Adjs Soc-Phys: Graphs for Accuracy

morgan moyer

2025-05-02

```r
table(d$Task,d$Label)
```

```
## 
##           test_sp test_val
##   SocPhys    2400        0
##   Valence       0     2400
```

```r
print(unique(d$Word))
```

```
##  [1] "rancid"        "fair"          "amiable"       "prosperous"   
##  [5] "soft"          "arrogant"      "oppressive"    "cruel"        
##  [9] "gritty"        "sour"          "judgmental"    "slim"         
## [13] "brittle"       "hostile"       "gorgeous"      "generous"     
## [17] "radiant"       "scrawny"       "sturdy"        "untrustworthy"
## [21] "selfish"       "corrupt"       "supportive"    "compassionate"
## [25] "moldy"         "lush"          "manipulative"  "elegant"      
## [29] "polite"        "fragrant"      "ugly"          "smooth"       
## [33] "trustworthy"   "youthful"      "sympathetic"   "unjust"       
## [37] "obese"         "wrinkled"      "clammy"        "harmonious"
```

Summary Stats

```r
agr <- d %>%
  group_by(Task) %>%
  summarize(MeanAccuracy = mean(Accuracy),
            SD = sd(Accuracy))
print(agr)
```

```
## # A tibble: 2 x 3
##   Task    MeanAccuracy    SD
##   <chr>          <dbl> <dbl>
## 1 SocPhys        0.842 0.365
## 2 Valence        0.930 0.256
```

```r
print(unique(d$Word))
```

```
##  [1] "rancid"        "fair"          "amiable"       "prosperous"   
##  [5] "soft"          "arrogant"      "oppressive"    "cruel"        
##  [9] "gritty"        "sour"          "judgmental"    "slim"         
## [13] "brittle"       "hostile"       "gorgeous"      "generous"     
## [17] "radiant"       "scrawny"       "sturdy"        "untrustworthy"
## [21] "selfish"       "corrupt"       "supportive"    "compassionate"
## [25] "moldy"         "lush"          "manipulative"  "elegant"      
## [29] "polite"        "fragrant"      "ugly"          "smooth"
## [33] "trustworthy"   "youthful"      "sympathetic"   "unjust"
## [37] "obese"         "wrinkled"      "clammy"        "harmonious"
```

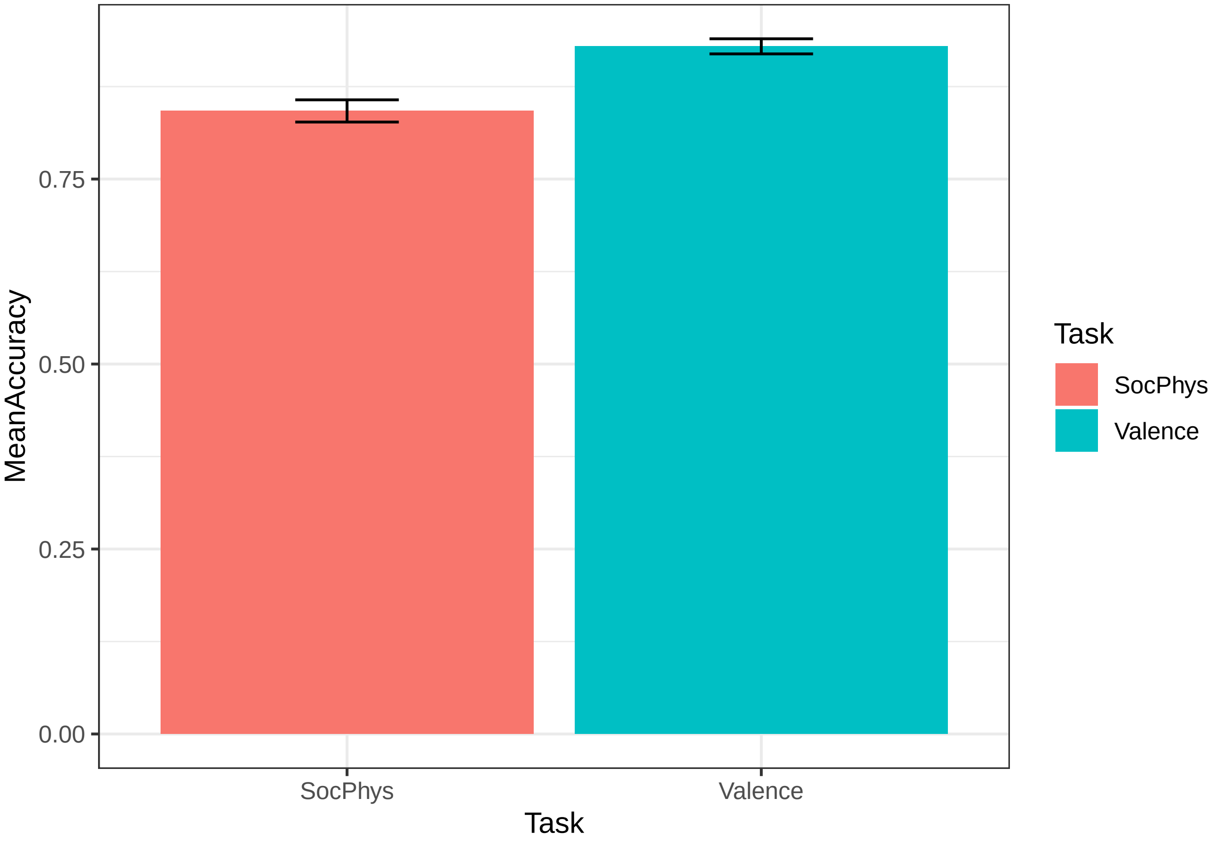
Graph Accuracy by Word

Values for valence/concreteness were gathered/normed first from Warriner et al and Brysbaert et al. From those studies, we can establish what an Accurate response is.

