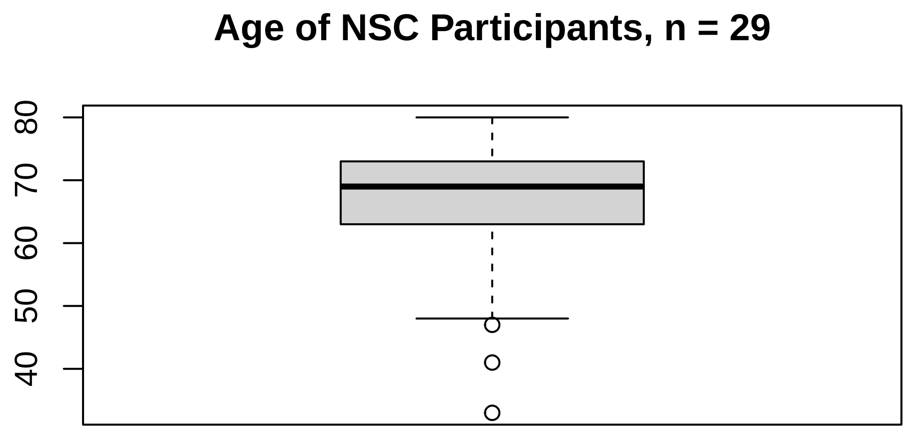
nsc_ortho

Making plots for patient characteristics

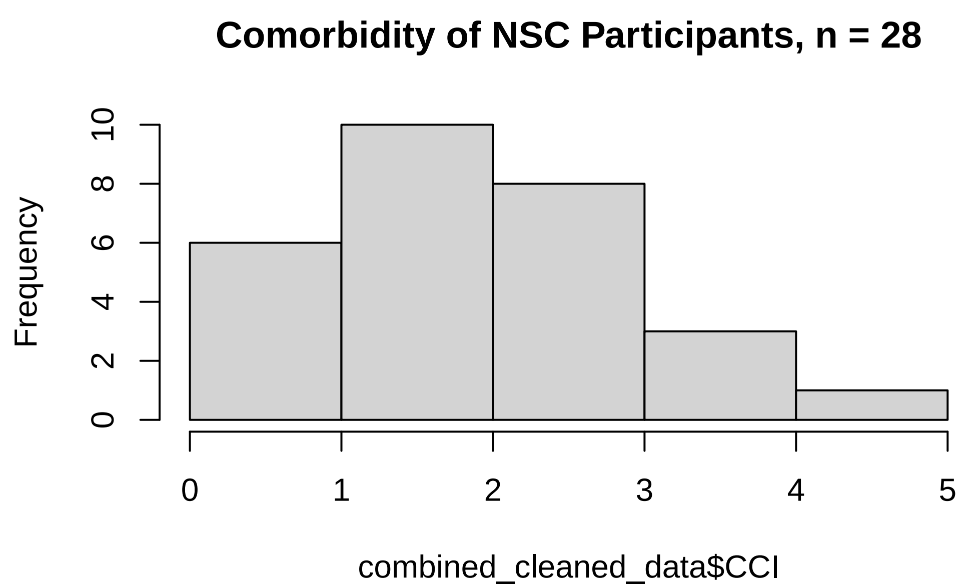
```
combined_cleaned_data = read.csv(paste0(wd, "data/Scoliosis_combined.csv")) %>%
  select(Age, Gender, Race, CCI) %>%
  filter(!is.na(Age))

table(combined_cleaned_data$Gender, combined_cleaned_data$Race)
```

```
                 Hispanic or Latino Mixed, more than 4 White
  Did not respond                  0                  0     1
  Female                           1                  1    25
  Male                             0                  0     1
```

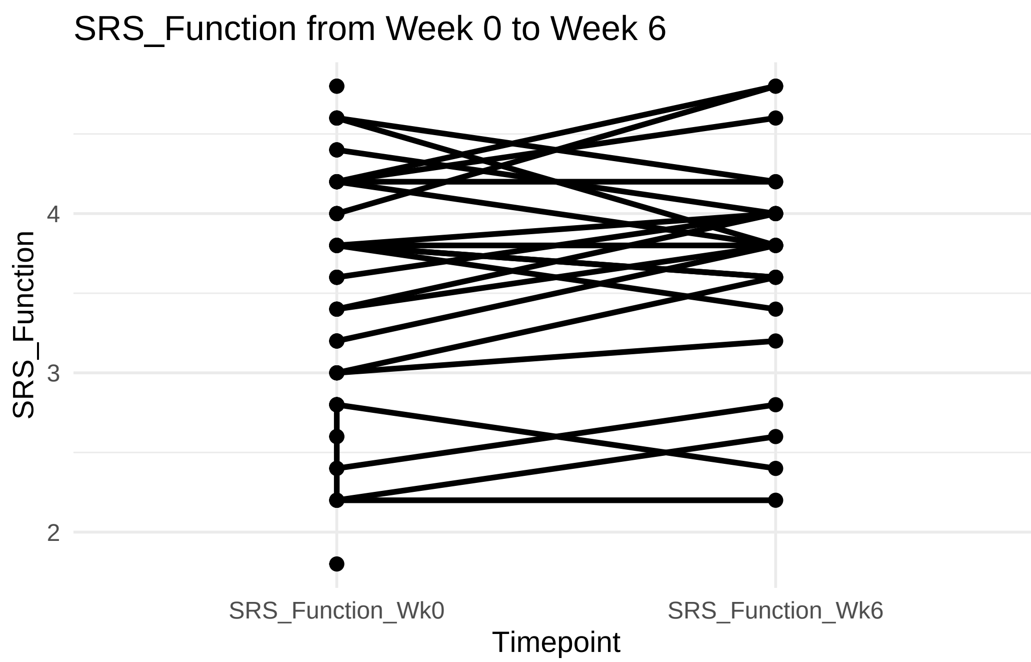
```
boxplot(combined_cleaned_data$Age, main = "Age of NSC Participants, n = 29")
```



