Charlie regression analysis

Carolyn Jane Anderson

2023-03-28

This notebook contains code for running regression models to explore the effect of prompt wording choices in the StudentEval dataset. It reads in pass@1 rates and feature counts obtained using the token_to_feature.py script.

```
data.raw = read.csv('../raw_data/interactions.csv',header=TRUE,stringsAsFactors=FALSE)

data <- data.raw %>% mutate(success = ifelse(tests_passed==total_tests,1,0),
                            firstAttempt = ifelse(first_attempt=="True",1,0),
                            lastAttempt = ifelse(last_attempt=="True",1,0),
                            group = ifelse(first_attempt=="True"&success==1,"SuccessFirst",ifelse(first_
                                           "UnsuccessFirst",
                                           ifelse(last_attempt=="True"&success==1,
                                                  "SuccessLast",
                                                  ifelse(last_attempt=="True","UnsuccessLast","Middle")
```

```
exclude = read.csv('exclude.csv',header=TRUE,stringsAsFactors=FALSE) %>% mutate(joined = paste(problem,s
data.sub <- data %>% mutate(join_ed = paste(problem,str_trim(submitted_text)))
remove <- subset(data.sub,join_ed %in% exclude$joined)
cleaned <- subset(data.sub,!(join_ed %in% exclude$joined))
```

```
pass.raw = read.csv('../computed_data/allprompts_starcoderbase_pass1.csv',header=TRUE)
pass <- subset(pass.raw,select=c("prompt","pass1"))
data.all <- merge(cleaned,pass,by="prompt")
```

```
features.raw <- read.delim("tokenized_features.tsv",header=TRUE) %>% mutate(id = prompt)
features.r <- subset(features.raw,select=-c(1))
features <- merge(data.all,features.r,by="id")
```

```
charCount <- function(x,c){
  counts <- unlist(map(x,function(y) str_count(tolower(y),"dict")/str_length(y)))
  counts - mean(counts)/sd(counts)
  }
```

```
data.all <- merge(cleaned,pass,by="prompt") %>% rowwise() %>%
  mutate(totalLength = str_length(submitted_text),
         longestSentence = max(unlist(map(str_split(submitted_text,"\\."),str_length))),
         sentCount = length(str_split(submitted_text,"\\.")[[1]]))
```

```
data.all %>% group_by(group) %>% summarize(a.totalLength = mean(totalLength),
                                           a.longSentence = mean(longestSentence),
                                           a.sentCount = mean(sentCount)
                                                                )
```

```
## # A tibble: 5 x 4
##   group          a.totalLength a.longSentence a.sentCount
```

```
##   <chr>                  <dbl>          <dbl>       <dbl>
## 1 Middle                  186.           115.        2.87
## 2 SuccessFirst            153.           105.        2.44
## 3 SuccessLast             204.           116.        3.06
## 4 UnsuccessFirst          157.           103.        2.55
## 5 UnsuccessLast           207.           115.        3.19
```

```
model <- lmer(pass1 ~ I(totalLength/100) * I(longestSentence/100) + (1+I(totalLength/100) + I(longestSe
```

```
summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## pass1 ~ I(totalLength/100) * I(longestSentence/100) + (1 + I(totalLength/100) +
##     I(longestSentence/100) | problem)
##    Data: data.all
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))
##
## REML criterion at convergence: 663.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0096 -0.5073 -0.1780  0.1407  3.4712
