Additional analyses

Group A

11/30/2019

Load the relevant libraries and datasets

```
library(tidyverse)
library(lubridate)
library(magrittr)
library(boot)
library(fastDummies)
library(GGally)
library(sjPlot)
library(radiant.data)
library(lsmeans)

# read in the data
tDataEx2 <- readRDS('Data/data_ex2_20170101-20170110.rds')
tDataEx3 <- readRDS('Data/data_ex3_20170704-20170710.rds')
tDataEx4 <- readRDS('Data/data_ex4_20170711-20170713.rds')
```

Authors' original code

All of the tables and plots associated with the article can be easily reproduced from the researchers' code and data below. They used concise R code. It is likely that they did exploratory analysis on their data before settling on the plots that they chose since their plots are detailed. They also did not include any plots for study one and they likely did make some plots for themselves during their research process. The two main plots that they chose are comprehensive summaries of their findings for study two and study three.

```
# function to bootstrap-calc 95% CI and return as data frame (ready for dplyr)
fCalcBootstrapCI <- function(.tData) {
  b <- boot(.tData, function(.tDataSub, .i) sum(.tDataSub$bHelplineShown[.i])/sum(.tDataSub$bResultsRea
  ci <- boot.ci(b, type='perc')
  return(data.frame(
    n = sum(.tData$bResultsReached),
    helpline = 100*sum(.tData$bHelplineShown)/sum(.tData$bResultsReached),
    ci_l = 100*ci$percent[4],
    ci_u = 100*ci$percent[5]
  ))
}


# study 1: comparison between Germany and the US
tDataEx2 %>%
  group_by(eLocation) %>%
  summarise(n=n())

tDataEx2 %>%
  mutate(Group=factor(eType,
                      levels=c('helpful', 'harmful', 'unrelated'),
                      labels=c('helpful', 'harmful', 'unrelated'))) %>%
  group_by(eLocation, dDay, Group) %>%
  do(fCalcBootstrapCI(.)) %>%
  ggplot(aes(dDay, helpline, group=Group, linetype=Group, shape=Group, fill=Group, color=Group)) +
  geom_ribbon(aes(ymin=ci_l, ymax=ci_u), fill='grey90', color=NA) +
  geom_line(size=.7) +
  geom_point(size=3.3) +
  facet_grid(eLocation ~ .) +
  scale_x_date(date_labels='%m/%d', date_breaks='1 day', name=' \nDate') +
  scale_y_continuous(limits=c(0, 100), breaks=seq(0, 100, 10), name='Share of results displaying the su
  theme_bw() +
  theme(legend.position = 'bottom') +
  scale_colour_brewer(palette='Paired')
# ggsave('ex2_percentage-over-time_countrywise.png', device='png', width=10)

# study 2: comparison across the globe (main language only)
tDataEx3 %>% nrow
tDataEx3 %>%
  group_by(dDay = as.Date(dCreate), eGroup) %>%
  do(fCalcBootstrapCI(.)) %>%
  ggplot(aes(dDay, helpline, color=eGroup)) +
    geom_line() +
    geom_point() +
    geom_ribbon(aes(ymin=ci_l, ymax=ci_u), alpha=.2, color=NA)

fCalcBootstrapCI(tDataEx3 %>% filter(eGroup == 'harmful'))
fCalcBootstrapCI(tDataEx3 %>% filter(eGroup == 'helpful'))

tDataEx3 %>%
  group_by(eCountry, eGroup) %>%
  do(fCalcBootstrapCI(.)) %>%
  arrange(eGroup, desc(relative)) %>%
  View

tDataEx3 %>%
  filter(eGroup != 'control') %>%
  group_by(eCountry, eGroup) %>%
  do(fCalcBootstrapCI(.)) %>%
  mutate(
    Group = factor(eGroup,
                   levels=c('helpful', 'harmful'),
                   labels=c('helpful', 'harmful')),
    Country = factor(eCountry,