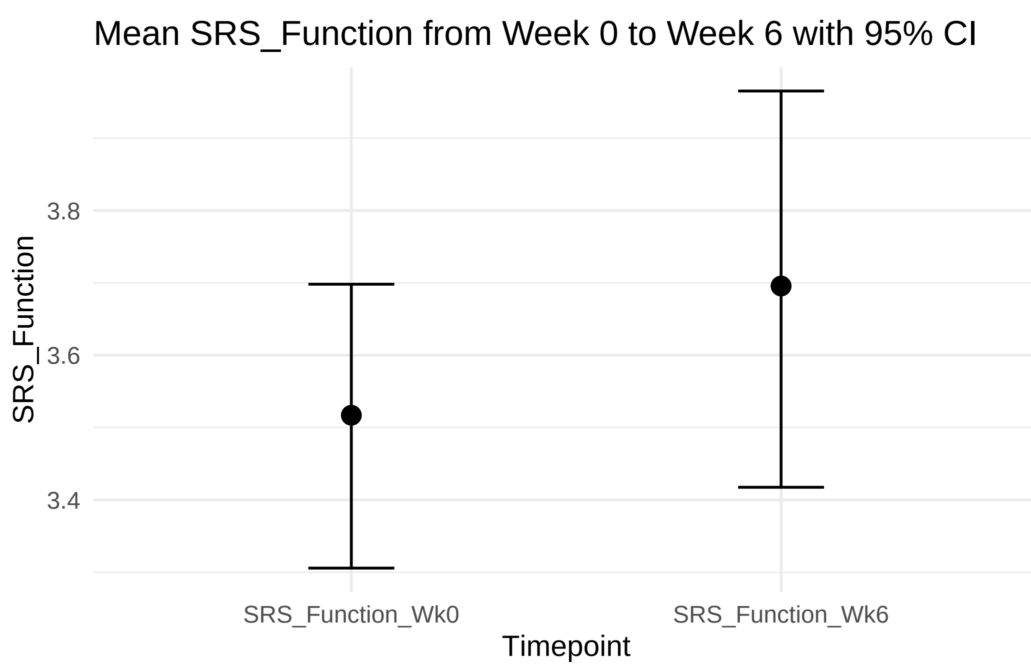

```r
hist(combined_cleaned_data$CCI, main = "Comorbidity of NSC Participants, n = 28")
```

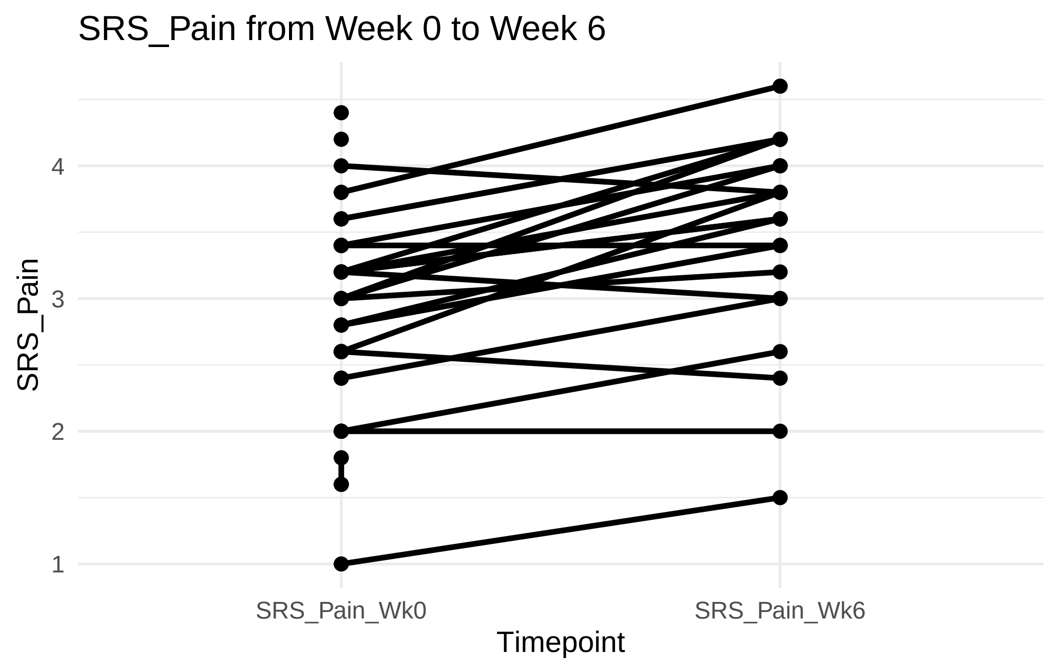


```r
meaningful_variables = c("SRS_Function", "SRS_Pain", "SRS_Image", "SRS_Mental", "SRS_Total")

for(select_variable in meaningful_variables){
  combined_cleaned_data = read.csv(paste0(wd, "data/Scoliosis_combined.csv")) %>%
    select(`Last.Name`, starts_with(select_variable)) %>%
    pivot_longer(cols = starts_with(select_variable), names_to = "timepoint", values_to = sel

  # Plot with individual participants
  p_individual = ggplot(combined_cleaned_data, aes(x = timepoint, y = .data[[select_variable
    geom_line(size = 1) +
    geom_point(size = 2) +
    labs(title = paste(select_variable, "from Week 0 to Week 6"),
         x = "Timepoint",
         y = select_variable) +
    theme_minimal() +
    theme(legend.position = "none")

  print(p_individual)

  # Plot with mean and confidence intervals
  p_summary = ggplot(combined_cleaned_data, aes(x = timepoint, y = .data[[select_variable]])
    stat_summary(fun.data = mean_cl_boot, geom = "errorbar", width = 0.2) +
    stat_summary(fun = mean, geom = "point", size = 3) +
    stat_summary(fun = mean, geom = "line", size = 1) +  # Line connecting the means
    labs(title = paste("Mean", select_variable, "from Week 0 to Week 6 with 95% CI"),
         x = "Timepoint",
         y = select_variable) +
    theme_minimal() +
    theme(legend.position = "none")
  print(p_summary)
}
```

