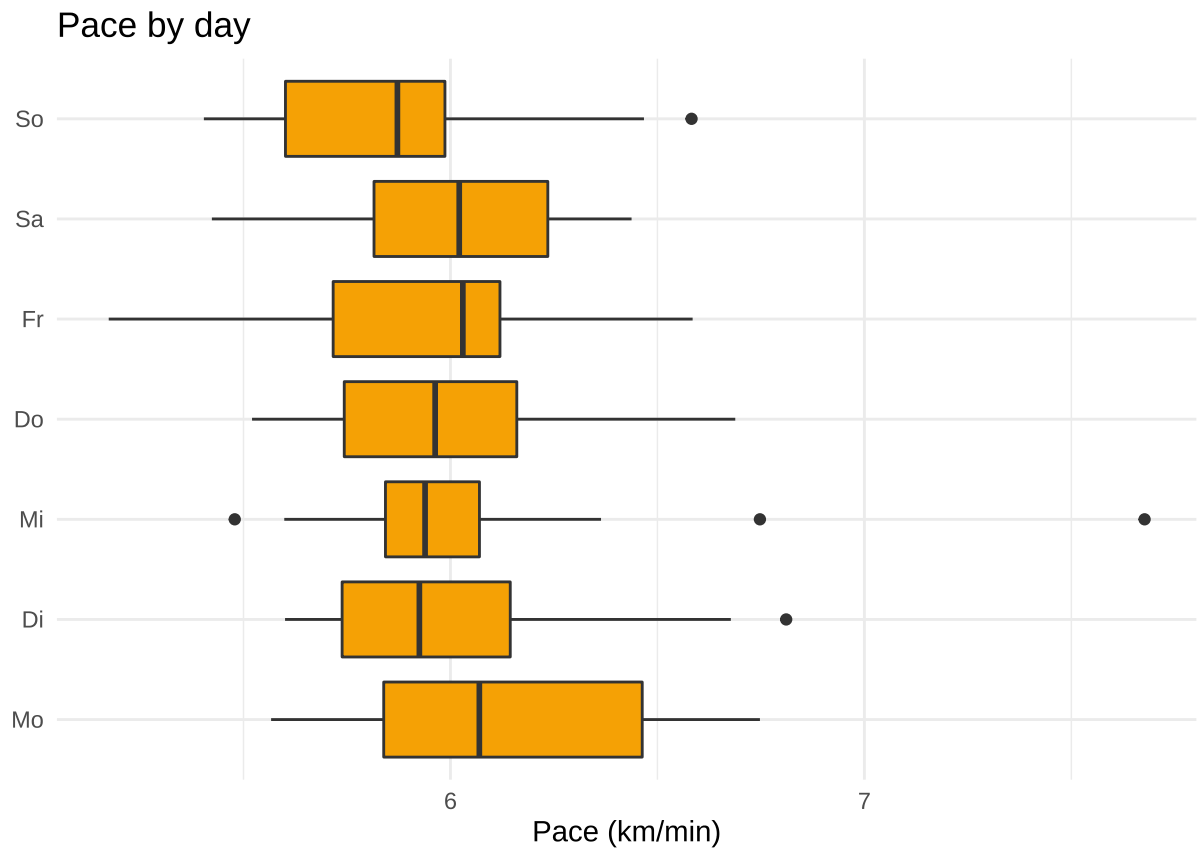


Distribution of duration

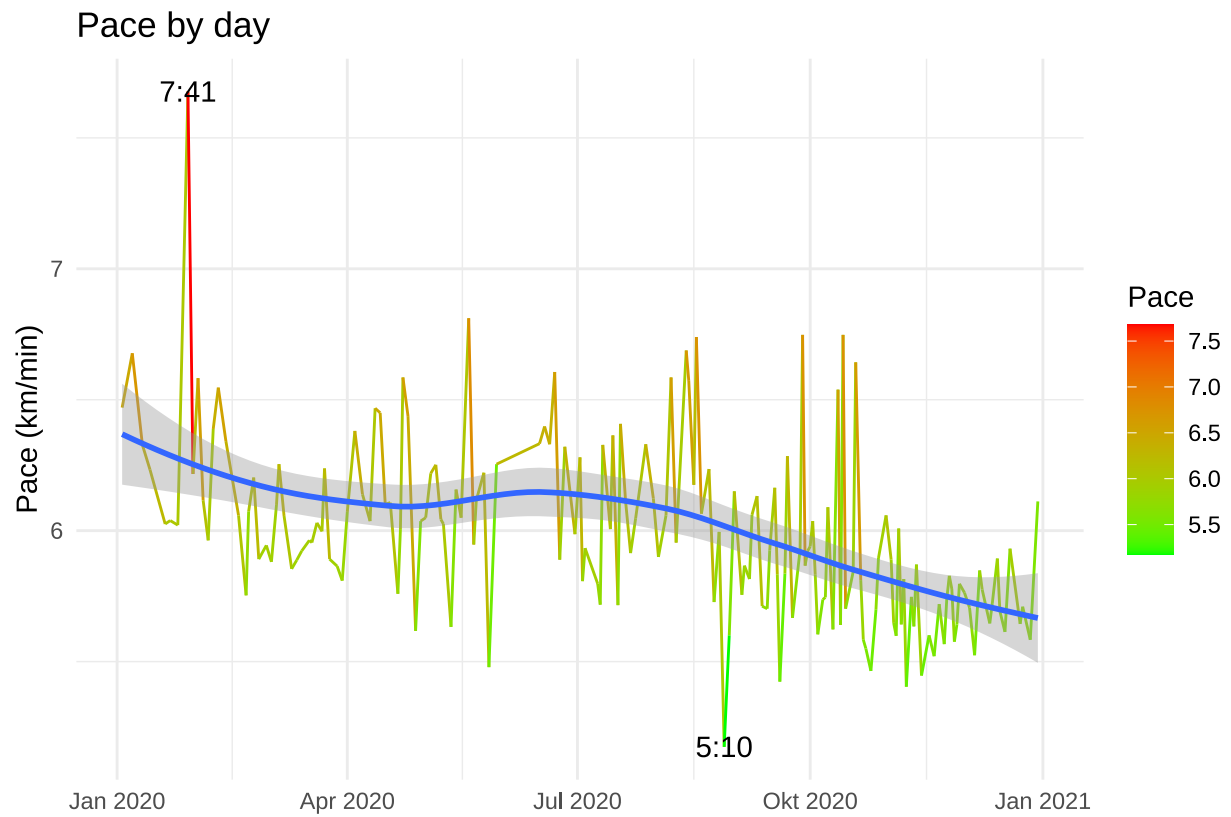


Length of runs by day

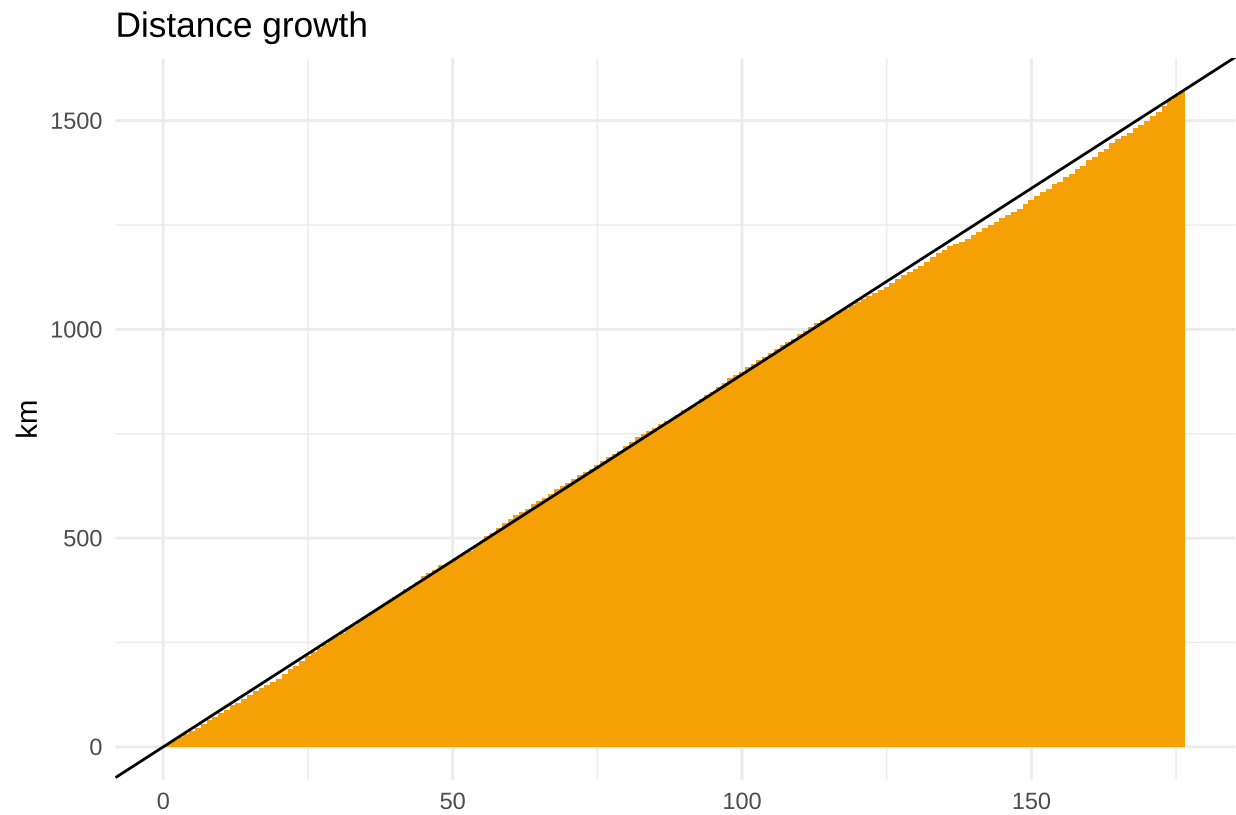




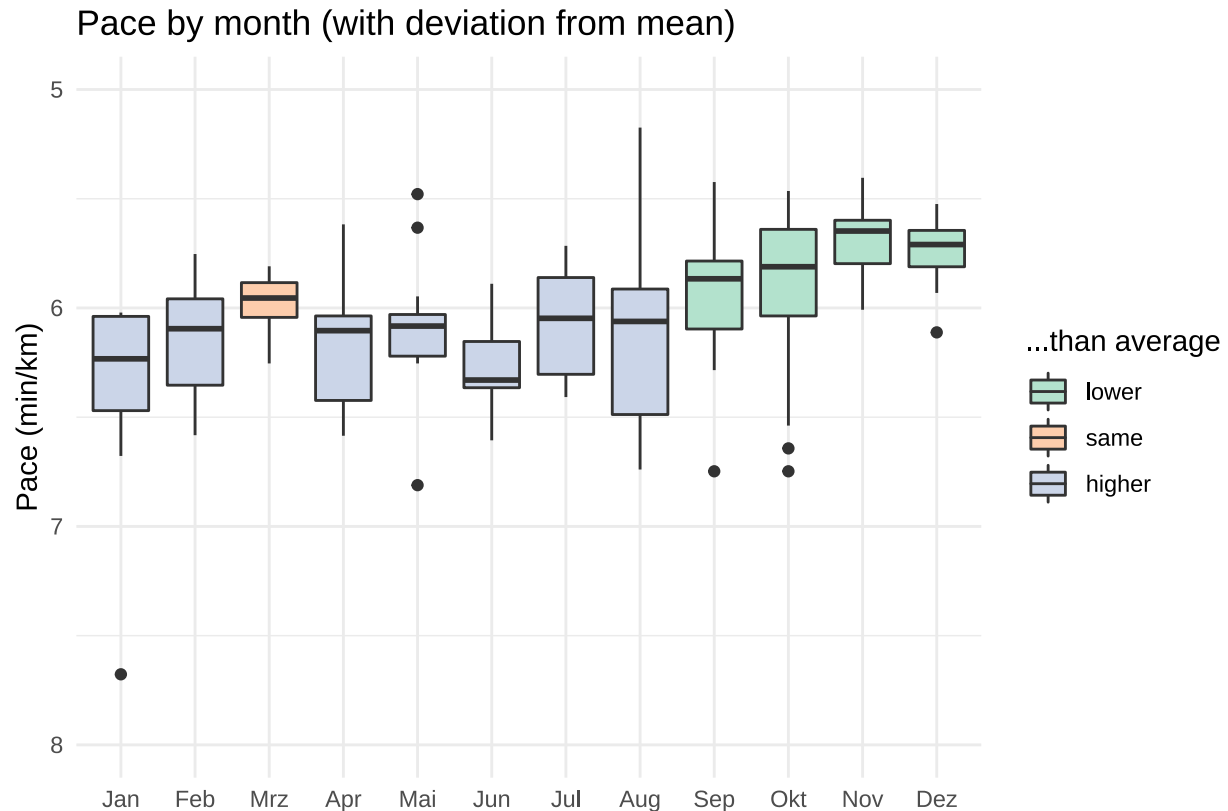
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```







```
## Loading required package: data.table
##
## Attaching package: 'data.table'
## The following objects are masked from 'package:dplyr':
##
##   between, first, last
```



```
## Picking joint bandwidth of 0.122

## Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x$label)): font family
## 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x$label)): font family
## 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x$label)): font family
## 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x$label)): font family
## 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x$label)): font family
## 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x$label)): font family
## 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x$label)): font family
## 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x$label)): font family
## 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x$label)): font family
## 'Arial Narrow' not found, will use 'sans' instead
```