                     levels=c('China', 'Canada', 'India', 'South Korea', 'Brazil', 'Germany', 'Japan',
                     labels=c('Singapore\n(Mandarin)', 'Canada\n(English)', 'India\n(Hindi)', 'South Ko
  ) %>%
  ggplot(aes(Country, helpline, fill=Group)) +
    geom_bar(stat='identity', position=position_dodge(), width=.80) + #geom_bar doppelt, um Reihenfolge
    geom_tile(aes(x='Ireland\n(English)',    y=11/2,   width=.98, height=11, fill='suicide rate per 100
    geom_tile(aes(x='Australia\n(English)',  y=10.6/2, width=.98, height=10.6, fill='suicide rate per 10
    geom_tile(aes(x='UK\n(English)',         y=6.2/2,  width=.98, height=6.2, fill='suicide rate per 100
    geom_tile(aes(x='USA\n(English)',        y=12.1/2, width=.98, height=12.1, fill='suicide rate per 10
    geom_tile(aes(x='Japan\n(Japanese)',     y=18.5/2, width=.98, height=18.5, fill='suicide rate per 10
```

```r
    geom_tile(aes(x='Germany\n(German)',     y=9.2/2,  width=.98, height=9.2, fill='suicide rate per 10
    geom_tile(aes(x='Brazil\n(Portuguese)',  y=5.8/2,  width=.98, height=5.8, fill='suicide rate per 10
    geom_tile(aes(x='South Korea\n(Korean)', y=28.9/2, width=.98, height=28.9, fill='suicide rate per 1
    geom_tile(aes(x='India\n(Hindi)',        y=21.1/2, width=.98, height=21.1, fill='suicide rate per 1
    geom_tile(aes(x='Canada\n(English)',     y=9.8/2,  width=.98, height=9.8, fill='suicide rate per 10
    geom_tile(aes(x='Singapore\n(Mandarin)', y=7.4/2,  width=.98, height=7.4, fill='suicide rate per 10
    geom_bar(stat='identity', position=position_dodge(), width=.80) +
    geom_errorbar(aes(ymin=ci_l, ymax=ci_u), width=.5, position=position_dodge(.9)) +
    scale_x_discrete() +
    scale_y_continuous(limits=c(0, 100), breaks=seq(0, 100, 10), name=' \nShare of results displaying t
    coord_flip() +
    theme_bw() +
    theme(legend.position = 'bottom') +
    #scale_fill_brewer(palette='Paired')
    scale_fill_manual(values=c('#1f78b4', '#a6cee3', 'grey80'))

# ggsave('ex3_percentage-per-country-and-group.png', device='png', width=10)

# study 3: comparison within multi-lingual country (india, china)
tDataEx4 %>% nrow
tDataEx4 %>%
  group_by(eCountry, eLanguage, eGroup) %>%
  do(fCalcBootstrapCI(.)) %>%
  View

tDataEx4 %>%
  mutate(
    dDay = as.Date(dCreate),
    eCountry = factor(eCountry,
                      levels=c('China', 'India'),
                      labels=c('Singapore', 'India'))) %>%
  group_by(eCountry, eLanguage, eGroup, dDay) %>%
  do(fCalcBootstrapCI(.)) %>%
  mutate(
    Language = eLanguage,
    Group = factor(eGroup,
                   levels=c('harmful', 'helpful', 'control'),
                   labels=c('harmful', 'helpful', 'unrelated'))
  ) %>%
  ggplot(aes(dDay, helpline, group=Language, linetype=Language, shape=Language, fill=Language, color=La
    geom_ribbon(aes(ymin=ci_l, ymax=ci_u), fill='grey90', color=NA) +
    geom_line(size=.7) +
    geom_point(size=3.3) +
    facet_grid(eCountry ~ Group) +
    scale_x_date(date_labels='%m/%d', date_breaks='1 day', name='Date') +
    scale_y_continuous(limits=c(0, 100), breaks=seq(0, 100, 10), name='Share of results displaying the
    theme_bw() +
    theme(legend.position = 'bottom') +
    scale_colour_brewer(palette='Paired')
# ggsave('ex4_percentage-over-time_languagewise.png', device='png', width=10)
```