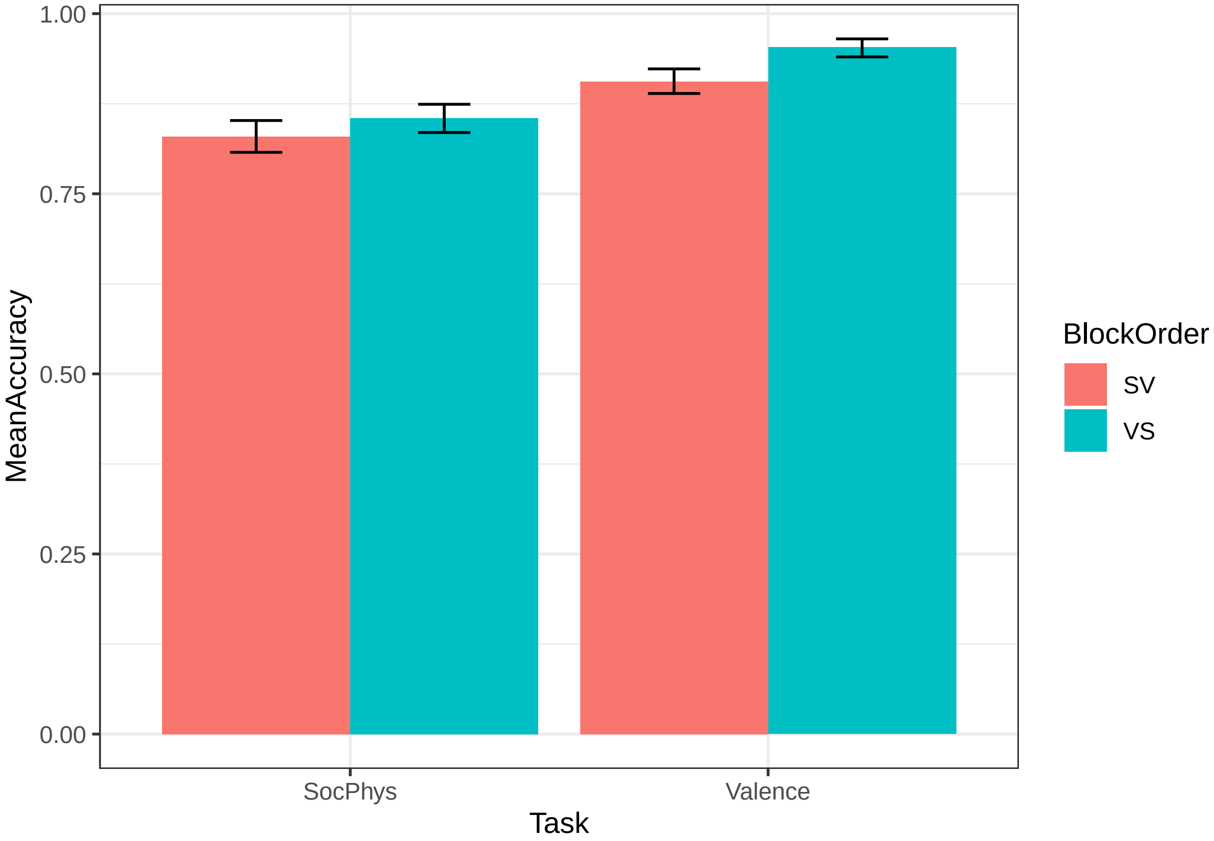
A response is accurate (coded as 1) if the participant response was consistent with the norming study; innacurate (or 0) otherwise.

Overall Accuracy

```
agr <- d %>%
  group_by(Task) %>%
  reframe(MeanAccuracy = mean(Accuracy),
          CILow = ci.low(Accuracy),
          CIHigh = ci.high(Accuracy)) %>%
  mutate(YMin = MeanAccuracy - CILow,
         YMax = MeanAccuracy + CIHigh)
# View(agr)

dodge = position_dodge(.9)
ggplot(data=agr, aes(x=Task,y=MeanAccuracy, fill=Task)) +
  geom_bar(position=dodge,stat="identity") +
  geom_errorbar(aes(ymin=YMin,ymax=YMax),width=.25,position=position_dodge(0.9))
```