[illegible]







[illegible]





[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]



[illegible]

```

## font family 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## font family 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## font family 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## font family 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## family 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## family 'Arial Narrow' not found, will use 'sans' instead

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, : font
## family 'Arial Narrow' not found, will use 'sans' instead

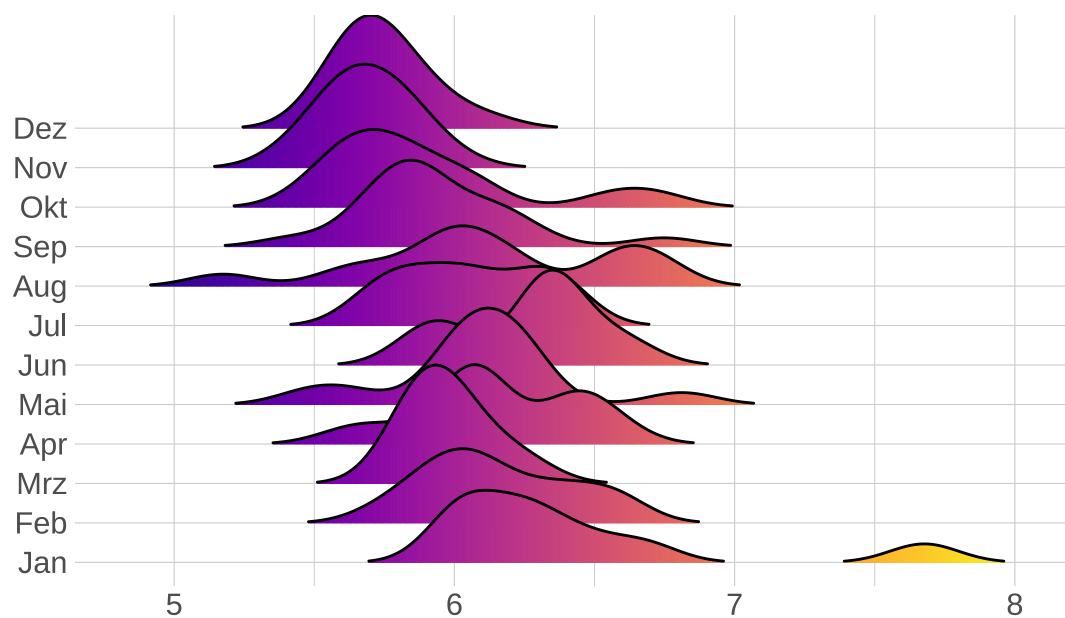
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## font family 'Arial Narrow' not found, will use 'sans' instead

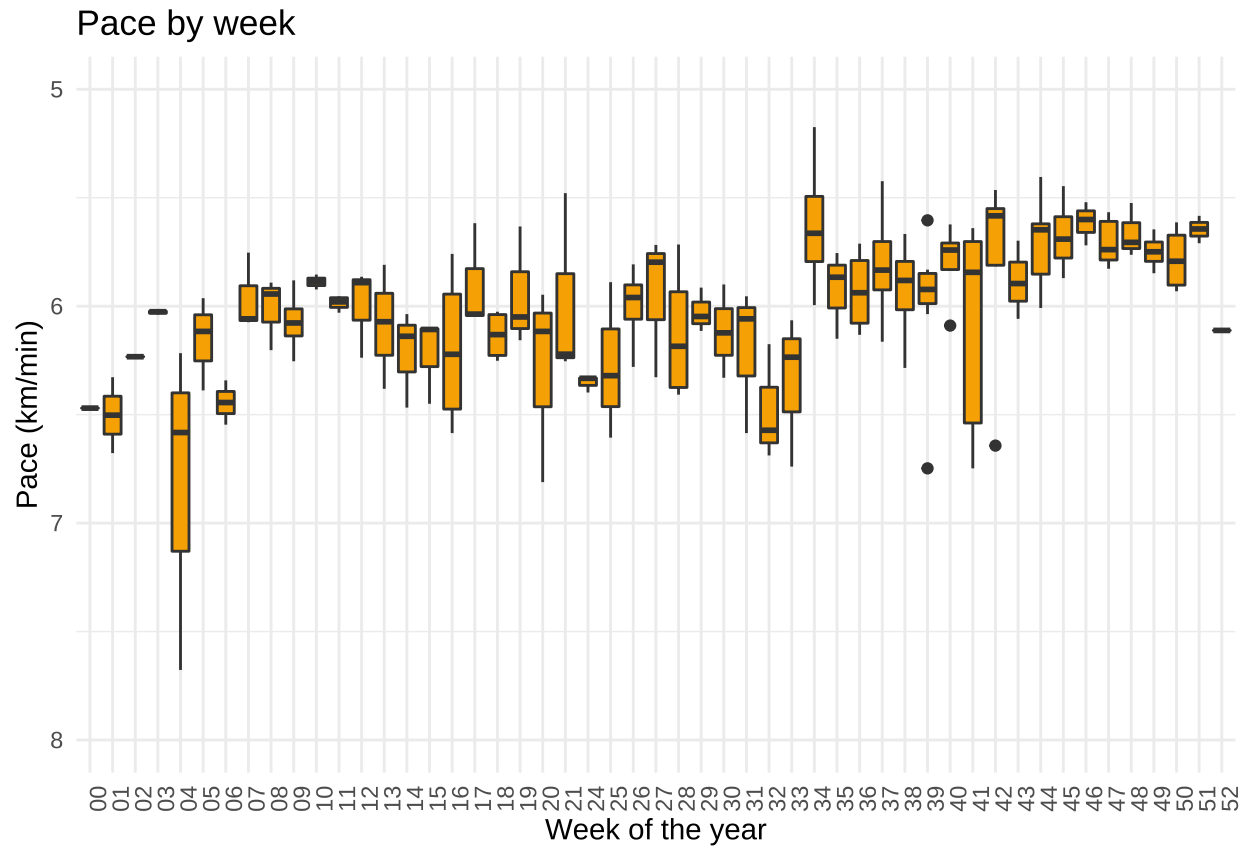
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## font family 'Arial Narrow' not found, will use 'sans' instead

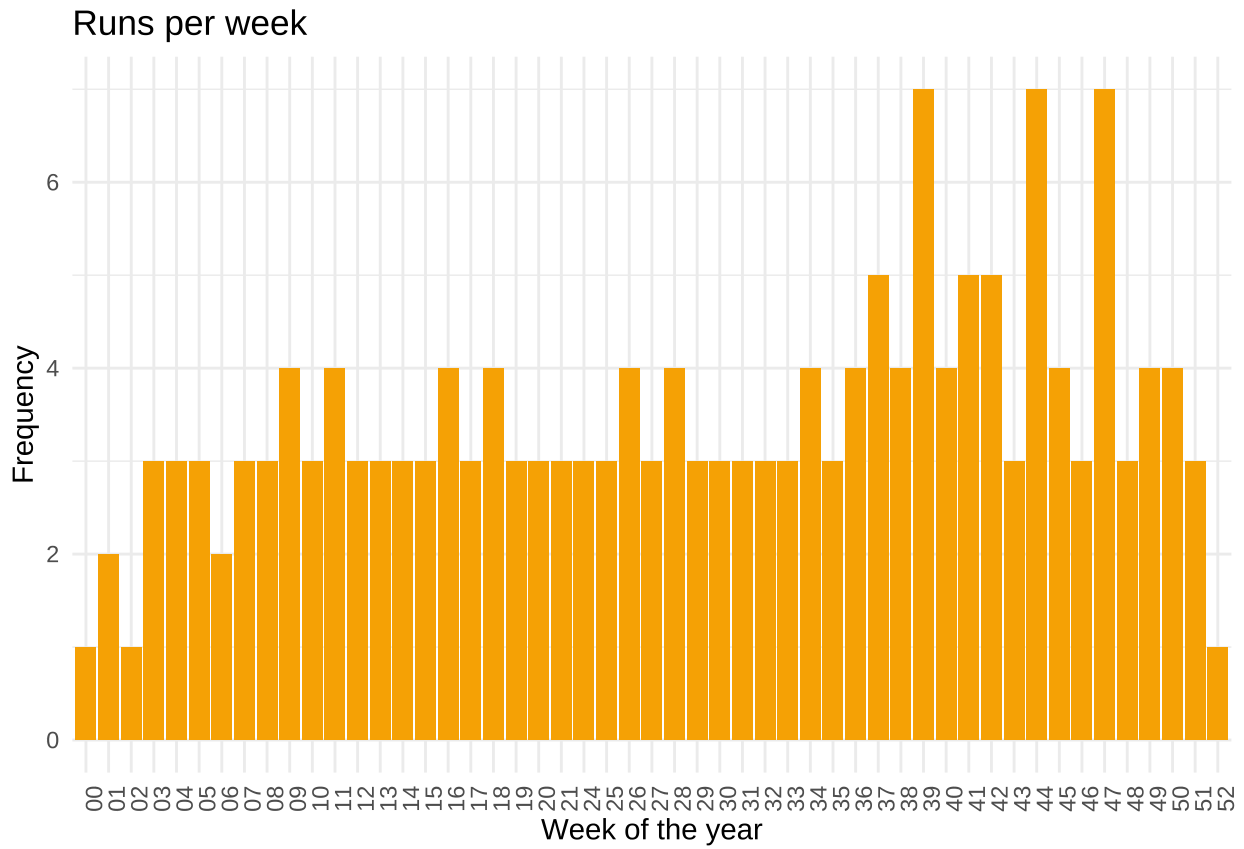
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## font family 'Arial Narrow' not found, will use 'sans' instead