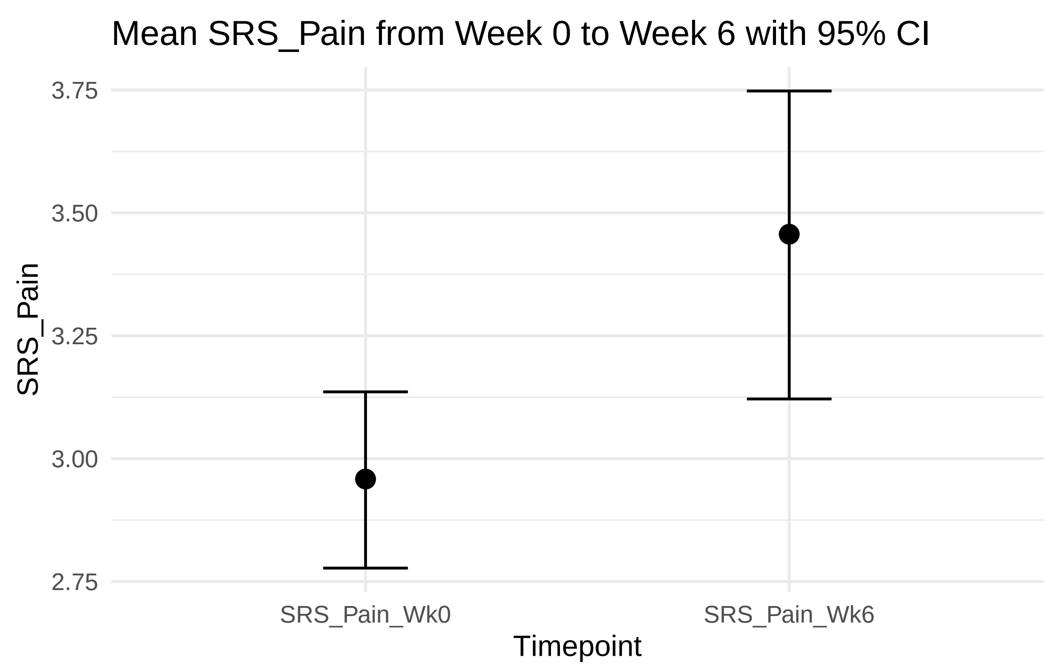


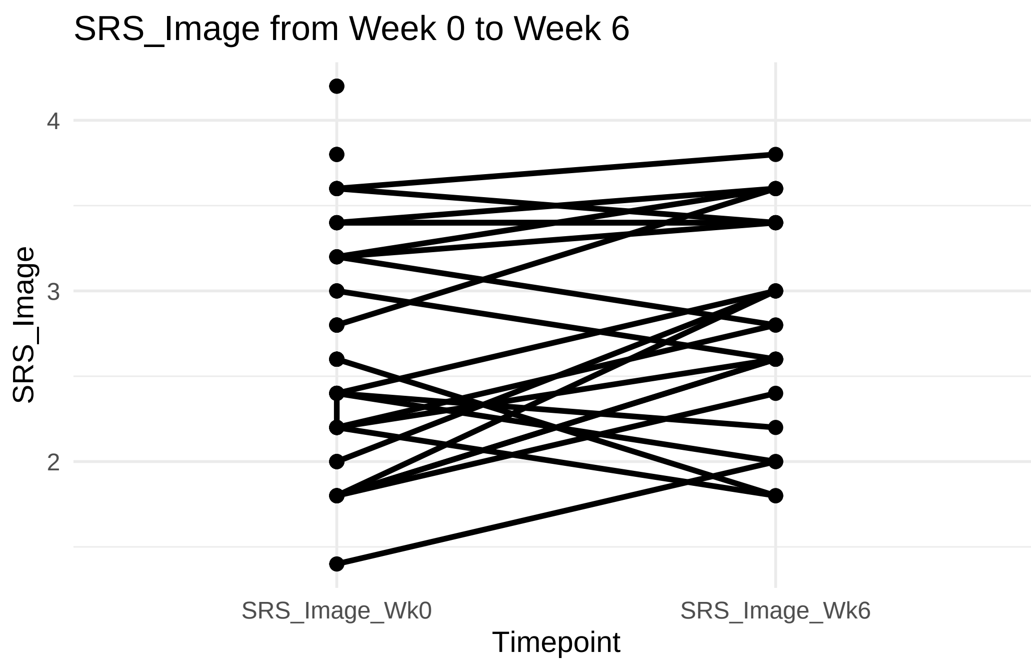

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```



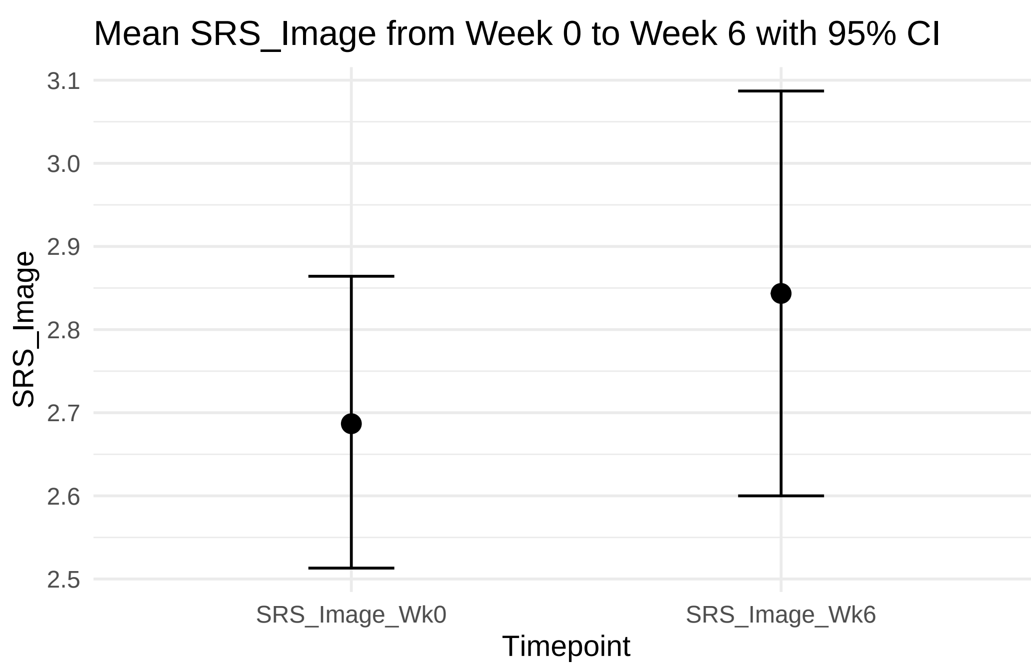