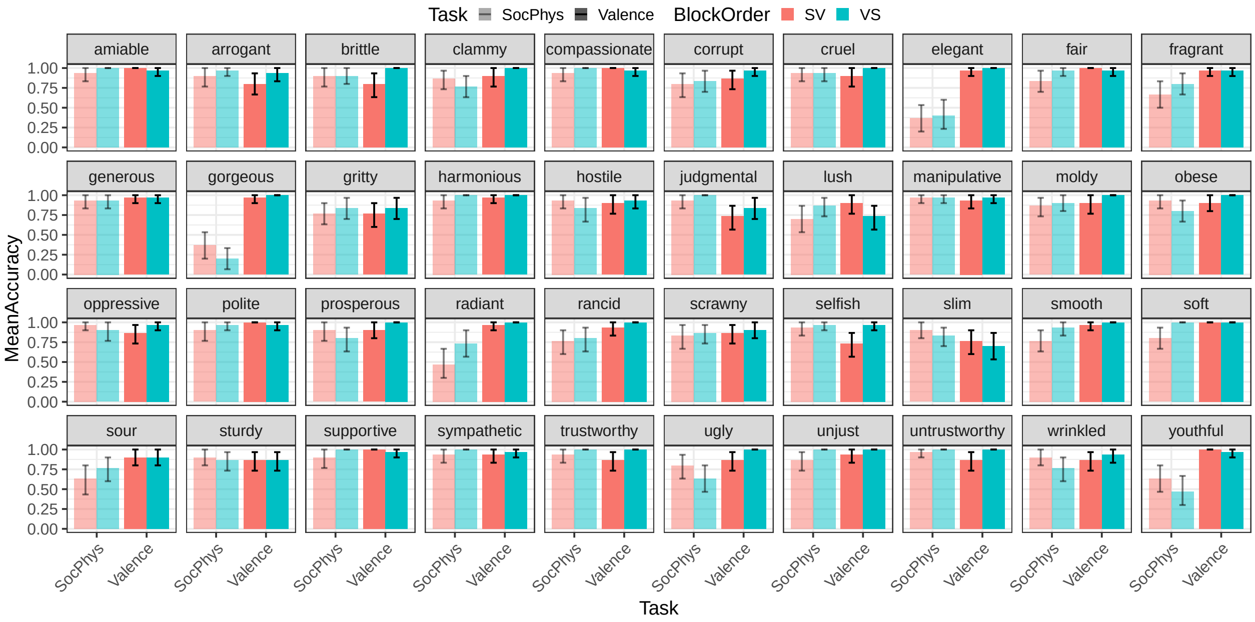
```r
agr <- d %>%
  group_by(Task,BlockOrder) %>%
  reframe(MeanAccuracy = mean(Accuracy),
          CILow = ci.low(Accuracy),
          CIHigh = ci.high(Accuracy)) %>%
  mutate(YMin = MeanAccuracy - CILow,
         YMax = MeanAccuracy + CIHigh)
# View(agr)

dodge = position_dodge(.9)
ggplot(data=agr, aes(x=Task,y=MeanAccuracy, fill=BlockOrder)) +
  geom_bar(position=dodge,stat="identity") +
  geom_errorbar(aes(ymin=YMin,ymax=YMax),width=.25,position=position_dodge(0.9))
```



Mean Accuracy by Word / Task

```r
agr <- d %>%
  group_by(Word,Task,BlockOrder) %>%
  reframe(MeanAccuracy = mean(Accuracy),
          CILow = ci.low(Accuracy),
          CIHigh = ci.high(Accuracy)) %>%
  mutate(YMin = MeanAccuracy - CILow,
         YMax = MeanAccuracy + CIHigh)
# View(agr)

# dodge = position_dodge(.9)
# ggplot(data=agr, aes(x=Task,y=MeanAccuracy,fill=BlockOrder,alpha=Task)) +
#   geom_bar(position=dodge,stat="identity",alpha = 0.8) +
#   facet_wrap(~Word,ncol=10) +
#   geom_errorbar(aes(ymin=YMin,ymax=YMax),width=.25,position=position_dodge(0.9)) +
#   theme(axis.text.x = element_text(angle = 45, hjust = 1))

ggplot(agr,aes(x=Task, y=MeanAccuracy, alpha=Task, fill=BlockOrder)) +
  geom_bar(position="dodge",stat="identity") +
  geom_errorbar(aes(ymin=YMin,ymax=YMax),width=.25,position=position_dodge(0.9)) +
  facet_wrap(~Word, ncol=10) +
  xlab("Task") +
  ylab("MeanAccuracy") +
  # guides(fill=FALSE) +
  guides(alpha=guide_legend(title="Task")) +
  theme(legend.key.size = unit(0.3, "cm"),
        legend.position = "top", # c(.5,1)
        legend.direction = "horizontal",
        legend.margin=margin(0,0,0,0),
        legend.box.margin=margin(0,0,-5,-5),legend.spacing.y = unit(0.001, 'cm')) +
    # scale_fill_manual(values=cbPalette) +
    # scale_color_manual(values=cbPalette) +
    scale_alpha_discrete(range = c(.5,1)) +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

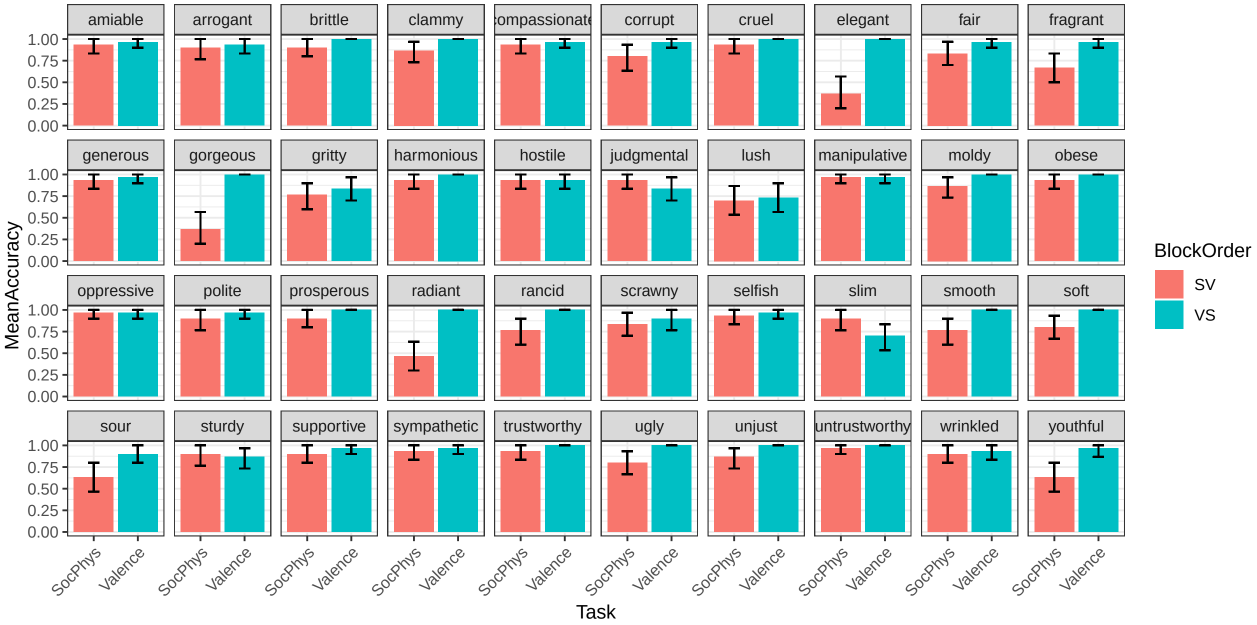
```
## Warning: Using alpha for a discrete variable is not advised.
```



Looking at only the first block

