MATH 530/630

Replication Project

Kermit and Miss Piggy

2017-11-29

Selected Paper

The paper we chose was “Different Visual Preference Patterns in Response to Simple and Complex Dynamic Social Stimuli in Preschool-Aged Children with Autism Spectrum Disorders” by Shi et al. Shi L, Zhou Y, Ou J, Gong J, Wang S, Cui X, et al. (2015) Different Visual Preference Patterns in Response to Simple and Complex Dynamic Social Stimuli in Preschool-Aged Children with Autism Spectrum Disorders. PLoS ONE 10(3): e0122280. doi:10.1371/journal.pone.0122280 Academic Editor: Cosimo Urgesi, Uni http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0122280

Abstract

Eye-tracking studies in young children with autism spectrum disorder (ASD) have shown a visual attention preference for geometric patterns when viewing paired dynamic social images (DSIs) and dynamic geometric images (DGIs). In the present study, eye-tracking of two different paired presentations of DSIs and DGIs was monitored in a group of 13 children aged 4 to 6 years with ASD and 20 chronologically age-matched typically developing children (TDC). The results indicated that compared with the control group, children with ASD attended significantly less to DSIs showing two or more children playing than to similar DSIs showing a single child. Visual attention preference in 4- to 6-year-old children with ASDs, therefore, appears to be modulated by the type of visual stimuli.

Methods

This study involves the comparison of eye tracking to different visual stimuli across two groups of children: an autism spectrum disorder group (n=13) with a previous DSM-IV diagnosis of an ASD made by a licensed clinician, and a typically developing group (n=20) recruited from a kindergarten in the same city in China. Exclusion criteria: any developmental disorder or psychiatric disorder, or any immediate family members with an ASD diagnosis.

None of the participants in either group had a history of seizure-associated disorders, acute medical or genetic conditions, or any visual impairment that was uncorrectable with prescription lenses. The two groups were matched in terms of chronological age, not developmental age, to reduce any influence of age-related deviation in object preference.

Goals

A previous study (Pierce K, Conant D, Hazin R, Stoner R, Desmond J. Preference for geometric patterns early in life as a risk factor for autism. Arch Gen Psychiatry. 2011; 68: 101-109) demonstrated that young children, age 14 - 42 months, with ASD spent significantly more time fixating on dynamic geometric images (DGIs) than age-matched children with delayed and typical development characteristics.

In the current study, the authors’ aim was to compare differences in visual attention patterns between children with autism spectrum disorder(ASD) and typically developing children(TDC). In Part I of the study, ASD and TDC children aged 4 to 6 years were shown paired geometric and social image data that was used by Pierce et al that consisted of geometric images and children doing youa. In Part II, the same participants we shown modified stimuli with pair images of two or more children playing and geometric images of images of letters and geometric shapes. Believing that a clear diagnosis of ASD is present, the authors chose to work with older children, age 4-6, than did Pierce et al.

```
#Read in data file
#Scale columns from milliseconds to seconds--needed to reproduce Table 2 correctly
proj_data <- read.table("projectData.csv", header = TRUE, sep = ",")
proj_data["P_gazeduration.to.DSIs.scale"] <- proj_data$P_gazeduration.to.DSIs/1000
proj_data["P_gazeduration.to.DGIs.scale"] <-
proj_data$P_gazeduration.to.DGIs/1000
proj_data["P_time.of.first.fixation.to.DSIs.scale"] <-
proj_data$P_time.of.first.fixation.to.DSIs/1000
proj_data["P_time.of.first.fixation.to.DGIs.scale"] <-
proj_data$P_time.of.first.fixation.to.DGIs/1000

proj_data["S_gazeduration.to.DSIs.scale"] <- proj_data$S_gazeduration.to.DSIs/1000
proj_data["S_gazeduration.to.DGIs.scale"] <-
proj_data$S_gazeduration.to.DGIs/1000
proj_data["S_time.of.first.fixation.to.DSIs.scale"] <-
proj_data$S_time.of.first.fixation.to.DSIs/1000
proj_data["S_time.of.first.fixation.to.DGIs.scale"] <-
proj_data$S_time.of.first.fixation.to.DGIs/1000
```