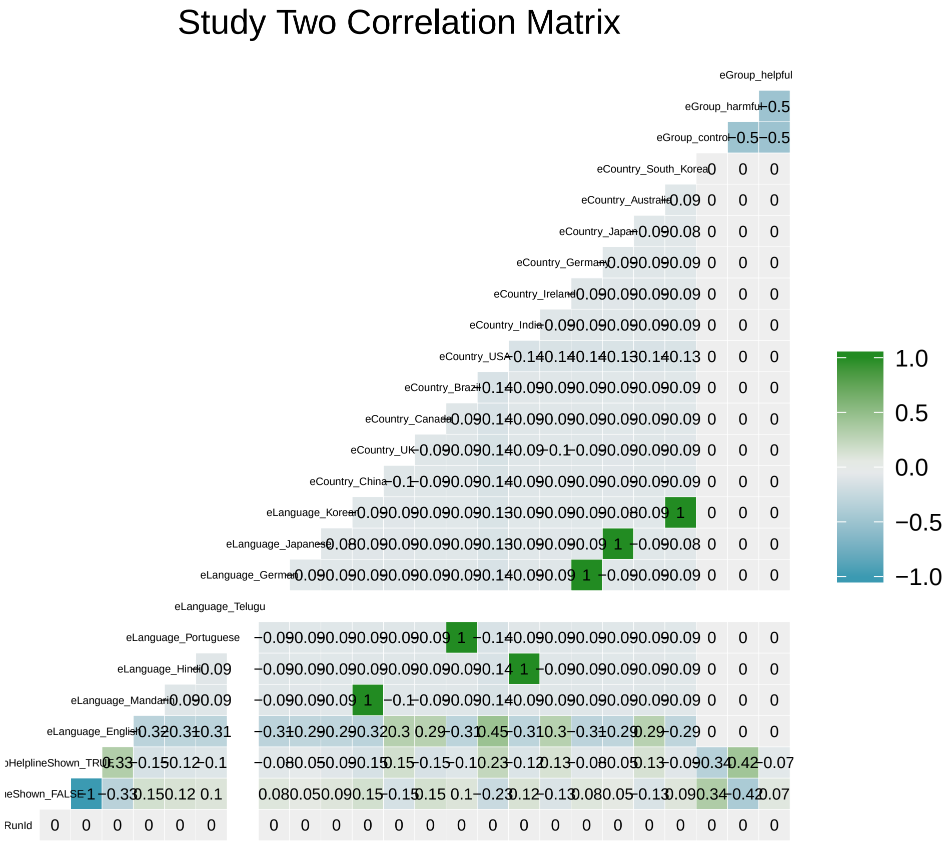
Our additional analyses of the same data

See the paper for descriptions of the output of this code.

Correlation Plot

What are the correlations between the variables in Study 2?

```
study2dummyRef <- dummy_cols(tDataEx3, select_columns = c("bHelplineShown", "eLanguage", "eCountry", "e(
ggcorr(study2dummyRef, hjust = 0.75, size = 1.5, label=T, label_round = 2,
  label_size = 2, high = "forestgreen") +
  ggplot2::labs(title = "Study Two Correlation Matrix") +
  theme(plot.title = element_text(hjust = 0.5))
```