##
## Random effects:
##  Groups   Name                    Variance Std.Dev. Corr
##  problem  (Intercept)             0.07789  0.2791
##           I(totalLength/100)      0.01666  0.1291   -0.59
##           I(longestSentence/100) 0.01606  0.1267   -0.12 -0.38
##  Residual                         0.07486  0.2736
## Number of obs: 1648, groups:  problem, 48
##
## Fixed effects:
##                                            Estimate Std. Error        df
## (Intercept)                                 0.09649    0.05077  59.77929
## I(totalLength/100)                          0.07270    0.02505  37.33341
## I(longestSentence/100)                      0.05827    0.03732  58.13778
## I(totalLength/100):I(longestSentence/100)  -0.01916    0.01010 156.16636
##                                           t value Pr(>|t|)
## (Intercept)                                 1.900  0.06222 .
## I(totalLength/100)                          2.903  0.00618 **
## I(longestSentence/100)                      1.561  0.12385
## I(totalLength/100):I(longestSentence/100)  -1.897  0.05963 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) I(tL/100) I(S/10
## I(ttlL/100) -0.599
## I(lngS/100) -0.459 -0.067
## I(L/100):I(  0.451 -0.395    -0.650
```

```
model <- lmer(pass1 ~ I(totalLength/100) + (1+I(totalLength/100)|problem),data=data.all,control=lmerCon
```

```
summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: pass1 ~ I(totalLength/100) + (1 + I(totalLength/100) | problem)
##    Data: data.all
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))
##
## REML criterion at convergence: 662
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0210 -0.5006 -0.1928  0.1266  3.4255
##
## Random effects:
##  Groups   Name                Variance Std.Dev. Corr
##  problem  (Intercept)         0.07428  0.2725
##           I(totalLength/100) 0.01435  0.1198   -0.68
##  Residual                     0.07671  0.2770
## Number of obs: 1648, groups:  problem, 48
##
## Fixed effects:
##                     Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)          0.15021    0.04314 42.75592   3.482  0.00116 **
## I(totalLength/100)  0.05772    0.02037 30.86956   2.833  0.00805 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## I(ttlL/100) -0.714
```

```
model <- lmer(pass1 ~ paramInd+functionnameInd+(1|problem),data=features,control=lmerControl(optimizer=
```

```
summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: pass1 ~ paramInd + functionnameInd + (1 | problem)
##    Data: features
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))
##
## REML criterion at convergence: 756.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.8033 -0.5148 -0.2374  0.1316  3.1610
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  problem  (Intercept) 0.03698  0.1923
##  Residual             0.08507  0.2917
## Number of obs: 1648, groups:  problem, 48
##
## Fixed effects:
```

```
##                   Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)      2.454e-01  2.950e-02  4.881e+01   8.322 6.36e-11 ***
## paramInd         8.798e-03  1.798e-02  1.642e+03   0.489   0.6247
## functionnameInd -7.394e-02  3.202e-02  1.624e+03  -2.309   0.0211 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) prmInd
## paramInd    -0.192
## functnnmInd -0.008 -0.238
```

```
model <- lmer(pass1 ~ listInd+dictInd+squareBraceInd+curlyBraceInd+arrayInd+variableInd+numberInd+intIn
```

```
summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: pass1 ~ listInd + dictInd + squareBraceInd + curlyBraceInd +
##     arrayInd + variableInd + numberInd + intInd + (1 | problem)
##    Data: features
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+08))
##
## REML criterion at convergence: 767
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.6642 -0.5273 -0.2296  0.0986  3.3031
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  problem  (Intercept) 0.03762  0.1939
##  Residual             0.08470  0.2910
## Number of obs: 1648, groups:  problem, 48
##
## Fixed effects:
##                   Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)      2.164e-01  3.135e-02  5.980e+01   6.902 3.73e-09 ***
