

P3 analysis

Mark Whiting

2/26/2020

We first build several core data frames:

- **Tasks** lists the 11 tasks we studied and their IDs
- **Questions** lists the 27 questions, their IDs and the answer type they use
- **Responses** lists 407 total responses, broken out by task-resondant pairs, with questions as columns. These encompass responses from 37 unique individuals.

To evaluate the inter rater reliability, we compute Fleiss' Kappa

[illegible]

and Krippendorff's Alpha

[illegible]

-0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03,
-0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03, -0.03

for each task-question pair (a 26, 11 matrix, because one question was dropped due to being a free text response). However, this does not give us what we want as each position in these results is calculated from a 1 dimensional vector, which is not how these statistics are intended to be used. All this tells us is that

Note that in Fleiss' Kappa, \mathbf{NA} for a 1 dimensional vector is similar to 1, as in no variance across respondents. Krippendorff's Alpha does this correctly and shows 1 in those positions.

So, we'd rather do this for subsets of the data, e.g. for each task, find the agreement across raters and questions, or for each question, find the agreement across raters and tasks, or even, for each rater, find the agreement across tasks and questions. In the next section, these assessments are performed.

```
questionNames = names(responses %>% select(-ResponseId, -Task))
perTaskAgreement = responses %>% group_by(Task) %>%