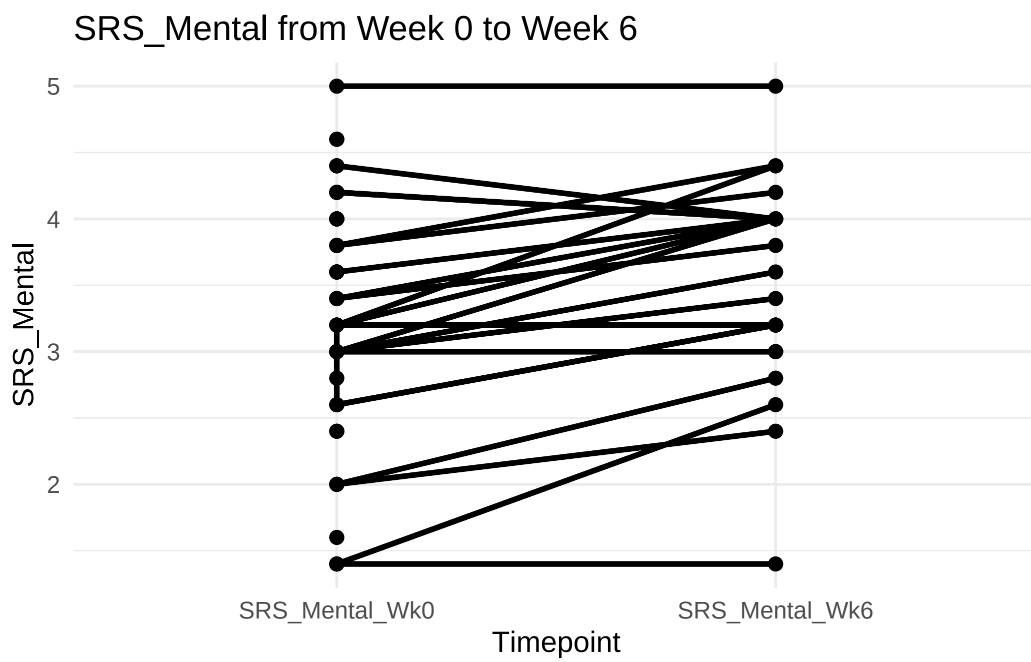

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```



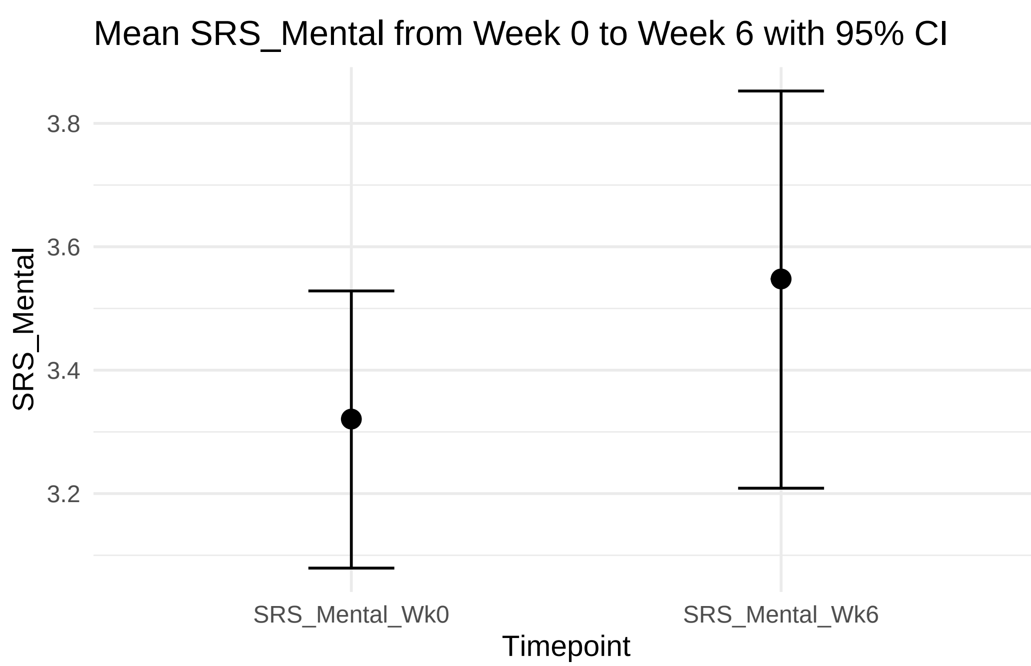


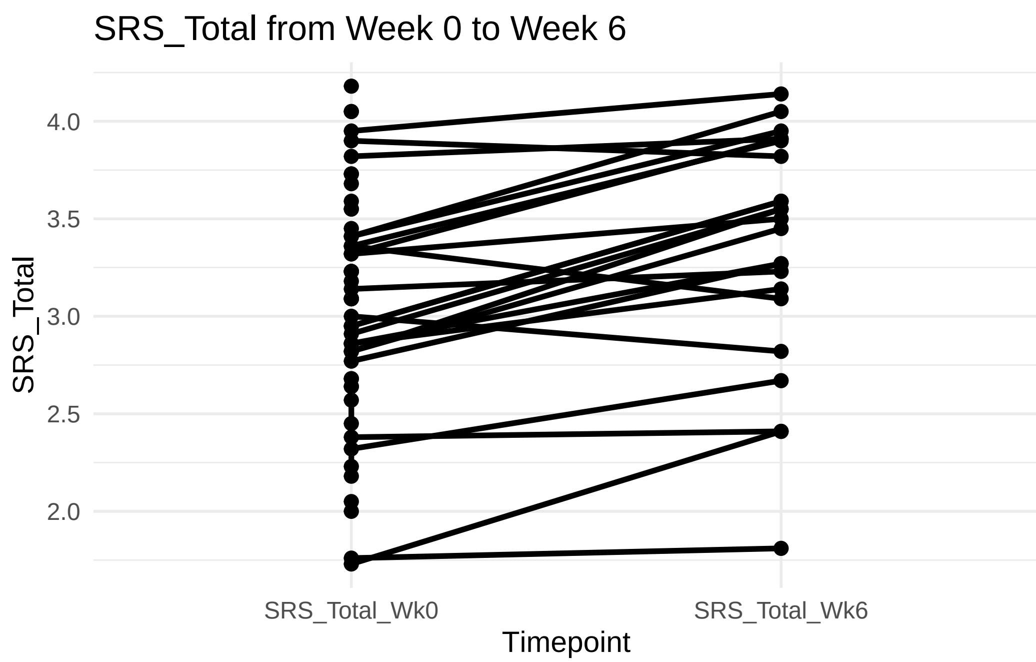

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```