## listInd          4.182e-02  1.862e-02  1.625e+03   2.246   0.0248 *
## dictInd          6.222e-03  4.712e-02  1.628e+03   0.132   0.8950
## squareBraceInd -2.108e-01  3.697e-01  1.603e+03  -0.570   0.5686
## curlyBraceInd   3.730e-01  2.142e-01  1.612e+03   1.741   0.0818 .
## arrayInd       -7.299e-02  3.695e-02  1.629e+03  -1.975   0.0484 *
## variableInd     2.650e-02  3.855e-02  1.611e+03   0.687   0.4919
## numberInd       9.049e-03  1.895e-02  1.637e+03   0.477   0.6331
## intInd          2.311e-02  1.953e-02  1.639e+03   1.183   0.2369
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) lstInd dctInd sqrBrI crlyBI arryIn vrblIn nmbrIn
## listInd     -0.305
## dictInd      0.003 -0.060
## squarBrcInd  0.029 -0.030 -0.080
## curlyBrcInd -0.029  0.024  0.019 -0.583
```

```
## arrayInd    -0.109  0.203 -0.047 -0.062  0.010
## variableInd  0.019 -0.086 -0.191 -0.066 -0.013 -0.065
## numberInd   -0.130 -0.046 -0.086 -0.052  0.045 -0.047 -0.115
## intInd      -0.106 -0.125 -0.056 -0.006  0.019 -0.011 -0.011  0.098
```

```
model <- lmer(pass1 ~ returnInd+inputInd+printInd+outputInd + (1|problem),data=features,control=lmerCon
```

```
summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: pass1 ~ returnInd + inputInd + printInd + outputInd + (1 | problem)
##    Data: features
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))
##
## REML criterion at convergence: 755.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.7512 -0.5606 -0.2379  0.0908  3.3380
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  problem  (Intercept) 0.03540  0.1881
##  Residual             0.08456  0.2908
## Number of obs: 1648, groups:  problem, 48
##
## Fixed effects:
##               Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)  1.969e-01  3.190e-02  7.165e+01   6.173 3.60e-08 ***
## returnInd    7.017e-02  1.657e-02  1.641e+03   4.233 2.43e-05 ***
## inputInd     2.174e-02  2.847e-02  1.620e+03   0.764    0.445
## printInd    -8.358e-03  2.693e-02  1.624e+03  -0.310    0.756
## outputInd    2.499e-02  2.239e-02  1.621e+03   1.116    0.265
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##           (Intr) rtrnIn inptIn prntIn
## returnInd -0.262
## inputInd  -0.196  0.014
## printInd  -0.022 -0.040 -0.803
## outputInd -0.127  0.235 -0.129  0.097
```

```
model <- lmer(pass1 ~ exampleInd+consecutiveInd+representInd + (1|problem),data=features,control=lmerCo
```

```
summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: pass1 ~ exampleInd + consecutiveInd + representInd + (1 | problem)
##    Data: features
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))
##
## REML criterion at convergence: 764.2
##
```

```
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.7688 -0.5186 -0.2485  0.1103  3.1840
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  problem  (Intercept) 0.03725  0.1930
##  Residual             0.08539  0.2922
## Number of obs: 1648, groups:  problem, 48
##
## Fixed effects:
##                  Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)     2.442e-01  2.915e-02  4.594e+01   8.379 8.29e-11 ***
## exampleInd     -1.181e-02  6.137e-02  1.616e+03  -0.192    0.847
## consecutiveInd  3.049e-03  3.023e-02  1.636e+03   0.101    0.920
## representInd    3.970e-03  4.265e-02  1.628e+03   0.093    0.926
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) exmplI cnsctI
## exampleInd  -0.019
## consectvInd -0.077 -0.040
## represntInd -0.014 -0.164 -0.428
```

```
model <- lmer(pass1 ~ elementInd+indexInd+keyInd + (1|problem),data=features,control=lmerControl(optimi
```

```
summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: pass1 ~ elementInd + indexInd + keyInd + (1 | problem)
##    Data: features
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))
##
## REML criterion at convergence: 750.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.8031 -0.5467 -0.2339  0.1048  3.2345
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  problem  (Intercept) 0.03982  0.1995
##  Residual             0.08441  0.2905
## Number of obs: 1648, groups:  problem, 48
##
## Fixed effects:
##              Estimate Std. Error        df t value Pr(>|t|)
## (Intercept) 2.274e-01  3.027e-02 4.700e+01   7.511 1.38e-09 ***
## elementInd  3.769e-02  2.789e-02 1.627e+03   1.351 0.176811
## indexInd    3.377e-02  2.576e-02 1.638e+03   1.311 0.190008
## keyInd      1.143e-01  3.243e-02 1.639e+03   3.523 0.000438 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Correlation of Fixed Effects:
##             (Intr) elmntI indxIn
## elementInd -0.083
## indexInd   -0.086  0.009
## keyInd     -0.102  0.042 -0.024
```