```

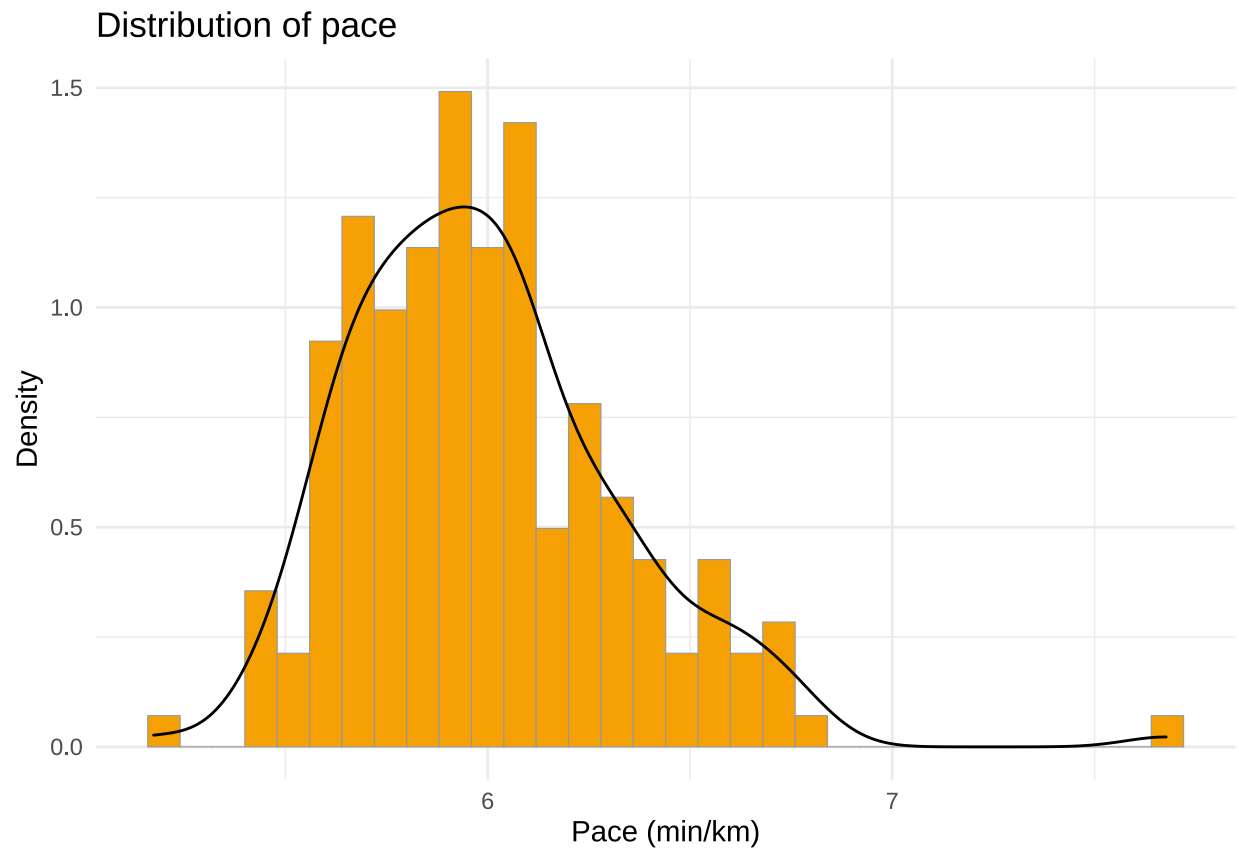
## Pace (Min/km) by month



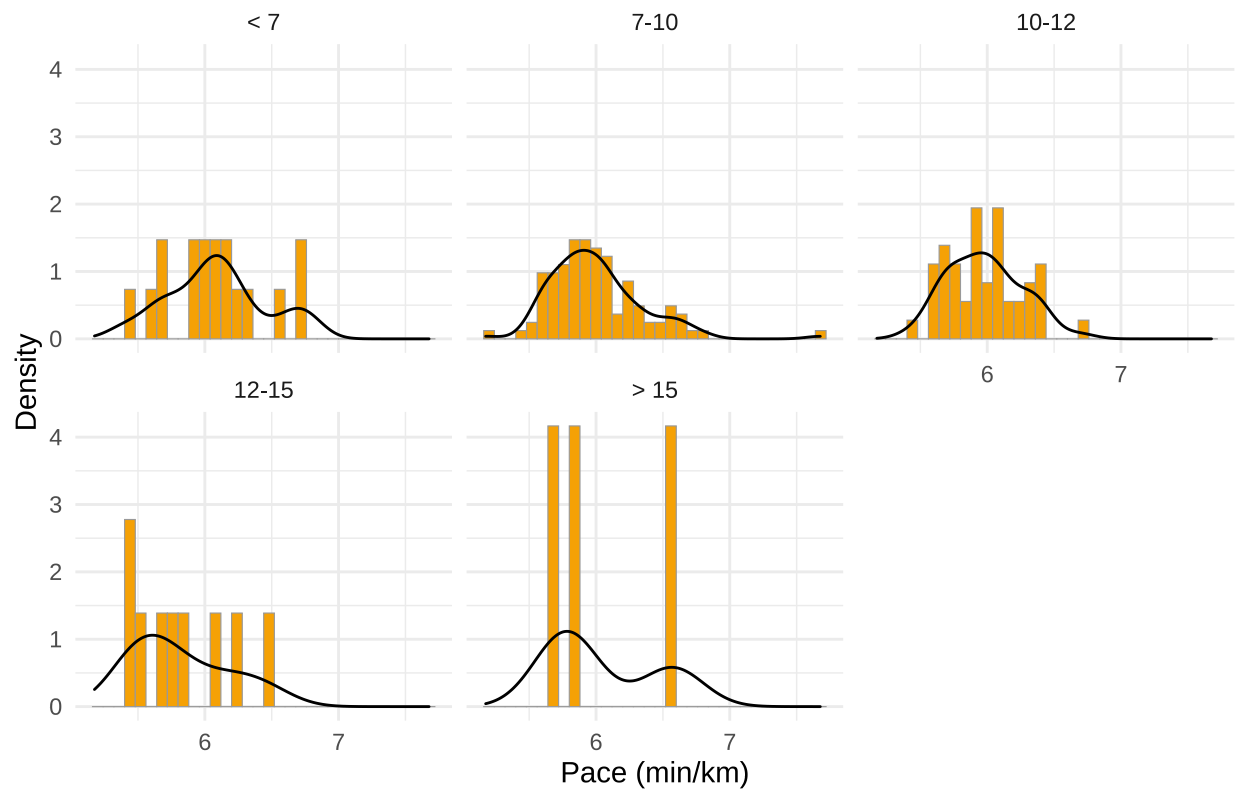