  summarise(
    fleissKappa = kappam.fleiss(t(cbind(
      .data[[questionNames[1]]],
      .data[[questionNames[2]]],
      .data[[questionNames[3]]],
      .data[[questionNames[4]]],
      .data[[questionNames[5]]],
      .data[[questionNames[6]]],
      .data[[questionNames[7]]],
      .data[[questionNames[8]]],
      .data[[questionNames[9]]],
      .data[[questionNames[10]]],
      .data[[questionNames[11]]],
      .data[[questionNames[12]]],
      .data[[questionNames[13]]],
      .data[[questionNames[14]]],
      .data[[questionNames[15]]],
      .data[[questionNames[16]]],
      .data[[questionNames[17]]],
      .data[[questionNames[18]]],
      .data[[questionNames[19]]],
      .data[[questionNames[20]]],
      .data[[questionNames[21]]],
      .data[[questionNames[22]]],
      .data[[questionNames[23]]],
      .data[[questionNames[24]]],
      .data[[questionNames[25]]],
      .data[[questionNames[26]]]
    )))$value
)
```

```
# kappam.fleiss(subsetResponse)
```

```
questionNames = names(responses %>% select(-ResponseId, -Task))

perQuestionAgreement = as.data.frame(Reduce(rbind, Map(function(question) {
  responses %>% select(ResponseId, Task, !!question)
  return(cbind(question, fleissKappa = kappam.fleiss(responses %>%
    select(ResponseId, Task, !!question) %>%
    spread(ResponseId, !!question))$value))
})))
```

```

},questionNames)))

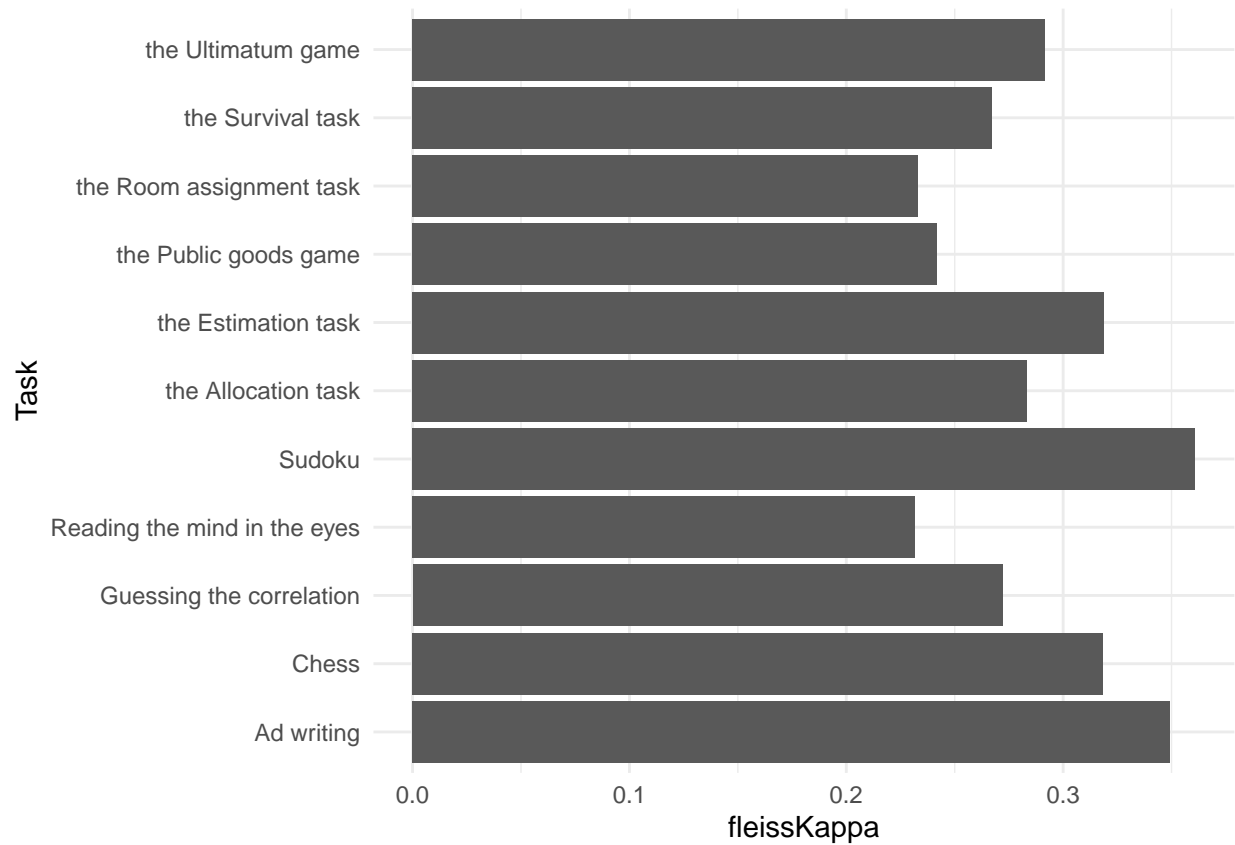
perQuestionAgreement <- perQuestionAgreement %>% mutate(fleissKappa = as.numeric(as.character(fleissKappa)))