Table 1 Replication: Sample Characteristics

Table 1 presents the sample characteristics of the two (ASD and TDC) groups. The authors do not name or describe the methods used to create Table 1, but based on the output they report, we were able to match the authors' values by using the Fisher exact test to compare means for gender between the two groups, and the student's t-test to compare mean age and Peabody Picture Vocabulary Test Verbal Intelligence Quotient (PPVT VIQ) scores.

```
#Calculate gender characteristics for both groups (Part A)
gender <- proj_data %>%
  group_by(group, gender) %>%
  summarise(numGender = n())

#Arrange in a matrix
genderMat <- matrix(c(gender$numGender[1], gender$numGender[2], gender$numGender[3],
                      gender$numGender[4]), nrow=2)

#Perform Fisher's exact test of means
genderT <- fisher.test(genderMat)
#Arrange gender counts for two groups and Fisher's test resultsin a table format
fig1ATable <- character(3)
for (i in 1:2) {
  fig1ATable[i] = paste(genderMat[2,i], "/", genderMat[1,i])
}
fig1ATable[3] = round(genderT$p.value, 3)
fig1A <- matrix(fig1ATable, ncol=3)
#Add column names and row name
colnames(fig1A) <- list("ASD", "TDC", "p-value")
rownames(fig1A) <- "Sex M/F"
```

```r
#Calculate the Means and SD for age and PPT VIQ scores for each group, along with the
#results of a t-test comparing the means between groups (Part B).

#Perform t-tests for age and PPVT.VIQ
ageT <- t.test(age.month~group, data=proj_data, var.equal=TRUE)
pptT <- t.test(PPVT.VIQ~group, data=proj_data, var.equal=TRUE)

#Select age and PPVT fields
fig1_data <- proj_data %>%
  select(group, age.month, PPVT.VIQ) %>%
  rename(age_months = age.month)

#Calculate means and standard deviations
means_fig1 <- fig1_data %>%
  group_by(group) %>%
  summarise_each(funs(mean(., na.rm=TRUE)))
```

```
`summarise_each()` is deprecated.
Use `summarise_all()`, `summarise_at()` or `summarise_if()` instead.
To map `funs` over all variables, use `summarise_all()`
```

```r
sd_fig1 <- fig1_data %>%
  group_by(group) %>%
  summarise_each(funs(sd(., na.rm=TRUE)))
```

```
`summarise_each()` is deprecated.
Use `summarise_all()`, `summarise_at()` or `summarise_if()` instead.
To map `funs` over all variables, use `summarise_all()`
```

```r
#Create rows of data for table
age= character(4)
ppt = character(4)
#First, format mean and standard deviation
for (i in 1:2) {
 age[i] <- paste(as.character(round(means_fig1[i,2],2)),
                 " (", as.character(round(sd_fig1[i,2], 2)), ")")
 ppt[i] <- paste(as.character(round(means_fig1[i,3],3)),
                 " (", as.character(round(sd_fig1[i,3],2)), ")")
}
#Next, add t-statistic and p values for each
age[3] = round(ageT$statistic, 2)
age[4] = round(ageT$p.value, 2)
ppt[3] = round(pptT$statistic, 2)
ppt[4] = format(pptT$p.value, scientific=TRUE, digits = 3)

#Create matrix
fig1B <- matrix(c(age, ppt), byrow=TRUE, nrow=2)

#Add rownames and column names
rownames(fig1B)=list("Age", "PPT-VIQ")
colnames(fig1B)=list("ASD Mean(SD)", "TDC Mean(SD)", "t value", "p-value")
```

```r
#Calculate the means, sd and range for the different assessments for the ASD group
#(Part C)
```

```r
#Select data--only from ASD group and columns for assessments
asd_data <- proj_data %>%
  filter(group == "ASD") %>%
  select(ADOSAB, ADOSD, ADOStotalscore, ADIA, ADIB, ADIC, ADID) %>%
  rename(ADOS_social_affect = ADOSAB,
         ADOS_restrictive_repetitive = ADOSD, ADI_R_A = ADIA, ADI_R_B = ADIB,
         ADI_R_C = ADIC, ADI_R_D = ADID )

#Calculate mean, standard deviation, min and max for each column
means_ASD <- asd_data %>%
  summarise_each(funs(mean))
```

```
`summarise_each()` is deprecated.
Use `summarise_all()`, `summarise_at()` or `summarise_if()` instead.
To map `funs` over all variables, use `summarise_all()`
```

```r
sd_ASD <- asd_data %>%
  summarise_each(funs(sd))
```

```
`summarise_each()` is deprecated.
Use `summarise_all()`, `summarise_at()` or `summarise_if()` instead.
To map `funs` over all variables, use `summarise_all()`
```

```r
min_ASD <- asd_data %>%
  summarise_each(funs(min))
```

```
`summarise_each()` is deprecated.
Use `summarise_all()`, `summarise_at()` or `summarise_if()` instead.
To map `funs` over all variables, use `summarise_all()`
```

```r
max_ASD <- asd_data %>%
  summarise_each(funs(max))
```

```
`summarise_each()` is deprecated.
Use `summarise_all()`, `summarise_at()` or `summarise_if()` instead.
To map `funs` over all variables, use `summarise_all()`
```

```r
#Put stats in row format
ASD_stats <- character(7)

#Format stats
for (i in 1:7) {
 ASD_stats[i] <- paste(as.character(round(means_ASD[i],2)),
                 " (", as.character(round(sd_ASD[i], 2)), ")",
                 " (", as.character(round(min_ASD[i], 2)), "-",
                 as.character(round(max_ASD[i], 2)), ")")
}

#Create matrix of results
fig1C <- matrix(ASD_stats, nrow=7)

#Add column names and rownames
rownames(fig1C) <- colnames(asd_data)
colnames(fig1C) <- "ASD Mean (SD) (Range)"
```

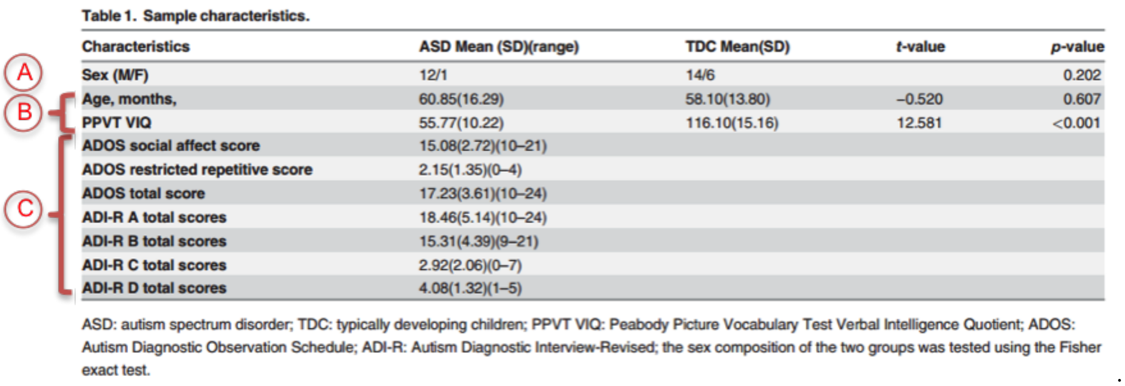


Table 1. Sample characteristics.