```
agr <- d %>%
  group_by(Task,Word,BlockOrder) %>%
  filter((Task == "Valence") & (BlockOrder == "VS") |
           (Task == "SocPhys") & (BlockOrder == "SV")) %>%
  mutate(MeanAccuracy = mean(Accuracy),
          CILow = ci.low(Accuracy),
          CIHigh = ci.high(Accuracy)) %>%
```

```r
  mutate(YMin = MeanAccuracy - CILow,
         YMax = MeanAccuracy + CIHigh)

agrr <- agr %>%
  group_by(Word,Task) %>%
  select(Word,Task,MeanAccuracy) %>%
  unique()
View(d)

dodge = position_dodge(.9)
ggplot(data=agr, aes(x=Task,y=MeanAccuracy,fill=BlockOrder)) +
  geom_bar(position=dodge,stat="identity") +
  facet_wrap(~Word,ncol=10) +
  geom_errorbar(aes(ymin=YMin,ymax=YMax),width=.25,position=position_dodge(0.9)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```r
m <- lmer(MeanAccuracy ~ BlockOrder + (1|Word), data =agr)
summary(m)
```

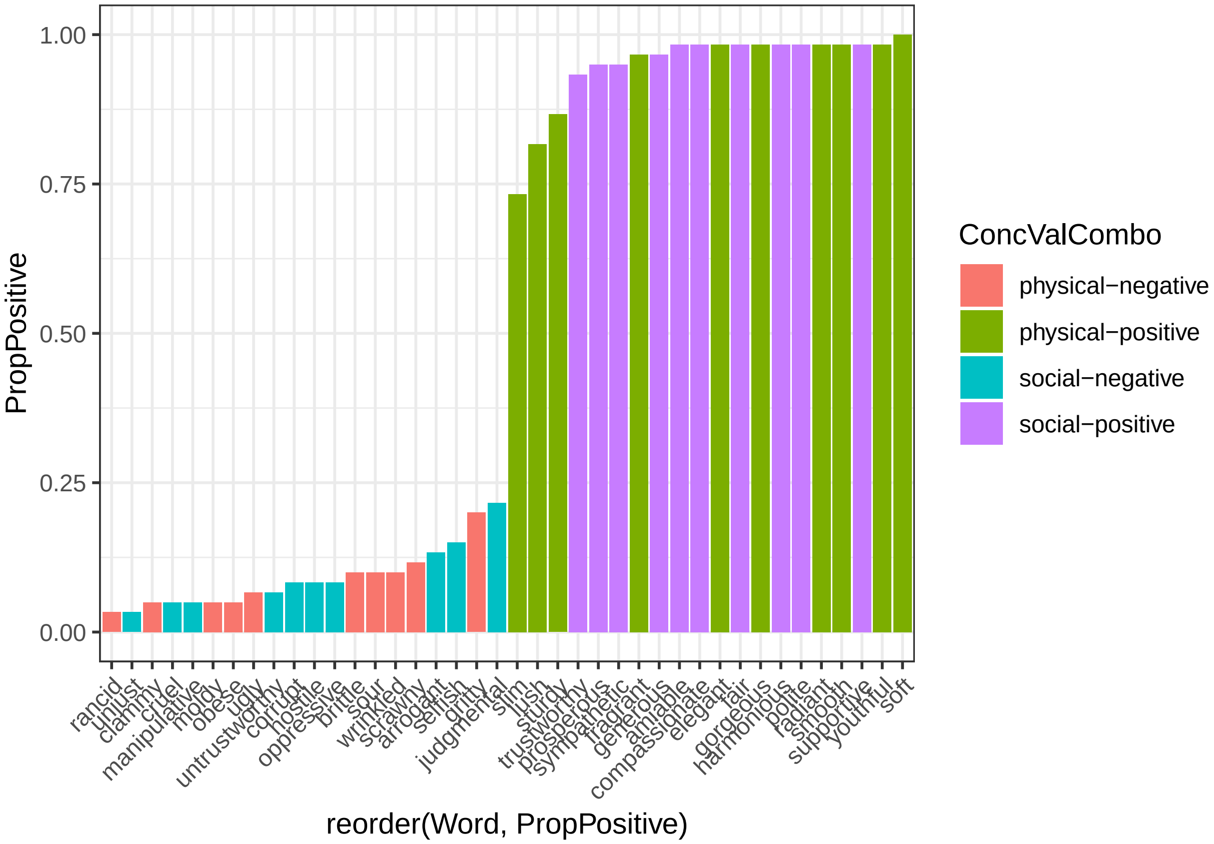
```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: MeanAccuracy ~ BlockOrder + (1 | Word)
##    Data: agr
##
## REML criterion at convergence: -4824.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.02460 -0.51971  0.00527  0.53888  2.93841
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Word     (Intercept) 0.006744 0.08212
##  Residual             0.007291 0.08539
## Number of obs: 2400, groups:  Word, 40
##
## Fixed effects:
##                 Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)  8.292e-01  1.322e-02 4.039e+01   62.73   <2e-16 ***
## BlockOrderVS 1.242e-01  3.486e-03 2.359e+03   35.62   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr)
## BlockOrdrVS -0.132
```

PropPositive and PropObjective