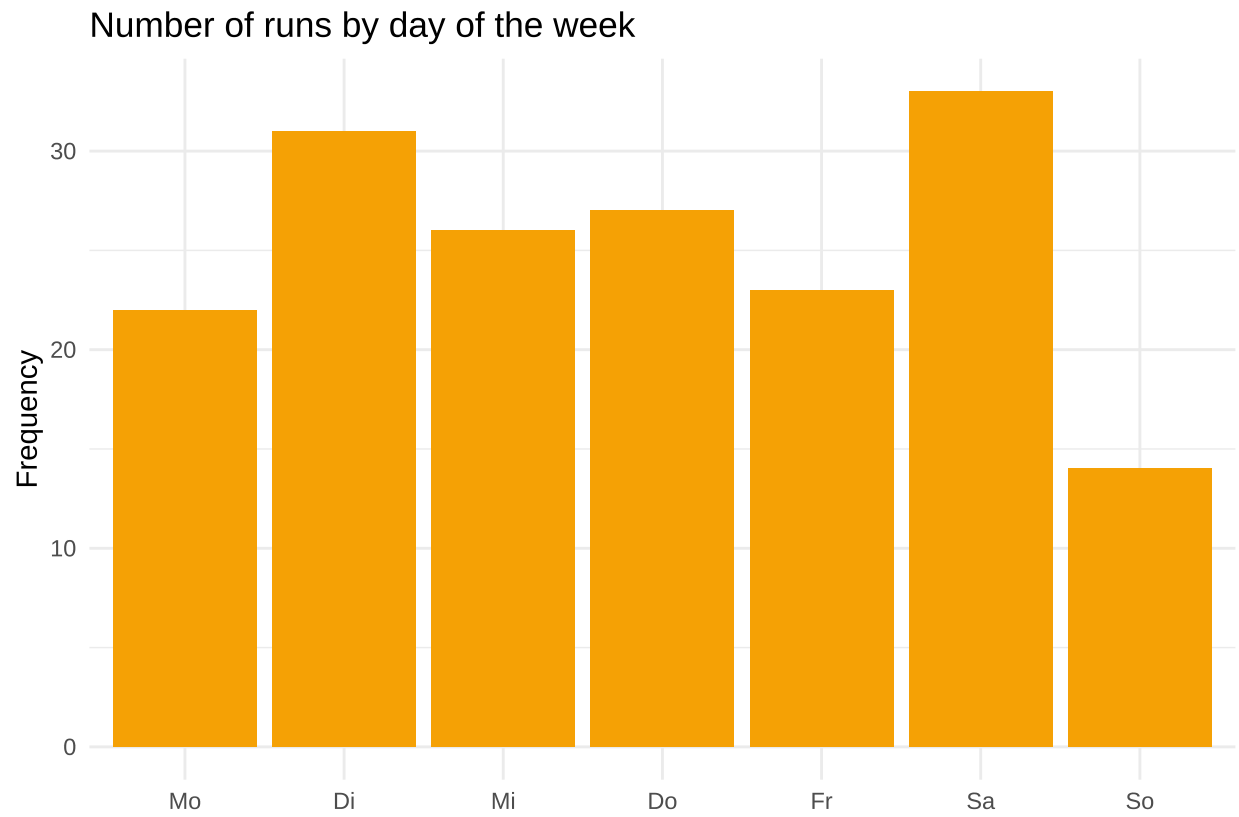




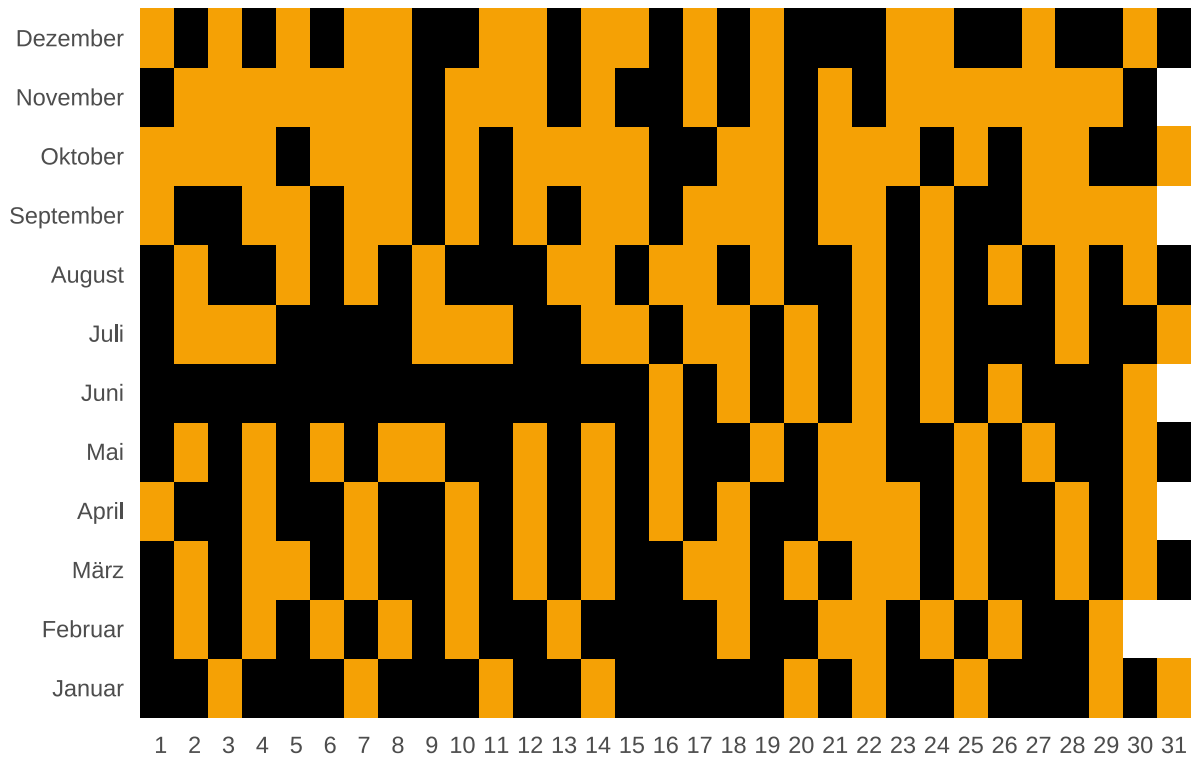
## Distribution of pace



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



# RUNRUNRUN



RUNRUNRUN