Characteristics	ASD Mean (SD)(range)	TDC Mean(SD)	*t*-value	*p*-value
Sex (M/F)	12/1	14/6		0.202
Age, months,	60.85(16.29)	58.10(13.80)	–0.520	0.607
PPVT VIQ	55.77(10.22)	116.10(15.16)	12.581	<0.001
ADOS social affect score	15.08(2.72)(10–21)			
ADOS restricted repetitive score	2.15(1.35)(0–4)			
ADOS total score	17.23(3.61)(10–24)			
ADI-R A total scores	18.46(5.14)(10–24)			
ADI-R B total scores	15.31(4.39)(9–21)			
ADI-R C total scores	2.92(2.06)(0–7)			
ADI-R D total scores	4.08(1.32)(1–5)			

ASD: autism spectrum disorder; TDC: typically developing children; PPVT VIQ: Peabody Picture Vocabulary Test Verbal Intelligence Quotient; ADOS: Autism Diagnostic Observation Schedule; ADI-R: Autism Diagnostic Interview-Revised; the sex composition of the two groups was tested using the Fisher exact test.

Part A

```
#Output part A
knitr::kable(fig1A)
```

	ASD	TDC	p-value
Sex M/F	12 / 1	14 / 6	0.202

Part B

```
#Output part B
knitr::kable(fig1B)
```

	ASD Mean(SD)	TDC Mean(SD)	t value	p-value
Age	60.85 (16.29)	58.1 (13.8)	0.52	0.61
PPT-VIQ	55.769 (10.22)	116.1 (15.15)	-12.58	1.02e-13

Part C

```
#Output part C
knitr::kable(fig1C)
```

	ASD Mean (SD) (Range)
ADOS_social_affect	15.08 (2.72) (10 - 21)
ADOS_restrictive_repetitive	2.15 (1.34) (0 - 4)
ADOStotalscore	17.23 (3.61) (10 - 24)
ADI_R_A	18.46 (5.14) (10 - 24)
ADI_R_B	15.31 (4.39) (9 - 21)
ADI_R_C	2.92 (2.06) (0 - 7)
ADI_R_D	4.08 (1.32) (1 - 5)

Table 2 Replication: Key Study Findings

The authors' descriptions of the statistical methods used in the analysis in Table 2 is rather sparse: "One-way ANOVA indicated that, in Part I of the study, there was no statistical difference in the fixation duration on DSIs or DGIs between the two groups… But in Part II, significant differences were found in fixation duration on DSIs, that met the Bonferroni-correction threshold of <0.025)."

Table 2 reports a comparison of eye tracking metrics across the ASD and TDC groups using the t-test and one-way ANOVA to compare means across the two groups. In this study, a t-test was used to compare "total fixation duration" across the ASD and TDC groups for both Part 1 and Part 2. Interestingly, a t-test was completed assuming equal variance for Part 1 (student's t-test), but the test for Part 2 did not assume equal variance (Welch's t-test). We do not see a reason for these variables to be treated differently, so we assume this may have been unintentional. Additionally, it is surprising that the authors chose to use a t-test for two of the items, but one-way ANOVA for the others when there were no more than two groups for any of the items.

For 8 of the variables in Table 2 (Part 1: Gaze duration to DSIs, Gaze duration to DGIs,Time of first fixation to DSIs, Time of first fixation to DGIs, and Part 2: Gaze duration to DSIs, Gaze duration to DGIs,Time of first fixation to DSIs, Time of first fixation to DGIs), the data was scaled by divinding by 1000. This affects the mean and standard deviation, but does not affect the F statistic or p-value. We can assume that their clock recorded miliseconds rather than seconds for these items, but it is not clear why this calculation was needed. To match values in this table, we have also performed this calculation.

Our values match the authors' in table two, with the excption that the two t-test items have the same value, but a negative rather than a positive sign. We believe this is because the order of the groups in our t-tests must have been opposite the order the author's used.