```
val <- d %>%
  filter(Task == "Valence") %>%
  # filter(Word %in% conc$Word) %>%
  group_by(Word,ConcValCombo) %>%
  mutate(Response.n = as.numeric(factor(Response, levels = c("negative", "positive"))) - 1) %>%  # Conv
  summarize(PropPositive = mean(Response.n))
```

```
## `summarise()` has grouped output by 'Word'. You can override using the
## `.groups` argument.
```

```
  # filter(PropPositive > .1 | PropPositive < .9)

dodge = position_dodge(.9)
ggplot(data=val, aes(x=reorder(Word,PropPositive),y=PropPositive,fill=ConcValCombo)) +
  geom_bar(position=dodge,stat="identity") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

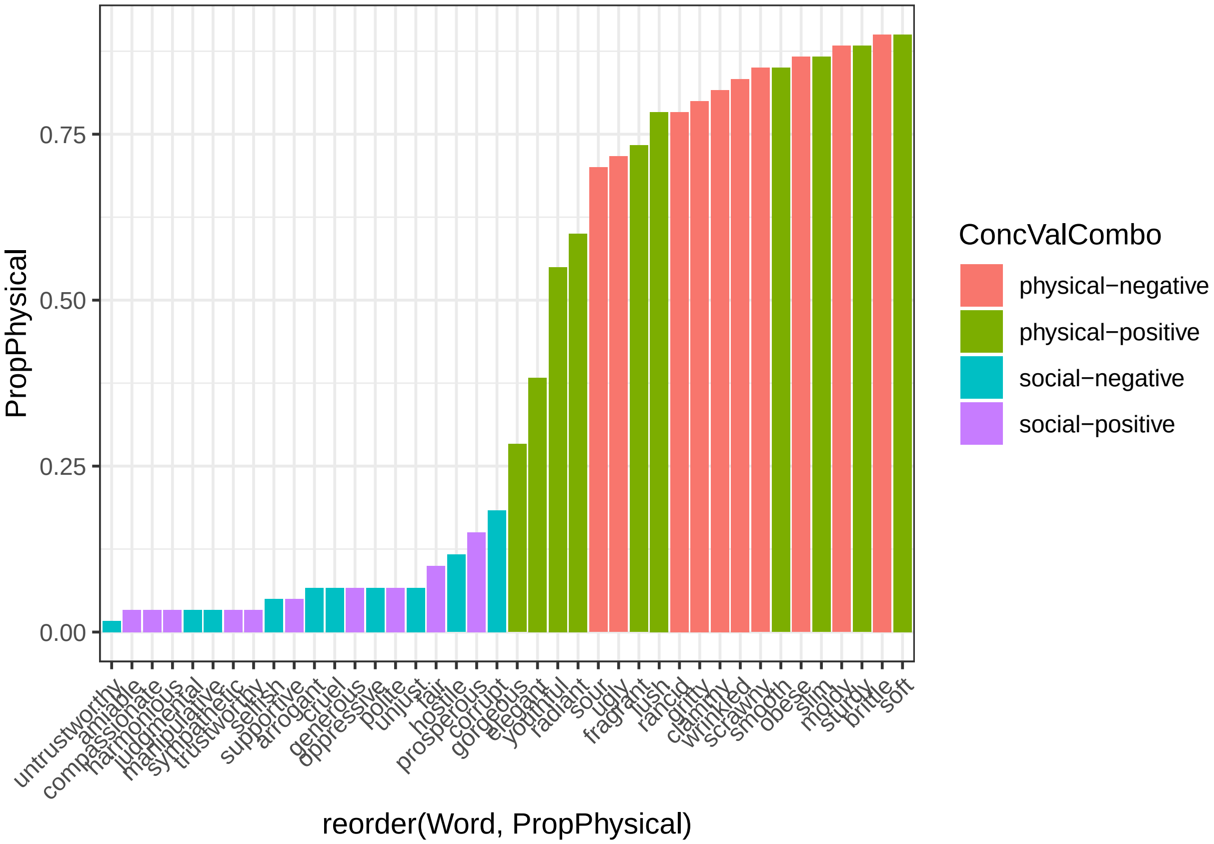


```
  # guides(fill = "none")
```

```
conc <- d %>%
  filter(Task == "SocPhys") %>%
  # filter(Word %in% conc$Word) %>%
  group_by(Word,ConcValCombo) %>%
  mutate(Response.n = as.numeric(factor(Response, levels = c("social", "physical"))) - 1) %>%  # Conver
  summarize(PropPhysical = mean(Response.n))
```

```
## `summarise()` has grouped output by 'Word'. You can override using the
## `.groups` argument.
```