questionNames = names(responses %>% select(-ResponseId, -Task))

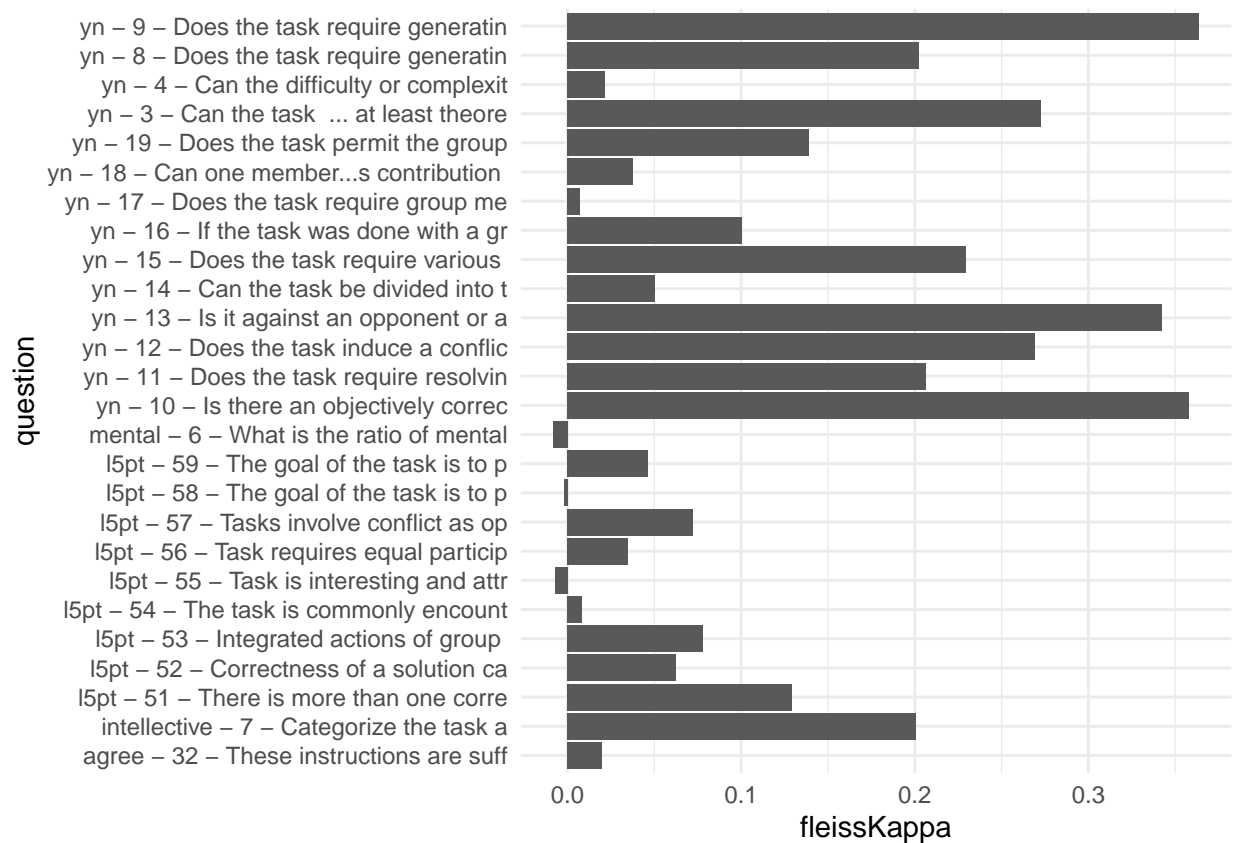
perRaterAgreement = responses %>% group_by(ResponseId) %>%
  summarise(
    fleissKappa = kappam.fleiss(t(cbind(
      .data[[questionNames[1]]],
      .data[[questionNames[2]]],
      .data[[questionNames[3]]],
      .data[[questionNames[4]]],
      .data[[questionNames[5]]],
      .data[[questionNames[6]]],
      .data[[questionNames[7]]],
      .data[[questionNames[8]]],
      .data[[questionNames[9]]],
      .data[[questionNames[10]]],
      .data[[questionNames[11]]],
      .data[[questionNames[12]]],
      .data[[questionNames[13]]],
      .data[[questionNames[14]]],
      .data[[questionNames[15]]],
      .data[[questionNames[16]]],
      .data[[questionNames[17]]],
      .data[[questionNames[18]]],
      .data[[questionNames[19]]],
      .data[[questionNames[20]]],
      .data[[questionNames[21]]],
      .data[[questionNames[22]]],
      .data[[questionNames[23]]],
      .data[[questionNames[24]]],
      .data[[questionNames[25]]],
      .data[[questionNames[26]]]
    )))$value
  )

ggplot(perTaskAgreement,aes(Task,fleissKappa)) +
  geom_col() +
  coord_flip() +
  theme_minimal()

```



```
ggplot(perQuestionAgreement, aes(question, fleissKappa)) +
  geom_col() +
  scale_x_discrete(labels=substring(perQuestionAgreement$question, 0, 40)) +
  coord_flip() +
  theme_minimal()
```



```
ggplot(perRaterAgreement, aes(ResponseId, fleissKappa)) +
  geom_col() +
  coord_flip() +
  theme_minimal()
```