Note: The Part 1 values are named beginning with a *P_* and the Part 2 values are named beginning with an *S_*

```
#Figure 2 Table
#Select fields needed
fig2data <- proj_data %>%
  select(group, P_total.fixationduration, P_DSIs.fixationduration, P_DGIs.fixationduration,
         P_percentageof.fixationduration.DSIs, P_percentage.of.fixationduration.DGIs,
         P_number.on.DSIs, P_number.on.DGIs, P_gazeduration.to.DSIs.scale,
         P_gazeduration.to.DGIs.scale, P_time.of.first.fixation.to.DSIs.scale,
         P_time.of.first.fixation.to.DGIs.scale, S_total.fixationduration,
         S_fixationduration.on.DSIs, S_fixationduration.on.DGIs,
         S_percentage.of.fixationduration.on.DSIs, S_percentage.of.fixationduration.on.DGIs,
         S_number.on.DSIs, S_number.on.DGIs, S_gazeduration.to.DSIs.scale,
         S_gazeduration.to.DGIs.scale, S_time.of.first.fixation.to.DSIs.scale,
         S_time.of.first.fixation.to.DGIs.scale)

#Calculate means and standard deviations for each field
means_fig2 <- fig2data %>%
  group_by(group) %>%
  summarise_each(funs(mean))
```

```
`summarise_each()` is deprecated.
Use `summarise_all()`, `summarise_at()` or `summarise_if()` instead.
To map `funs` over all variables, use `summarise_all()`
```

```
sd_fig2 <- fig2data %>%
  group_by(group) %>%
  summarise_each(funs(sd))
```

```
`summarise_each()` is deprecated.
Use `summarise_all()`, `summarise_at()` or `summarise_if()` instead.
To map `funs` over all variables, use `summarise_all()`
```

```
#Arrange in a table
numStats <- ncol(means_fig2)
fig2Mat <- matrix(character(4*(numStats-1)), ncol=4)
```

```r
#For each measurement
for (i in 2:numStats) {
  #Output mean and standard deviation for ASD group
 fig2Mat[i-1,1] <- paste(as.character(round(means_fig2[1,i],2)),
                 " (", as.character(round(sd_fig2[1, i], 3)), ")")
 #Output mean and standard deviation for TDC group
 fig2Mat[i-1,2] <- paste(as.character(round(means_fig2[2, i],2)),
               " (", as.character(round(sd_fig2[2, i], 2)), ")")
 #Perform one way ANOVA
 aovModel <- summary(aov(fig2data[,i]~fig2data[,1]))
 #Output F-statistic
 fig2Mat[i-1, 3] <- as.character(round(aovModel[[1]][[1,4]], 2))
 #Output p-value
 fig2Mat[i-1, 4] <- as.character(round(aovModel[[1]][[1,5]], 2))
}
#Add row names and column names
rownames(fig2Mat) <- colnames(means_fig2)[2:numStats]
colnames(fig2Mat) <- list("ASD Mean(SD) (n=13)", "TDC Mean(SD) (n=20)", "F", "p Value")

#Perform t-tests for total fixation duration.  Note: for Part 1 only they used
#var.equal = TRUE
pt1Total <- t.test(P_total.fixationduration~group, data=fig2data, var.equal=TRUE)
pt2Total <- t.test(S_total.fixationduration~group, data=fig2data)

#Convert results matrix to data frame
fig2 <- as.data.frame(fig2Mat)

#Update F-statistics and p-values for two t-tests
fig2$F <- as.character(fig2$F)
fig2$`p Value` <- as.character(fig2$`p Value`)
fig2["P_total.fixationduration", "F"] = paste("t value = ", as.character(round(pt1Total$statistic, 2)))
fig2["S_total.fixationduration", "F"] = paste("t value = ", as.character(round(pt2Total$statistic, 2)))
fig2["P_total.fixationduration", "p Value"] = as.character(round(pt1Total$p.value, 2))
fig2["S_total.fixationduration", "p Value"] = as.character(round(pt2Total$p.value, 2))
```

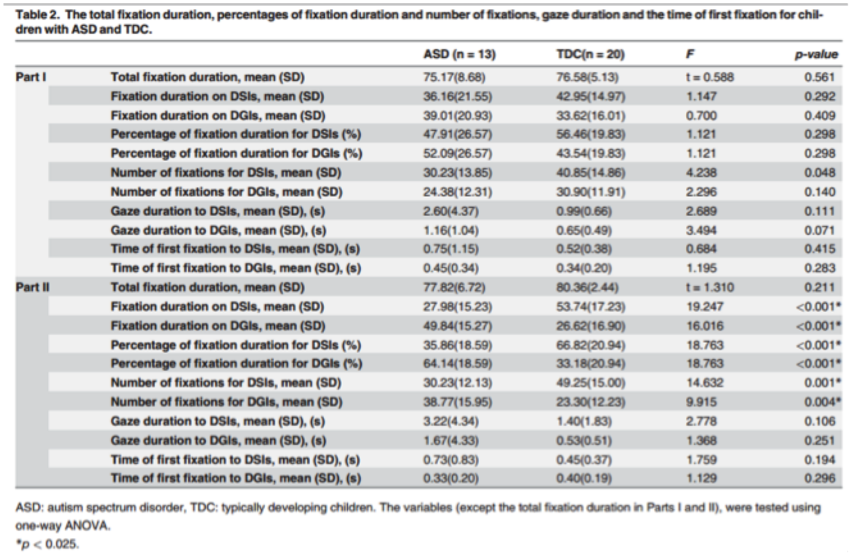


Table 2. The total fixation duration, percentages of fixation duration and number of fixations, gaze duration and the time of first fixation for children with ASD and TDC.