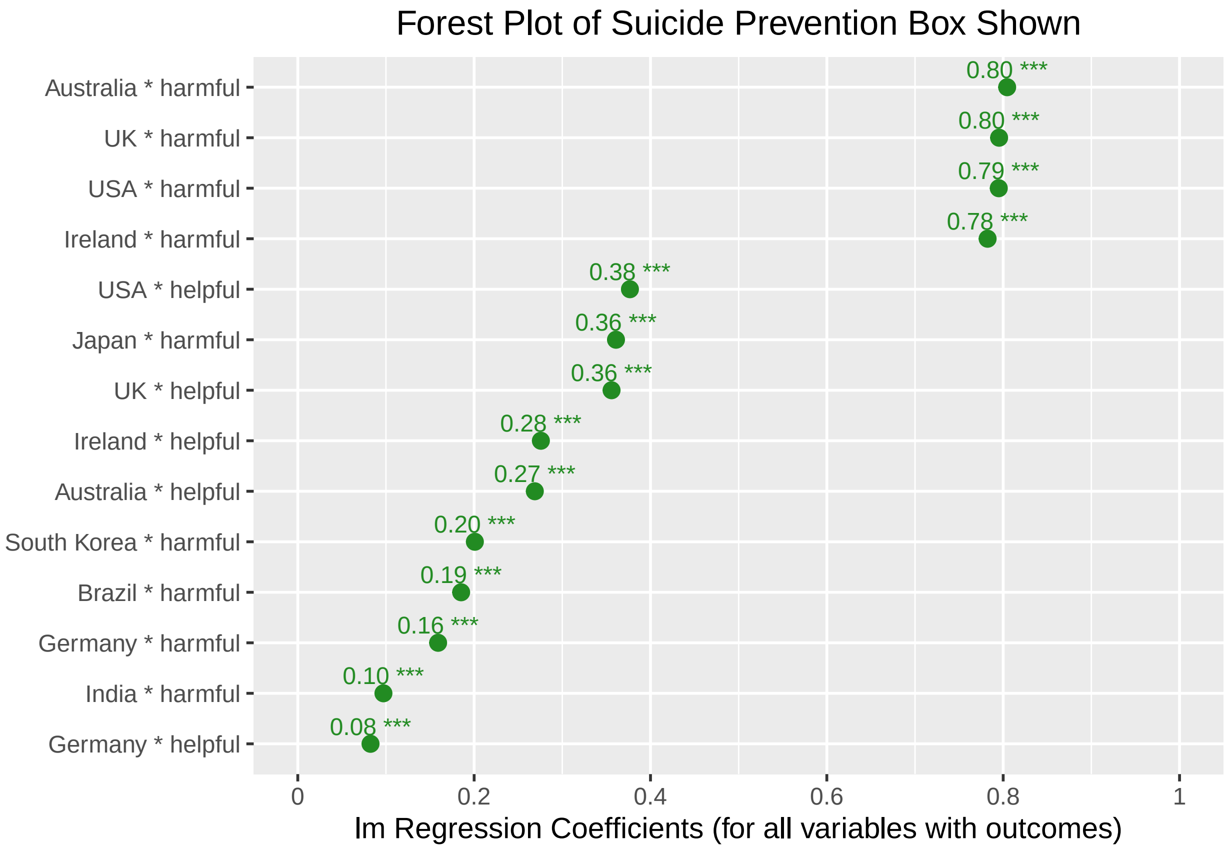


Forest Plot for the Linear Model Coefficients

How do the linear coefficients compare accross a linear model of the data with an interaction around country and phrase type?

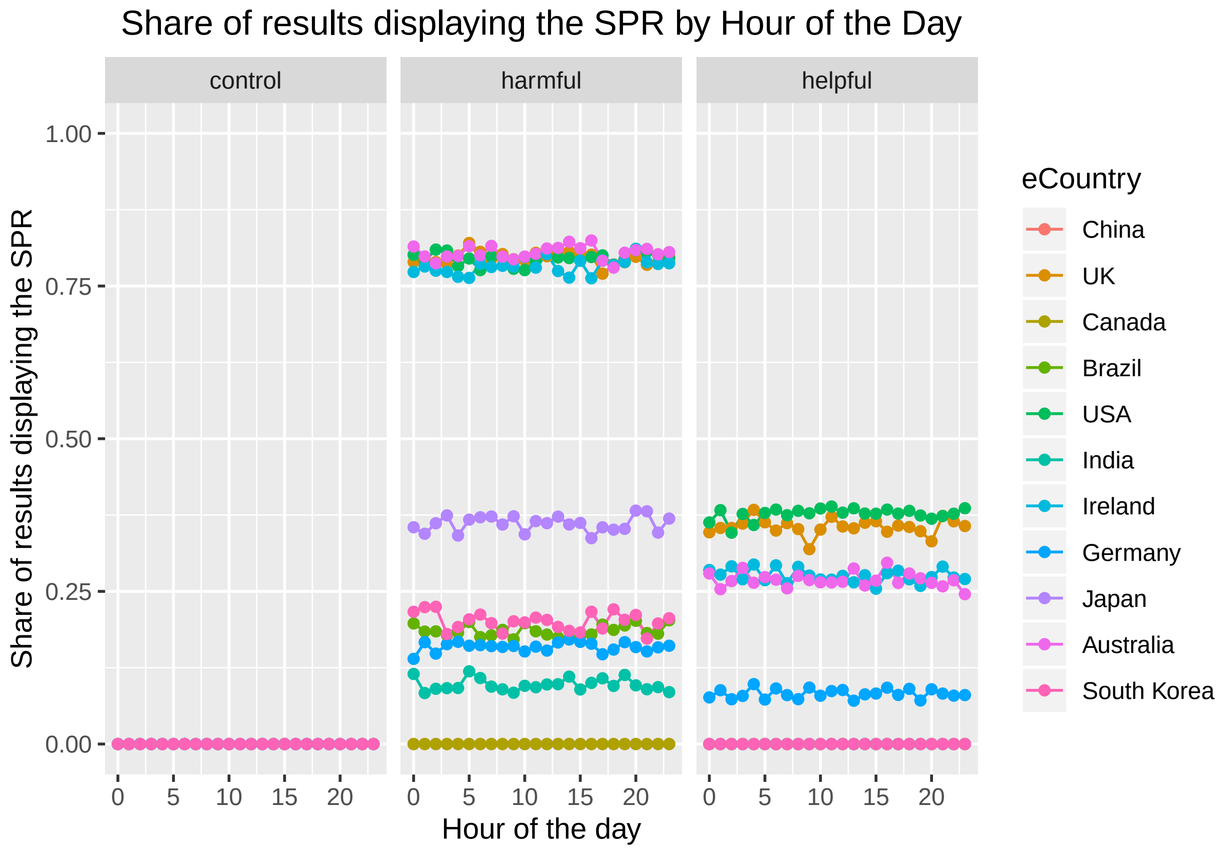
```
# This is an effective linear model
SPRlm <- lm(bHelplineShown ~ eCountry*eGroup, data = tDataEx3)

# plot of linear model coefficients
plot_model(SPRlm, sort.est=TRUE, wrap.labels = 75, rm.terms = c("eCountry [Ireland]", "eCountry [UK]",
       theme(plot.title = element_text(hjust = 0.5))
```