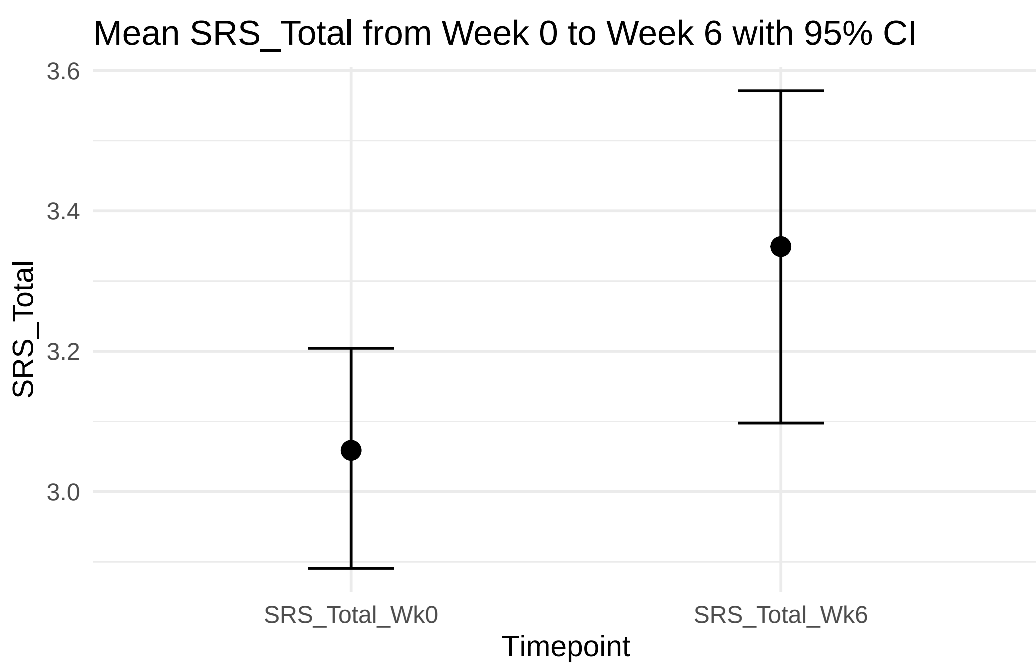



```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```

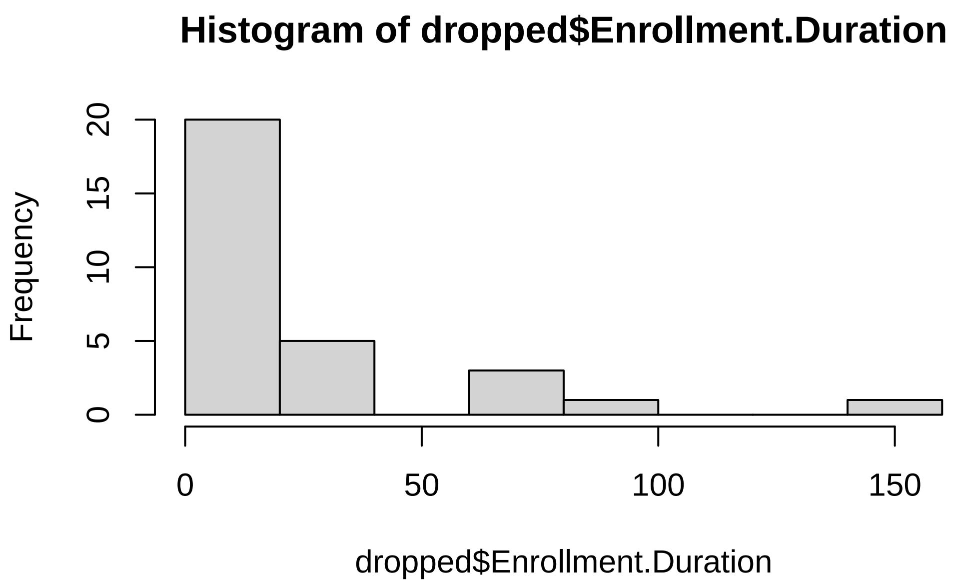




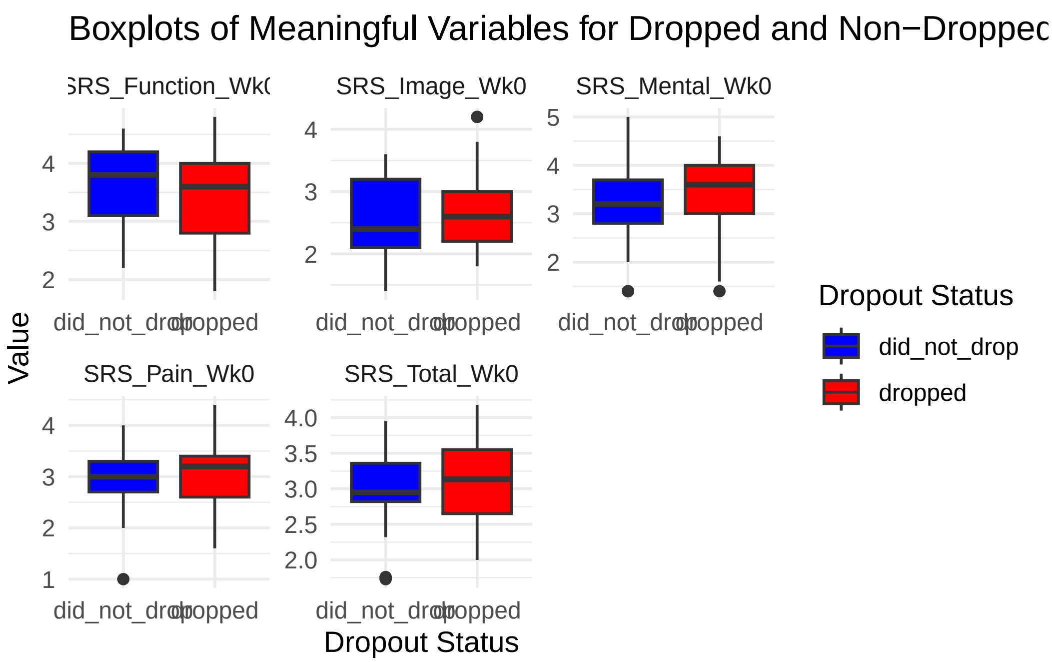