		ASD (n = 13)	TDC(n = 20)	*F*	*p-value*
Part I	Total fixation duration, mean (SD)	75.17(8.68)	76.58(5.13)	t = 0.588	0.561
	Fixation duration on DSIs, mean (SD)	36.16(21.55)	42.95(14.97)	1.147	0.292
	Fixation duration on DGIs, mean (SD)	39.01(20.93)	33.62(16.01)	0.700	0.409
	Percentage of fixation duration for DSIs (%)	47.91(26.57)	56.46(19.83)	1.121	0.298
	Percentage of fixation duration for DGIs (%)	52.09(26.57)	43.54(19.83)	1.121	0.298
	Number of fixations for DSIs, mean (SD)	30.23(13.85)	40.85(14.86)	4.238	0.048
	Number of fixations for DGIs, mean (SD)	24.38(12.31)	30.90(11.91)	2.296	0.140
	Gaze duration to DSIs, mean (SD), (s)	2.60(4.37)	0.99(0.66)	2.689	0.111
	Gaze duration to DGIs, mean (SD), (s)	1.16(1.04)	0.65(0.49)	3.494	0.071
	Time of first fixation to DSIs, mean (SD), (s)	0.75(1.15)	0.52(0.38)	0.684	0.415
	Time of first fixation to DGIs, mean (SD), (s)	0.45(0.34)	0.34(0.20)	1.195	0.283
Part II	Total fixation duration, mean (SD)	77.82(6.72)	80.36(2.44)	t = 1.310	0.211
	Fixation duration on DSIs, mean (SD)	27.98(15.23)	53.74(17.23)	19.247	<0.001*
	Fixation duration on DGIs, mean (SD)	49.84(15.27)	26.62(16.90)	16.016	<0.001*
	Percentage of fixation duration for DSIs (%)	35.86(18.59)	66.82(20.94)	18.763	<0.001*
	Percentage of fixation duration for DGIs (%)	64.14(18.59)	33.18(20.94)	18.763	<0.001*
	Number of fixations for DSIs, mean (SD)	30.23(12.13)	49.25(15.00)	14.632	0.001*
	Number of fixations for DGIs, mean (SD)	38.77(15.95)	23.30(12.23)	9.915	0.004*
	Gaze duration to DSIs, mean (SD), (s)	3.22(4.34)	1.40(1.83)	2.778	0.106
	Gaze duration to DGIs, mean (SD), (s)	1.67(4.33)	0.53(0.51)	1.368	0.251
	Time of first fixation to DSIs, mean (SD), (s)	0.73(0.83)	0.45(0.37)	1.759	0.194
	Time of first fixation to DGIs, mean (SD), (s)	0.33(0.20)	0.40(0.19)	1.129	0.296

ASD: autism spectrum disorder, TDC: typically developing children. The variables (except the total fixation duration in Parts I and II), were tested using one-way ANOVA.

*$p < 0.025$.

```r
#Output Table 2
knitr::kable(fig2)
```

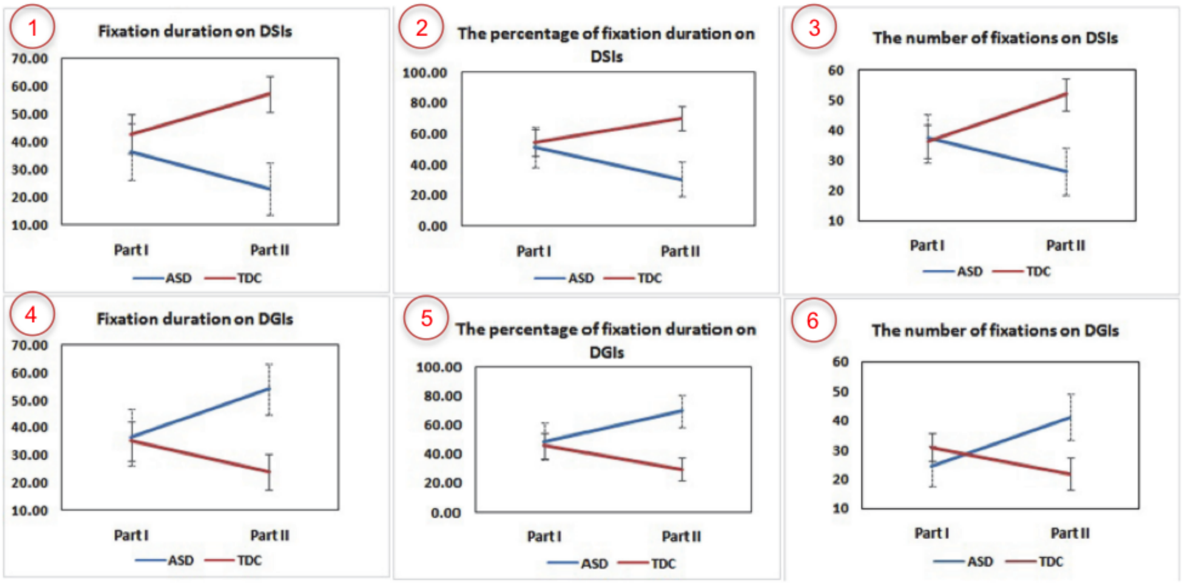
	ASD Mean(SD) (n=13)	TDC Mean(SD) (n=20)	F	p Value
P_total.fixationduration	75.17 (8.678)	76.58 (5.13)	t value = -0.59	0.56
P_DSIs.fixationduration	36.16 (21.554)	42.95 (14.97)	1.15	0.29
P_DGIs.fixationduration	39.01 (20.93)	33.62 (16.01)	0.7	0.41
P_percentageof.fixationduration.DSIs	47.91 (26.565)	56.46 (19.83)	1.12	0.3
P_percentage.of.fixationduration.DGIs	52.09 (26.565)	43.54 (19.83)	1.12	0.3
P_number.on.DSIs	30.23 (13.851)	40.85 (14.86)	4.24	0.05
P_number.on.DGIs	24.38 (12.312)	30.9 (11.91)	2.3	0.14
P_gazeduration.to.DSIs.scale	2.6 (4.369)	0.99 (0.66)	2.69	0.11
P_gazeduration.to.DGIs.scale	1.16 (1.041)	0.65 (0.49)	3.49	0.07
P_time.of.first.fixation.to.DSIs.scale	0.75 (1.148)	0.52 (0.38)	0.68	0.41
P_time.of.first.fixation.to.DGIs.scale	0.45 (0.343)	0.34 (0.2)	1.2	0.28
S_total.fixationduration	77.81 (6.725)	80.36 (2.44)	t value = -1.31	0.21
S_fixationduration.on.DSIs	27.98 (15.228)	53.74 (17.23)	19.25	0
S_fixationduration.on.DGIs	49.84 (15.266)	26.62 (16.9)	16.01	0
S_percentage.of.fixationduration.on.DSIs	35.86 (18.589)	66.82 (20.94)	18.76	0
S_percentage.of.fixationduration.on.DGIs	64.14 (18.589)	33.18 (20.94)	18.76	0
S_number.on.DSIs	30.23 (12.125)	49.25 (15)	14.63	0
S_number.on.DGIs	38.77 (15.949)	23.3 (12.23)	9.92	0
S_gazeduration.to.DSIs.scale	3.22 (4.344)	1.4 (1.83)	2.78	0.11
S_gazeduration.to.DGIs.scale	1.67 (4.332)	0.53 (0.51)	1.37	0.25
S_time.of.first.fixation.to.DSIs.scale	0.73 (0.833)	0.45 (0.37)	1.76	0.19
S_time.of.first.fixation.to.DGIs.scale	0.33 (0.202)	0.4 (0.19)	1.13	0.3