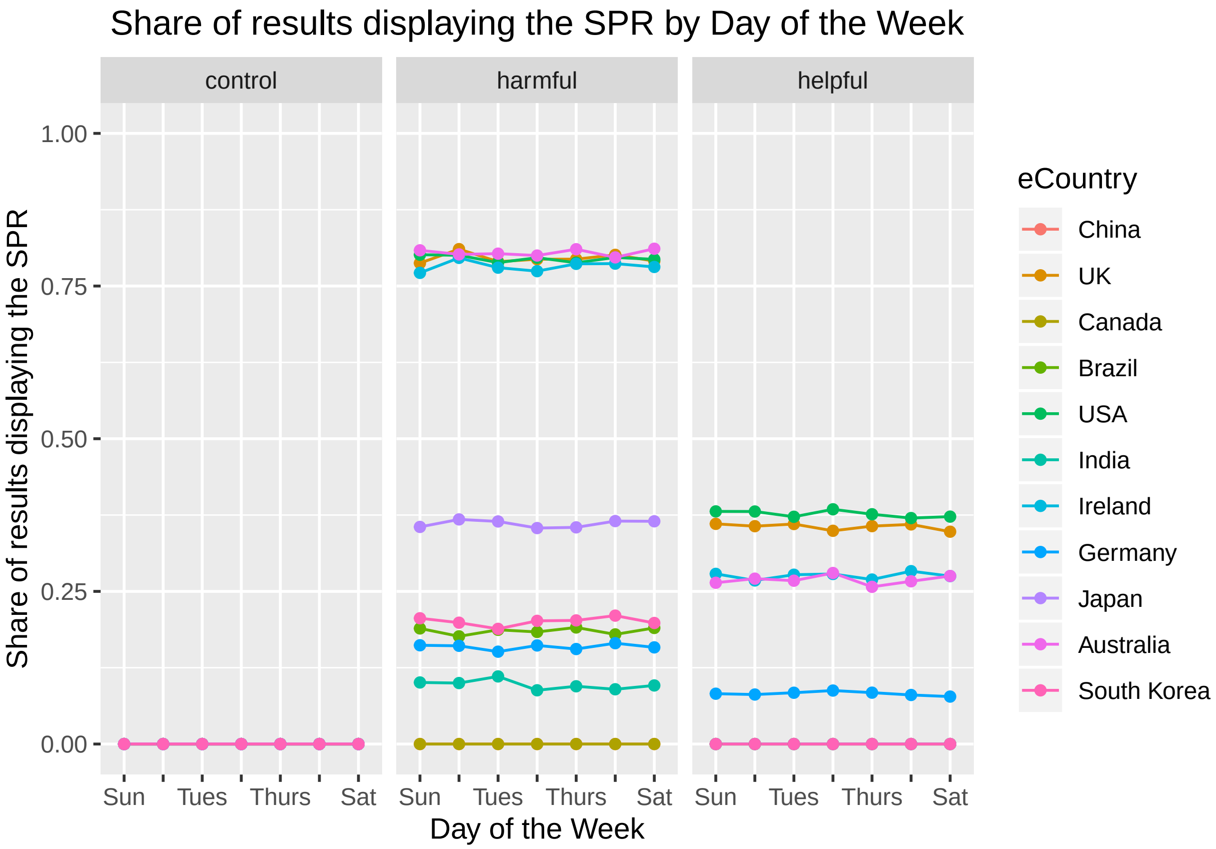
SPR display rate by time and day of week

```
# SPR by time and country
tDataEx3 %>%
  mutate(Hour = hour(dCreate)) %>%
  group_by(Hour, eCountry, eGroup) %>%
  summarize(Percent_shown = sum(bHelplineShown) / n()) %>%
  ggplot(aes(x = Hour, y = Percent_shown, group = eCountry, color = eCountry)) +
  geom_line() +
  geom_point() +
  facet_wrap(~eGroup) +
  scale_y_continuous(limits = c(0, 1)) +
  labs(title = "Share of results displaying the SPR by Hour of the Day",
       x = "Hour of the day",
       y = "Share of results displaying the SPR")  +
  theme(plot.title = element_text(hjust = 0.5))
```



```
# ggsave(filename = "Plots/Time.png",
#        plot = last_plot(),
#        device = "png",
#        width = 10,
#        height = 4)

# SPR by weekday and country
tDataEx3 %>%
  mutate(WeekDay = wday(dCreate)) %>%
  group_by(WeekDay, eCountry, eGroup) %>%
  summarize(Percent_shown = sum(bHelplineShown) / n()) %>%
  ggplot(aes(x = as.factor(WeekDay), y = Percent_shown, group = eCountry, color = eCountry)) +
  geom_line() +
  geom_point() +
  facet_wrap(~eGroup) +
  scale_y_continuous(limits = c(0, 1)) +
  scale_x_discrete(labels = c("Sun", "", "Tues", "",
                              "Thurs", "", "Sat")) +
  labs(title = "Share of results displaying the SPR by Day of the Week",
       x = "Day of the Week",
       y = "Share of results displaying the SPR") +
  theme(plot.title = element_text(hjust = 0.5))
```



```
# ggsave(filename = "Plots/WeekDay.png",
#        plot = last_plot(),
#        device = "png",
#        width = 10,
#        height = 4)
```