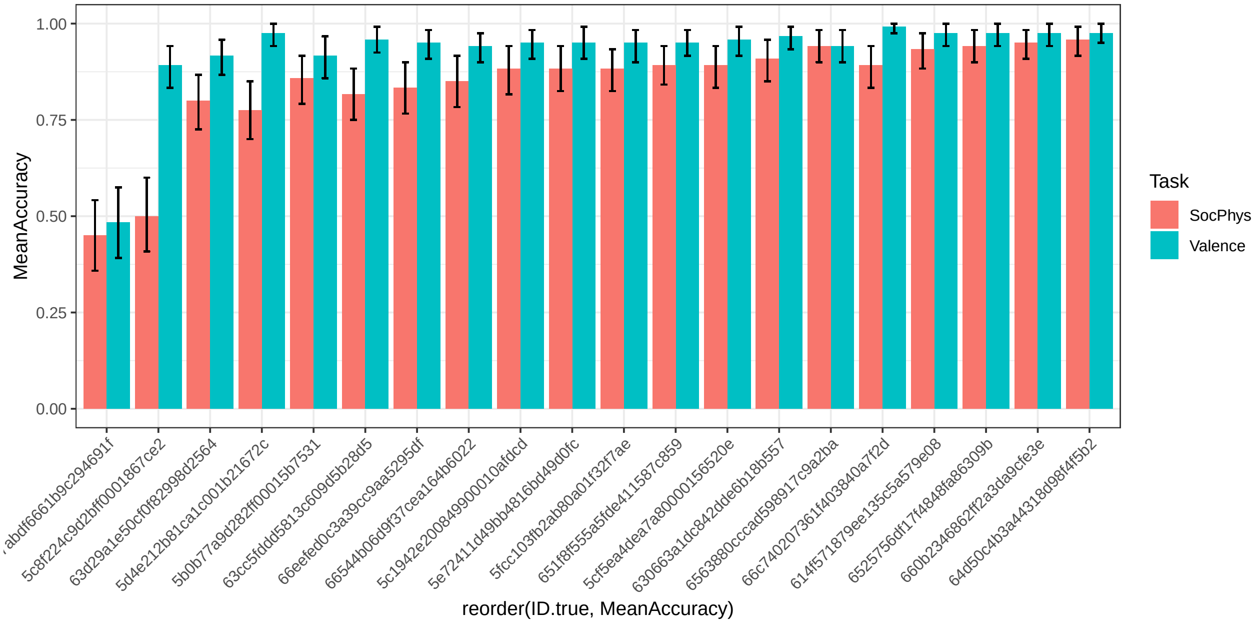
```
  # filter(PropPositive > .1 | PropPositive < .9)

dodge = position_dodge(.9)
ggplot(data=conc, aes(x=reorder(Word,PropPhysical),y=PropPhysical,fill=ConcValCombo)) +
  geom_bar(position=dodge,stat="identity") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
  # guides(fill = "none")
```

Accuracy by Participant

```
agr <- d %>%
  # filter(PennElementType == "Selector") %>%
  select(ID.true,Task,Accuracy) %>%
  group_by(ID.true,Task) %>%
  mutate(MeanAccuracy = mean(Accuracy),
         CILow = ci.low(Accuracy),
         CIHigh = ci.high(Accuracy)) %>%
  mutate(YMin = MeanAccuracy - CILow,
        YMax = MeanAccuracy + CIHigh)

dodge = position_dodge(.9)
ggplot(data=agr, aes(x=reorder(ID.true,MeanAccuracy),y=MeanAccuracy,fill=Task)) +
  geom_bar(position=dodge,stat="identity") +
  geom_errorbar(aes(ymin=YMin,ymax=YMax),width=.25,position=position_dodge(0.9)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



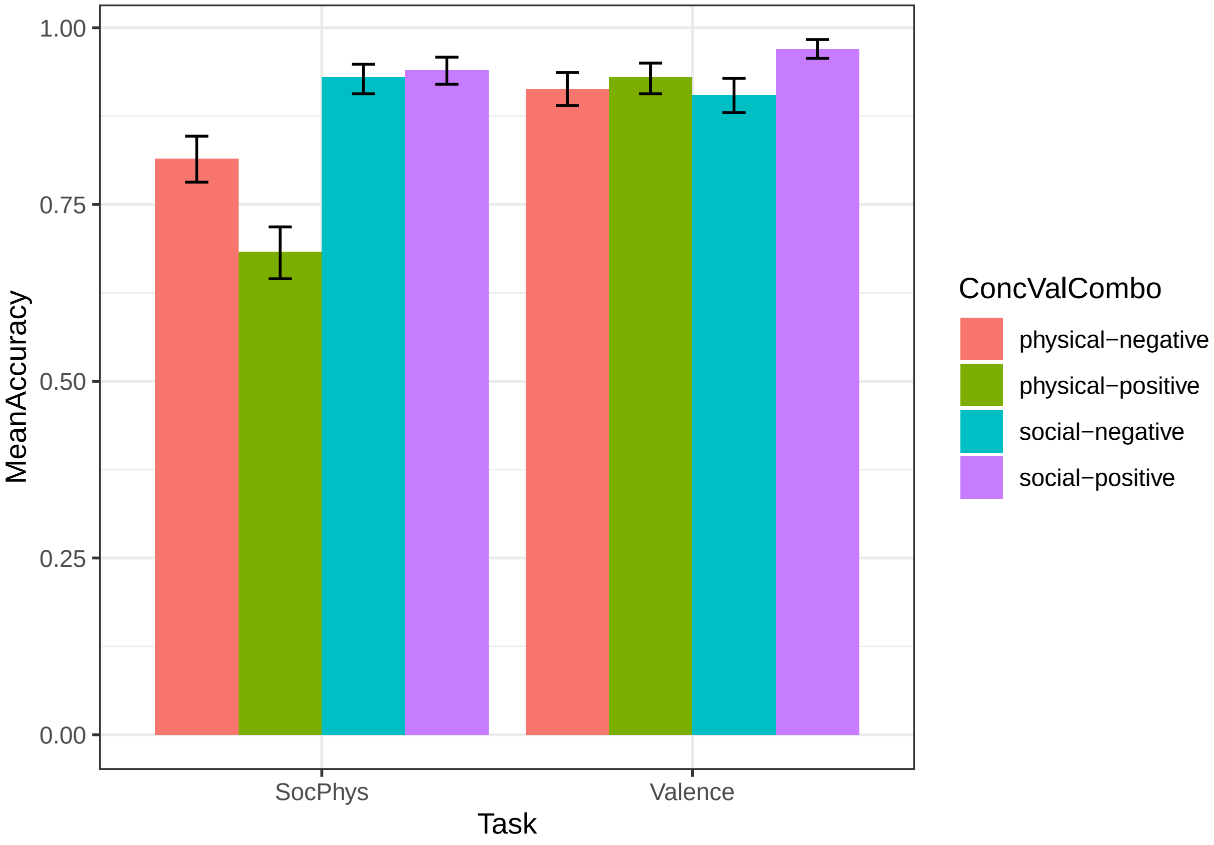
```
  # guides(fill = "none")
```

Mean Accuracy by ConcValCombo