Figure 2 Replication: Estimated marginal means

The author's description of Figure 2 comes from the figure title "Estimated marginal means for fixation duration, the percentage of fixation duration and the number of fixations on DSIs and DGIs by the ASD and TDC groups in Part I and Part II. PPVT VIQ is co-varied."

We spent considerable time attempting to replicate Figure 2, but it was difficult due to missing details about how these graphs were created. We used the library `lsmeans` to calculate marginal means using a linear model to control for PPVT VIQ. We tried various combinations of predictors and interactions in the linear model, but weren't able to replicate the values exactly. We even tried plotting the raw means without any covariates, which didn't match either. Finally, we noticed that the values for part 1 were very close, but the numbers for Part 2 were different, suggesting perhaps the authors utilized two linear models (one for part 1, and one for part 2). We tried that and it seems the closest to the paper's graphs.

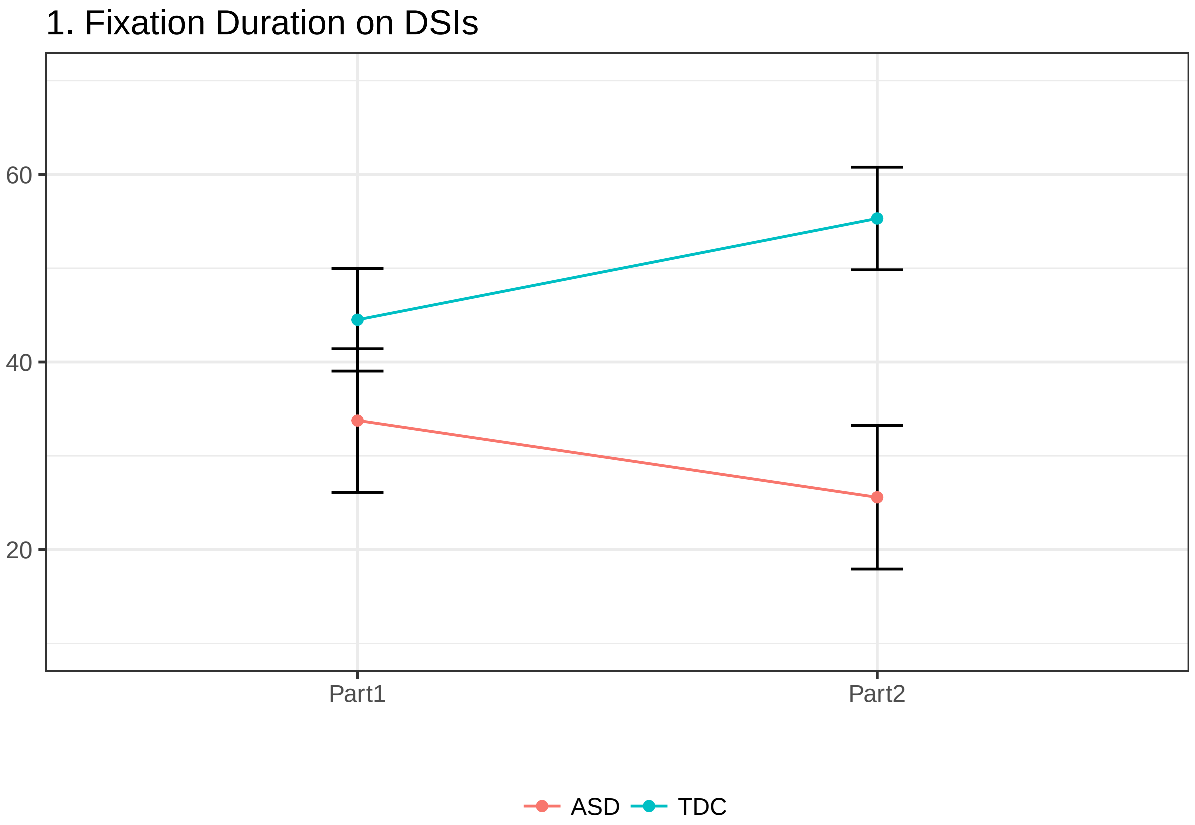
We interpret "PPVT VIQ is co-varied" in the description of Figure 2 to mean that they controlled for PPVT VIQ, which means they would hold this value constant across all groups. This does not seem entirely appropriate, as in Table 1, the authors note that there is considerable difference in PPVT VIQ across the ASD and TDC groups.



