Wilcoxon Signed Rank Test

Joseph Oliveira

7/4/2020

Loading data in from the `Wilcoxon_Test.Rmd` cleaning procedure

```
faces <- faces %>%
  group_by(Group, `Participant #`, Time, Survey) %>%
  summarise(avg_resp = mean(Response)) %>%
  ungroup()
```

Wilcoxon Test function

1. Cleaned `faces` data is filtered for the Time (`Pre` or `Post`) and Group (`Experimental` or `Control`). Depends one what is being compared
2. Two data sets are created to create the vectors for comparison
3. 3 Cases need to be handled a. If the vector of response variables in data set A are longer than B b. If the vector of response variables in data set B are longer than A c. If the vector of response variables in data set A and B are equal length

```
faces_wilcox <- function(time1, group1, time2, group2, survey) {
  # Filter to data we need for comparison
  faces_ <- faces %>%
    filter(Time %in% c(time1, time2) & Group %in% c(group1, group2))

  # Create 2 datasets
  comp1 <- faces_ %>%
    filter(Time == time1, Group == group1, Survey == survey)

  comp2 <- faces_ %>%
    filter(Time == time2, Group == group2, Survey == survey)

  # 3 Cases need to be handled:
  # 1. If the lengths are unequal. Sample longer sample to obtain equal comparison
  if (length(comp1$avg_resp) > length(comp2$avg_resp)) {

    comp1_spl_resp <- sample(comp1$avg_resp, replace = T, size = length(comp2$avg_resp))

    wilcox.test(comp1_spl_resp, comp2$avg_resp)

    # Second case for unequal lengths
    } else if (length(comp1$avg_resp) < length(comp2$avg_resp)) {
```

```
    comp2_spl_resp <- sample(comp2$avg_resp, replace = T, size = length(comp1$avg_resp))

    wilcox.test(comp1$avg_resp, comp2_spl_resp)

    # if they are equal, then run comparison.
    } else wilcox.test(comp1$avg_resp, comp2$avg_resp)
  }
```

Creating the comparison groups for the function above.

```
distinct_groupings <- faces %>%
  distinct(Group, Time, Survey)

dist_grp_exp_pre <- distinct_groupings %>%
  filter(Group == 'Experimental', Time == 'Pre')

dist_grp_ctrl_pre <- distinct_groupings %>%
  filter(Group == 'Control', Time == 'Pre')

dist_grp_exp_post <- distinct_groupings %>%
  filter(Group == 'Experimental', Time == 'Post')

dist_grp_ctrl_post <- distinct_groupings %>%
  filter(Group == 'Control', Time == 'Post')

first_comp <- inner_join(dist_grp_exp_pre, dist_grp_ctrl_pre, by = c("Survey", "Time"))
secnd_comp <- inner_join(dist_grp_exp_post, dist_grp_ctrl_post, by = c("Survey", "Time"))
third_comp <- inner_join(dist_grp_exp_pre, dist_grp_exp_post, by = c("Survey", "Group"))
forth_comp <- inner_join(dist_grp_ctrl_pre, dist_grp_ctrl_post, by = c("Survey", "Group"))

group_comparison <- bind_rows(first_comp, secnd_comp) %>% mutate_all(as.character)

time_comparison <- bind_rows(third_comp, forth_comp) %>% mutate_all(as.character)
```

Running Wilcoxon tests

```
group_wilcoxon <- group_comparison %>%
  mutate(test = pmap(list(time1 = Time, group1 = Group.x,
                          time2 = Time, group2 = Group.y,
                          survey = Survey),
                     faces_wilcox),
         p.value = unlist(map(test, function(x) `$`(`[`(`[`(x)), 'p.value'))))

time_wilcoxon  <- time_comparison %>%
  mutate(test = pmap(list(time1 = Time.x, group1 = Group,
                          time2 = Time.y, group2 = Group,
                          survey = Survey),
                     faces_wilcox),
         p.value = unlist(map(test, function(x) `$`(`[`(`[`(x)), 'p.value'))))
```

```r
knitr::kable(select(group_wilcoxon, -test) %>%
               group_by(Survey) %>%
               arrange(Survey, p.value, Time))
```

Group.x	Time	Survey	Group.y	p.value
Experimental	Post	AKS	Control	0.2911040
Experimental	Pre	AKS	Control	0.3574898
Experimental	Post	FACES	Control	0.5745038
Experimental	Pre	FACES	Control	1.0000000
Experimental	Post	FES	Control	0.2289507
Experimental	Pre	FES	Control	0.9360747
Experimental	Post	FPPS	Control	0.1142857
Experimental	Pre	FPPS	Control	0.3094241
Experimental	Pre	SCS	Control	0.0571429
Experimental	Post	SCS	Control	0.5516159
Experimental	Post	SEAS	Control	0.1986078
Experimental	Pre	SEAS	Control	0.9356223

```r
knitr::kable(select(time_wilcoxon, -test) %>%
               group_by(Survey) %>%
               arrange(Survey, p.value, Group))
```

Group	Time.x	Survey	Time.y	p.value
Experimental	Pre	AKS	Post	0.2886460
Control	Pre	AKS	Post	0.6235742
Experimental	Pre	FACES	Post	0.0567903
Control	Pre	FACES	Post	0.9357363
Experimental	Pre	FES	Post	0.0311456
Control	Pre	FES	Post	0.7479209
Experimental	Pre	FPPS	Post	0.7117698
Control	Pre	FPPS	Post	0.7715034
Experimental	Pre	SCS	Post	0.2052307
Control	Pre	SCS	Post	0.3428571
Experimental	Pre	SEAS	Post	0.1007655
Control	Pre	SEAS	Post	0.8095268