```
agr <- d %>%
  group_by(Task,ConcValCombo) %>%
  summarize(MeanAccuracy = mean(Accuracy), CILow = ci.low(Accuracy), CIHigh = ci.high(Accuracy)) %>%
  mutate(YMin = MeanAccuracy - CILow, YMax = MeanAccuracy + CIHigh)
```

```
## `summarise()` has grouped output by 'Task'. You can override using the
## `.groups` argument.
```

```
dodge = position_dodge(.9)
ggplot(data=agr, aes(x=Task,y=MeanAccuracy,fill=ConcValCombo)) +
  geom_bar(position=dodge,stat="identity") +
  # facet_wrap(~Task) +
  geom_errorbar(aes(ymin=YMin,ymax=YMax),width=.25,position=position_dodge(0.9))
```



```
  # theme(axis.text.x = element_text(angle = 45, hjust = 1))
  # guides(fill = "none")
```

Remove participants who aren't accurate

```
length(unique(d$ID.true))
```

```
## [1] 20
```

```
inacc.parts <- d %>%
  group_by(ID.true,Task) %>%
  summarise(MeanAccuracy = mean(Accuracy)) %>%
  filter(MeanAccuracy < .75)
```

```
## `summarise()` has grouped output by 'ID.true'. You can override using the
## `.groups` argument.
```

```
# How many participants have Accuracy < .75?
length(unique(inacc.parts$ID.true))
```

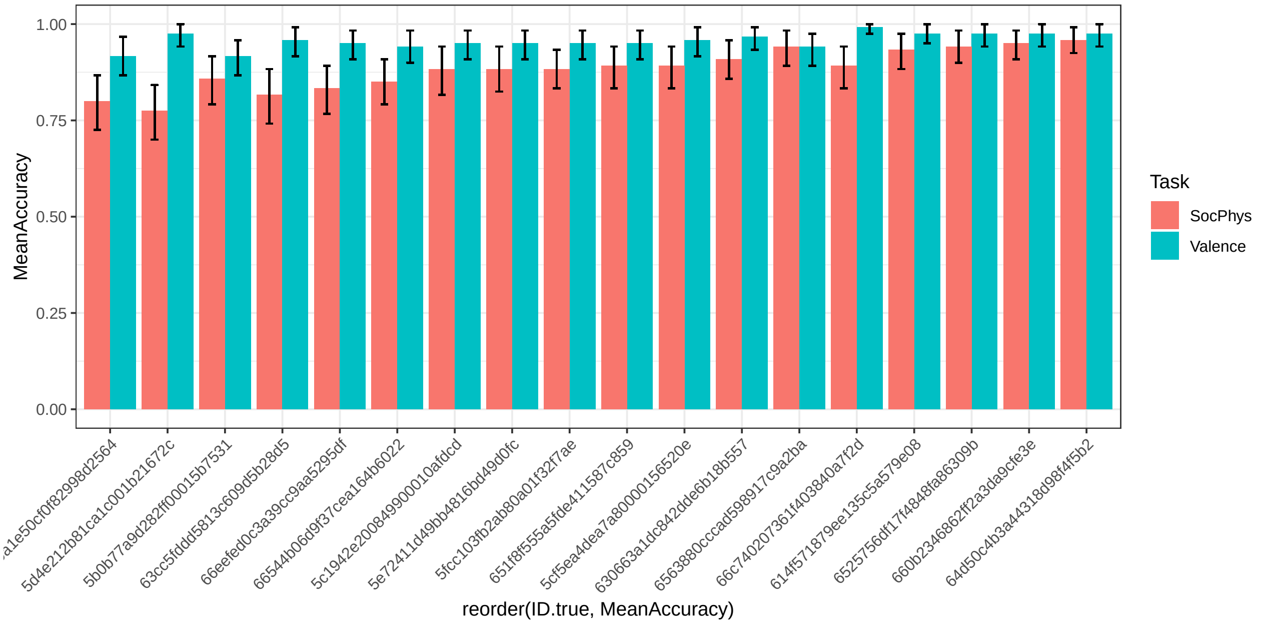
```
## [1] 2
```