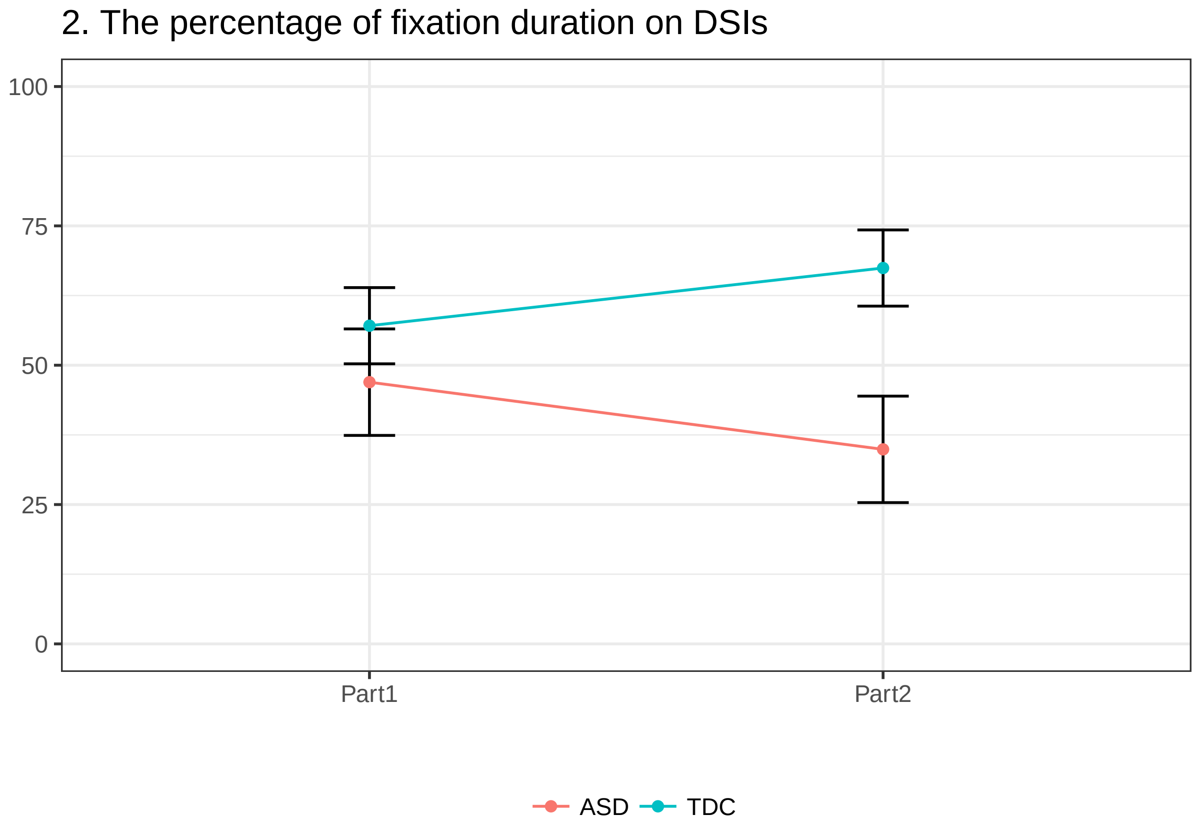
First try: PPVT.VIQ held constant across all groups, group and part interact.

These graphs were close, but didn't match the paper's exactly.