```
`geom_line()`: Each group consists of only one observation.
i Do you need to adjust the group aesthetic?
```



```r
finding_drop_outs = read.csv(paste0(wd, "data/Scoliosis_combined.csv")) %>%
  mutate(dropped = ifelse(is.na(SRS_Function_Wk6) & !is.na(SRS_Function_Wk0), "dropped", "did
dropped = finding_drop_outs %>% filter(dropped == "dropped")
hist(dropped$Enrollment.Duration)
```




```r
meaningful_var_wk0 = paste0(meaningful_variables, "_Wk0")
finding_drop_outs_long = finding_drop_outs %>%
  select(one_of(c(meaningful_var_wk0, "dropped"))) %>%
  pivot_longer(cols = -dropped, names_to = "Variable", values_to = "Value")

# Create boxplots with dropped and non-dropped participants
ggplot(finding_drop_outs_long, aes(x = dropped, y = Value, fill = dropped)) +
  geom_boxplot() +
  facet_wrap(~ Variable, scales = "free") +
  theme_minimal() +
  labs(title = "Boxplots of Meaningful Variables for Dropped and Non-Dropped Participants", x
  scale_fill_manual(values = c("dropped" = "red", "did_not_drop" = "blue"), name = "Dropout S
```



Performing stats tests for each of the comparisons

```
all_results_table = data.frame()
for(select_variable in meaningful_variables){
  combined_cleaned_data = read.csv(paste0(wd, "data/Scoliosis_combined.csv")) %>%
    select(`Last.Name`, starts_with(select_variable)) %>%
    pivot_longer(cols = starts_with(select_variable), names_to = "timepoint", values_to = sel
  formula_select = paste(select_variable, "~ timepoint + (1 | Last.Name)") %>% as.formula()
  mixed_model = lmer(formula_select, data = combined_cleaned_data)
  print(summary(mixed_model))
  results_frame = summary(mixed_model)[["coefficients"]] %>% as.data.frame() %>% filter(!row
  all_results_table = rbind(all_results_table, results_frame)
}
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: formula_select
   Data: combined_cleaned_data

REML criterion at convergence: 146.9

Scaled residuals:
    Min      1Q  Median      3Q     Max
```

```
-1.2789 -0.4661  0.0256  0.5028  1.6399

Random effects:
 Groups    Name        Variance Std.Dev.
 Last.Name (Intercept) 0.4446   0.6668
 Residual              0.1001   0.3165
Number of obs: 76, groups:  Last.Name, 52

Fixed effects:
                          Estimate Std. Error       df t value Pr(>|t|)
(Intercept)                3.53452    0.10225 55.24106  34.568   <2e-16 ***
timepointSRS_Function_Wk6  0.11479    0.09089 25.49150   1.263    0.218
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr)
tmpSRS_F_W6 -0.207
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: formula_select
   Data: combined_cleaned_data

REML criterion at convergence: 138.5

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.45079 -0.36617  0.04153  0.32510  1.43065

Random effects:
 Groups    Name        Variance Std.Dev.
 Last.Name (Intercept) 0.41763  0.6462
 Residual              0.08292  0.2880
Number of obs: 76, groups:  Last.Name, 52

Fixed effects:
                      Estimate Std. Error       df t value Pr(>|t|)
(Intercept)            2.98047    0.09803 55.39208  30.405  < 2e-16 ***
timepointSRS_Pain_Wk6  0.50329    0.08293 25.93226   6.069 2.08e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
```

```
            (Intr)
tmpSRS_P_W6 -0.196
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: formula_select
   Data: combined_cleaned_data

REML criterion at convergence: 141.7

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.66426 -0.57291  0.06158  0.48617  1.27462

Random effects:
 Groups    Name        Variance Std.Dev.
 Last.Name (Intercept) 0.3015   0.5491
 Residual              0.1384   0.3720
Number of obs: 76, groups:  Last.Name, 52

Fixed effects:
                       Estimate Std. Error       df t value Pr(>|t|)
(Intercept)             2.69282    0.09181 58.39296  29.330   <2e-16 ***
timepointSRS_Image_Wk6  0.21428    0.10477 27.30242   2.045   0.0506 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr)
tmpSRS_I_W6 -0.276
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: formula_select
   Data: combined_cleaned_data

REML criterion at convergence: 163.8

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.6398 -0.3996  0.1015  0.3795  1.3704

Random effects:
 Groups    Name        Variance Std.Dev.
 Last.Name (Intercept) 0.6416   0.8010
```

```
 Residual              0.1022   0.3197
Number of obs: 76, groups:  Last.Name, 52

Fixed effects:
                        Estimate Std. Error       df t value Pr(>|t|)
(Intercept)              3.32824    0.11951 54.03095  27.848  < 2e-16 ***
timepointSRS_Mental_Wk6  0.37521    0.09244 24.72128   4.059 0.000433 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr)
tmpSRS_M_W6 -0.178
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: formula_select
   Data: combined_cleaned_data

REML criterion at convergence: 108.1

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.36575 -0.40933  0.02681  0.45015  1.35699

Random effects:
 Groups    Name        Variance Std.Dev.
 Last.Name (Intercept) 0.29726  0.5452
 Residual              0.04932  0.2221
Number of obs: 76, groups:  Last.Name, 52