```
d.inaccurate.removed <- d %>%
  anti_join(inacc.parts, by = "ID.true")

# Sanity check
length(unique(d.inaccurate.removed$ID.true))
```

```
## [1] 18
```

Accuracy by Participant

```
agr <- d.inaccurate.removed %>%
  # filter(PennElementType == "Selector") %>%
  select(ID.true,Task,Accuracy) %>%
  group_by(ID.true,Task) %>%
  mutate(MeanAccuracy = mean(Accuracy),
         CILow = ci.low(Accuracy),
         CIHigh = ci.high(Accuracy)) %>%
  mutate(YMin = MeanAccuracy - CILow,
        YMax = MeanAccuracy + CIHigh)

dodge = position_dodge(.9)
ggplot(data=agr, aes(x=reorder(ID.true,MeanAccuracy),y=MeanAccuracy,fill=Task)) +
  geom_bar(position=dodge,stat="identity") +
  geom_errorbar(aes(ymin=YMin,ymax=YMax),width=.25,position=position_dodge(0.9)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

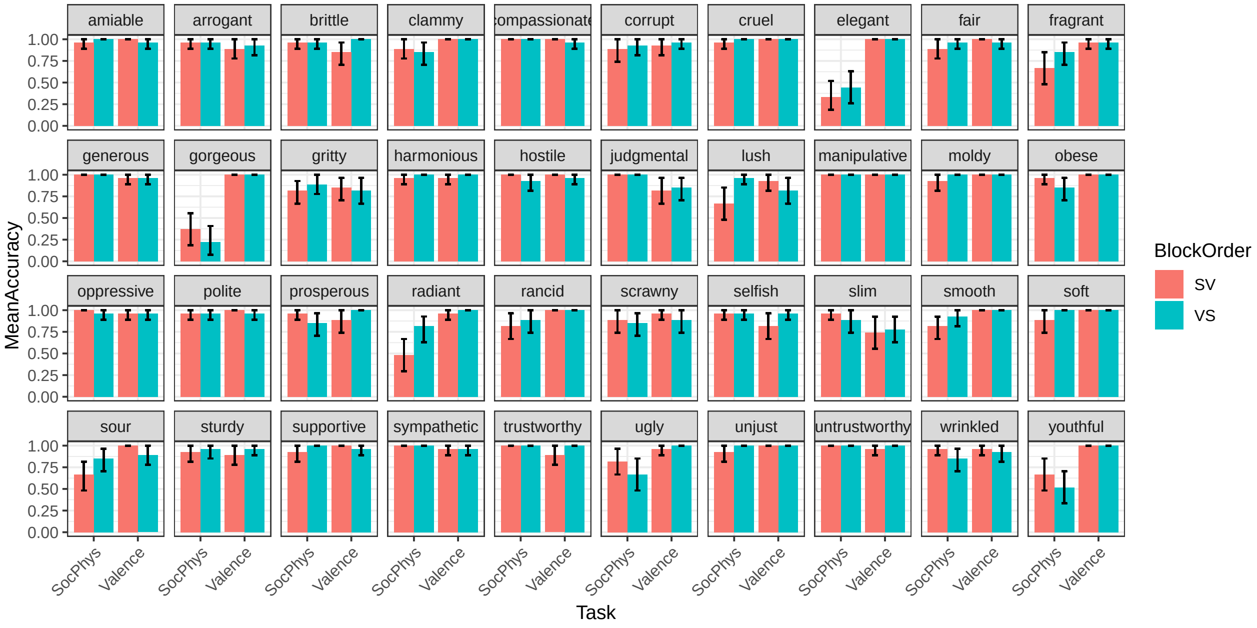


```
  # guides(fill = "none")
```

Mean Accuracy by Word / Task

Looking at only the first block

```
agr <- d.inaccurate.removed %>%
  group_by(Task,Word,BlockOrder) %>%
  # filter((Task == "Valence") & (BlockOrder == "VC") |
  #          (Task == "Concrete") & (BlockOrder == "CV")) %>%
  mutate(MeanAccuracy = mean(Accuracy),
         CILow = ci.low(Accuracy),
         CIHigh = ci.high(Accuracy)) %>%
  mutate(YMin = MeanAccuracy - CILow,
         YMax = MeanAccuracy + CIHigh)

agrr <- agr %>%
  group_by(Word,Task) %>%
  select(Word,Task,MeanAccuracy) %>%
  unique()

dodge = position_dodge(.9)
ggplot(data=agr, aes(x=Task,y=MeanAccuracy,fill=BlockOrder)) +
  geom_bar(position=dodge,stat="identity") +
  facet_wrap(~Word,ncol=10) +
  geom_errorbar(aes(ymin=YMin,ymax=YMax),width=.25,position=position_dodge(0.9)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```


amiable
arrogant
brittle
clammy
ompassionate
corrupt
cruel
elegant
fair
fragrant
generous
gorgeous
gritty
harmonious
hostile
judgmental
lush
manipulative
moldy
obese
oppressive
polite
prosperous
radiant
rancid
scrawny
selfish
slim
smooth
soft
sour
sturdy
supportive
sympathetic
trustworthy
ugly
unjust
untrustworthy
wrinkled
youthful
MeanAccuracy
SocPhys
Valence
Task
BlockOrder
sv
vs