Contrast model use English as reference group (extra)

```
data <- tDataEx4
res1 <- lm(bHelplineShown~factor(eLanguage), data=data)
language.est = lsmeans(res1, "eLanguage")
#use English as reference group
Contrasts.English.ref = list(Mandarin.vs.English = c(1, -1, 0, 0), Hindi.vs.English = c(1, 0, -1,0), Te
Test1 = contrast(language.est, Contrasts.English.ref)
test(Test1)
```

```
##  contrast            estimate      SE     df t.ratio p.value
##  Mandarin.vs.English    0.327 0.00184 202305 177.744 <.0001
##  Hindi.vs.English       0.299 0.00184 202305 162.204 <.0001
##  Telugu.vs.English      0.327 0.00181 202305 181.069 <.0001
```

Contrast model compare each group to overall mean

```
#compare each group to overall mean
Contrasts.group.averageref = list(English.vs.average = c(3/4, -1/4,-1/4,-1/4), Mandarin.vs.average = c(
Test2 = contrast(language.est, Contrasts.group.averageref)
test(Test2)
```

```
##  contrast            estimate      SE     df t.ratio p.value
##  English.vs.average    0.2384 0.00103 202305 231.152 <.0001
##  Mandarin.vs.average  -0.0891 0.00126 202305 -70.405 <.0001
##  Hindi.vs.average     -0.0603 0.00126 202305 -47.678 <.0001
##  Telugu.vs.average    -0.0891 0.00124 202305 -71.805 <.0001
```