Fixed effects:
                       Estimate Std. Error       df t value Pr(>|t|)
(Intercept)             3.07037    0.08158 54.40432  37.637  < 2e-16 ***
timepointSRS_Total_Wk6  0.33464    0.06417 25.08335   5.215 2.12e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr)
tmpSRS_T_W6 -0.181
```

```r
all_results_table
```

```
                            Estimate Std. Error       df  t value     Pr(>|t|)
timepointSRS_Function_Wk6 0.1147869 0.09089167 25.49150 1.262898 2.180560e-01
timepointSRS_Pain_Wk6     0.5032884 0.08292518 25.93226 6.069186 2.082541e-06
timepointSRS_Image_Wk6    0.2142754 0.10476552 27.30242 2.045285 5.057363e-02
timepointSRS_Mental_Wk6   0.3752132 0.09244160 24.72128 4.058922 4.333132e-04
timepointSRS_Total_Wk6    0.3346418 0.06417148 25.08335 5.214806 2.119097e-05
```

The Minimum Clinically Important Difference in SRS-22R Total Score, Appearance, Activity and Pain Domains After Surgical Treatment of Adult Spinal Deformity

https://journals.lww.com/spinejournal/abstract/2015/03150/the_minimum_clinically_important_difference_
"Results: A total of 1321 patients were included in the analysis; 83% were females and 10% were smokers. Mean age was 53 years. Mean body mass index was 26.3 kg/m. Mean preoperative SRS-22R appearance score was 2.50 improving to 3.62 at 1 year postoperatively ($P < 0.001$). Mean preoperative SRS-22R activity score was 2.96 and it improved to 3.33 at 1 year postoperatively ($P < 0.001$). Mean preoperative SRS-22R pain score was 2.73 improving to 3.60 at 1 year postoperatively ($P < 0.001$). Mean preoperative total score was 2.93 and it improved to 3.65 at 1 year postoperatively ($P < 0.001$). There was a statistically significant difference in domain scores among the responses to the anchors ($P < 0.001$). The different calculation methods yielded MCID values of 0.19 to 1.23 for appearance, 0.23 to 0.60 for activity, 0.24 to 0.57 for pain, 0.16 to 0.43 for subscore, and 0.17 to 0.71 for total score."

```r
files = list.files(paste0(wd, "data"), full.names = TRUE)
files = files[grep("week", files)]

data_list = setNames(lapply(files, function(x) read.csv(x, check.names = FALSE)), basename(fi
# combined_data = purrr::reduce(data_list, full_join, by = c("First Name", "Last Name"))
```

First starting with the effectiveness of the actual intervention in terms of metrics

```r
select_starting_variables = function(input_df){
  org_df = input_df %>%
    mutate(`First Name` = str_trim(`First Name`),
           `Last Name` = str_trim(`Last Name`)) %>%
    mutate(name = interaction(`First Name`, `Last Name`)) %>%
    rename_with(~ "pain_score", contains("pain level")) %>%
    rename(times_last_wk = `How many times did you perform the Scoliosis Realignment Therapy
    mutate(times_last_wk = as.numeric(times_last_wk)) %>%
    select(name, pain_score, times_last_wk)
```

```r
  return(org_df)
}
wk_0_values = select_starting_variables(data_list[["week0.csv"]])
wk_6_values = select_starting_variables(data_list[["week6.csv"]])
wk_0_values$timepoint = "wk0"; wk_6_values$timepoint = "wk6"

eval_pain_improve = full_join(wk_0_values, wk_6_values, by = "name", suffix = c("wk0", "wk6"
head(eval_pain_improve)
```

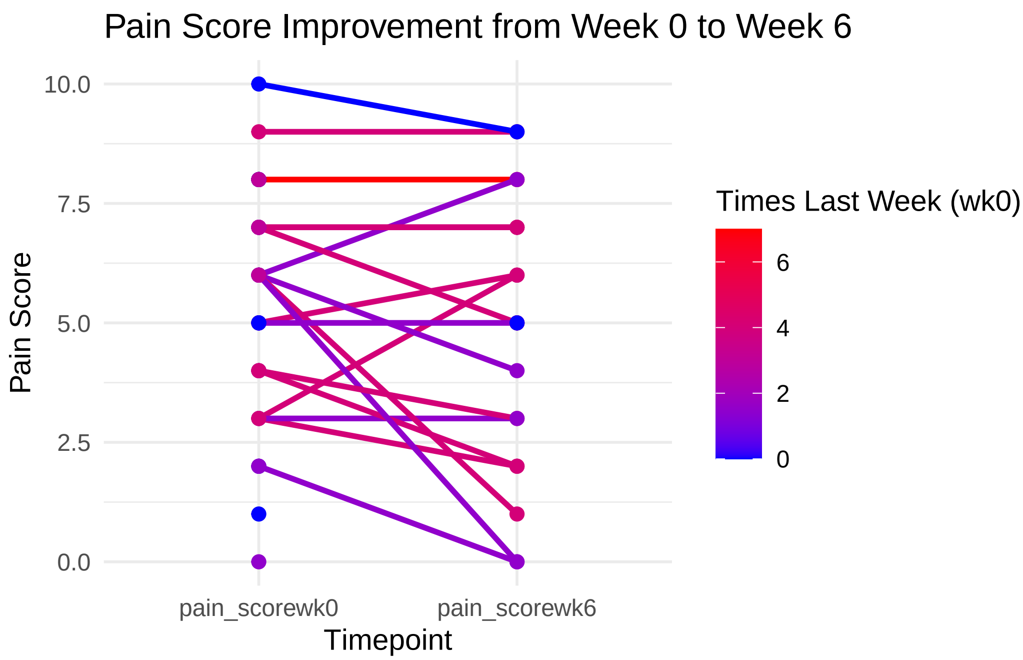
```
                  name pain_scorewk0 times_last_wkwk0 timepointwk0
1        Lu Ann.Blough             5                4          wk0
2         Sandy.Rausch             7                4          wk0
3          Diana.Potts             7                4          wk0
4        Mary.Patrenos             4                4          wk0
5 Harmony (Mary).Joyce             5                4          wk0
6        Aliya.Kuerban             6                0          wk0
  pain_scorewk6 times_last_wkwk6 timepointwk6
1             6                3          wk6
2            NA               NA         <NA>
3             5                3          wk6
4             2                3          wk6
5            NA               NA         <NA>
6            NA               NA         <NA>
```

We will need some manual review in case people are inputting their names inconsistently (e.g. with typos)

```r
long_data = eval_pain_improve %>%
  pivot_longer(cols = starts_with("pain_score"), names_to = "timepoint", values_to = "pain_sc
  mutate(timepoint = recode(timepoint, "pain_score_wk0" = "Week 0", "pain_score_wk6" = "Week

# Create the plot
ggplot(long_data, aes(x = timepoint, y = pain_score, group = name)) +
  geom_line(aes(color = times_last_wkwk0), size = 1) + # Color lines by `times_last_wk` at we
  geom_point(aes(color = times_last_wkwk0), size = 2) + # Color points by `times_last_wk` at
  scale_color_gradient(low = "blue", high = "red") + # Customize the color scale
  labs(title = "Pain Score Improvement from Week 0 to Week 6",
       x = "Timepoint",
       y = "Pain Score",
       color = "Times Last Week (wk0)") +
  theme_minimal()
```

```
Warning: Removed 24 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
Warning: Removed 24 rows containing missing values or values outside the scale range
(`geom_point()`).
```



now we actually do some stats on the dataset, starting with complete cases

```r
pain_complete_cases = full_join(wk_0_values, wk_6_values, by = "name", suffix = c("wk0", "wk6
  filter(!is.na(pain_scorewk0) & !is.na(pain_scorewk6))
dim(pain_complete_cases)
```

```
[1] 19  7
```

```r
t_test_result = t.test(pain_complete_cases$pain_scorewk0, pain_complete_cases$pain_scorewk6,
t_test_result
```

```
	Paired t-test

data:  pain_complete_cases$pain_scorewk0 and pain_complete_cases$pain_scorewk6
```

```
t = 1.7354, df = 18, p-value = 0.09975
alternative hypothesis: true mean difference is not equal to 0
95 percent confidence interval:
 -0.1773444  1.8615550
sample estimates:
mean difference
      0.8421053
```

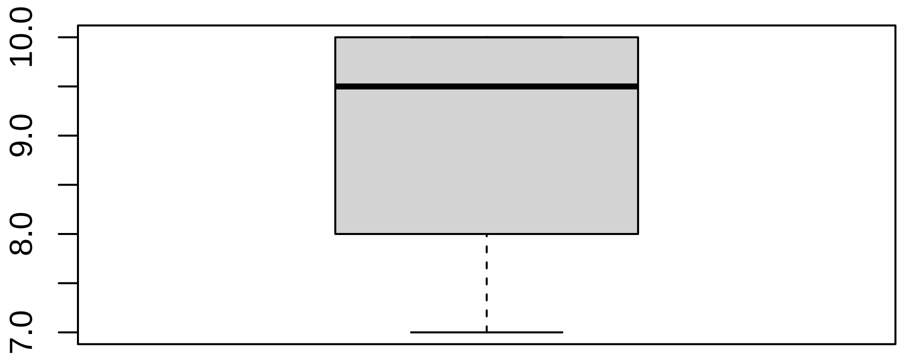
```r
stats_pain_improve = rbind(wk_0_values, wk_6_values)
mixed_model = lmer(pain_score ~ timepoint + (1 | name), data = stats_pain_improve)
summary(mixed_model)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: pain_score ~ timepoint + (1 | name)
   Data: stats_pain_improve

REML criterion at convergence: 275.6

Scaled residuals:
     Min       1Q   Median       3Q      Max
-2.08803 -0.38506  0.05652  0.53855  1.42569

Random effects:
 Groups   Name        Variance Std.Dev.
 name     (Intercept) 3.884    1.971
 Residual             2.177    1.476
Number of obs: 62, groups:  name, 43

Fixed effects:
             Estimate Std. Error      df t value Pr(>|t|)
(Intercept)    5.6396     0.3809 51.5967  14.805   <2e-16 ***
timepointwk6  -0.8152     0.4471 24.7687  -1.823   0.0803 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr)
timepontwk6 -0.338
```

```r
wk_8_satisfaction_scores = data_list[["week8.csv"]]
table(wk_8_satisfaction_scores$`How would you feel if you could no longer use National Scolic
```

```
Somewhat disappointed     Very disappointed
                    3                     8
```

```r
boxplot(wk_8_satisfaction_scores$`On a scale of 1-10, how likely are you to recommend the Nat
```



```r
extract_numeric = function(text) {
  cleaned_text = str_replace_all(text, "[^\\d\\.]", "") # Remove non-numeric characters excep
  as.numeric(cleaned_text)
}

wk_2_costs = data_list[["week2.csv"]] %>%
  select(Timestamp, `Join Date`, `In the past 3 months, how much did you spend on healthcare
  rename(cost_3_months = `In the past 3 months, how much did you spend on healthcare for your
  filter(nchar(Timestamp) > 1 & nchar(`Join Date`) > 1) %>%
  mutate(Timestamp = mdy_hms(Timestamp, truncated = Inf),
         `Join Date` = mdy(`Join Date`)) %>%
  mutate(Time_Difference = as.numeric(difftime(Timestamp, `Join Date`, units = "days"))) %>%
```

```r
  mutate(cost_3_months = extract_numeric(cost_3_months)) %>%
  filter(!is.na(cost_3_months))

head(wk_2_costs)
```

```
            Timestamp  Join Date cost_3_months Time_Difference
1 2024-04-26 06:27:00 2024-02-24             0        62.26875
2 2024-04-26 06:40:00 2024-03-06           120        51.27778
3 2024-04-26 09:15:00 2024-04-05           100        21.38542
4 2024-04-26 11:19:00 2024-02-07           900        79.47153
5 2024-04-26 12:15:00 2024-01-31          5000        86.51042
6 2024-04-26 15:43:00 2023-11-15             0       163.65486
```

```r
mean(wk_2_costs$cost_3_months)
```

```
[1] 576.8333
```

```r
wk_2_recent = wk_2_costs %>% filter(Time_Difference < 90)
mean(wk_2_recent$cost_3_months)
```

```
[1] 626.9048
```