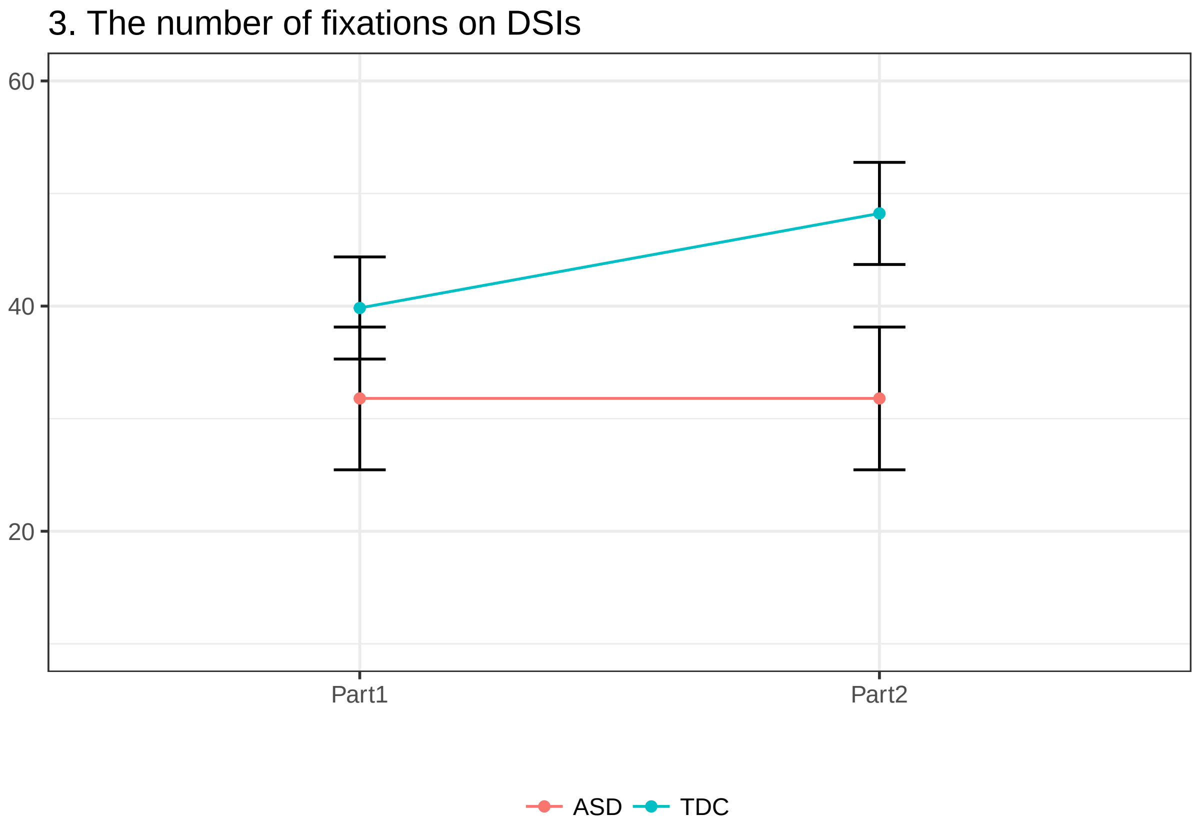
```
#Using lsmeans, calculate marginal means with PPVT.VIQ as a covariate.
#Allow group and part to interact, but not PPVT.VIQ
#Not correct assumption since PPVT.VIQ varies greatly with group, but this is what matches
#paper best (still not exact)
#Plot 1: Fixation Duration on DSIs

#Select and Reformat data
fig3data <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_DSIs.fixationduration, S_fixationduration.on.DSIs) %>%
  dplyr::rename(Part1 = P_DSIs.fixationduration, Part2 = S_fixationduration.on.DSIs) %>%
  gather(part, dur_time, Part1:Part2)

#Generate linear model
lmodel1 <- lm(dur_time~PPVT.VIQ + group*part, data=fig3data)
refGrid <- ref.grid(lmodel1)

#Calculate marginal means
marMeans <- lsmeans(refGrid,"part", by="group")
marMeansC <- cld(marMeans)

#Plot marginal means
ggplot(marMeansC, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(10,70)) +
  #scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("1. Fixation Duration on DSIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



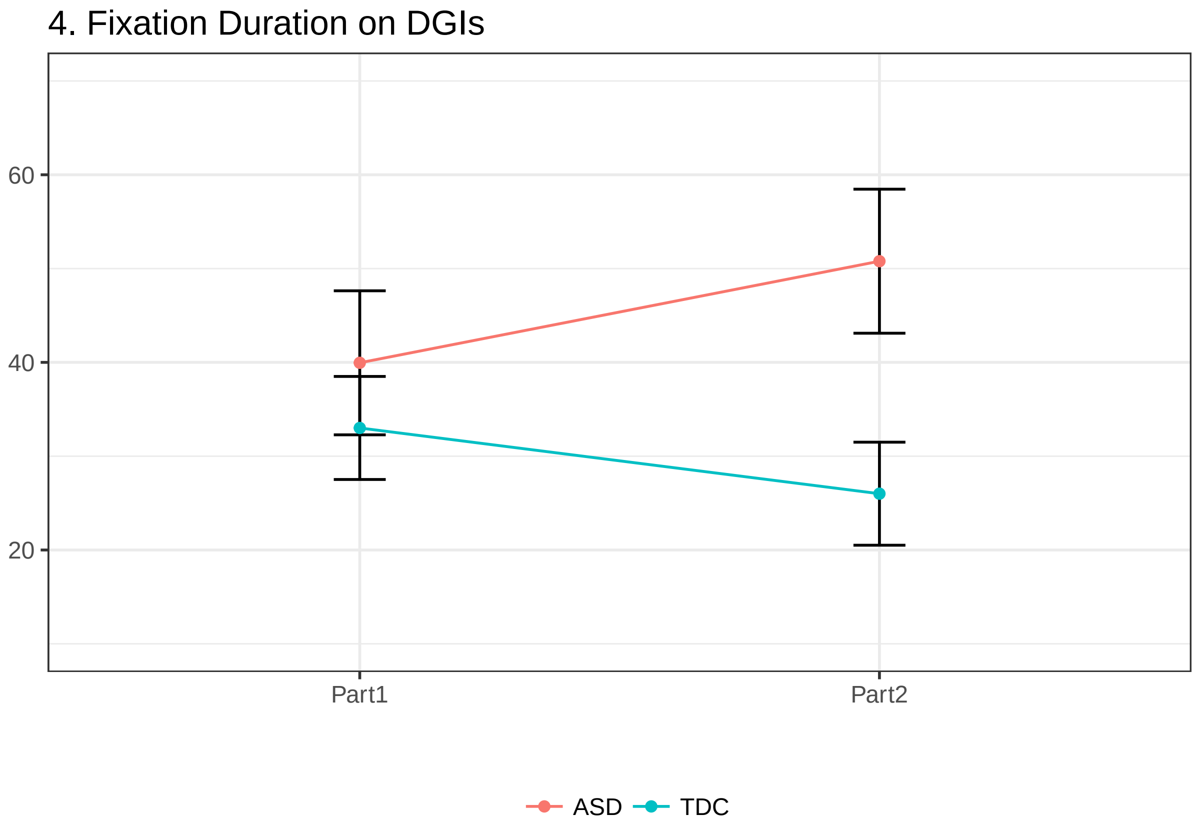
```
#Plot 2: The percentage of fixation duration on DSIs
#Select and reformat data
fig3data <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_percentageof.fixationduration.DSIs,
                S_percentage.of.fixationduration.on.DSIs) %>%
  dplyr::rename(Part1 = P_percentageof.fixationduration.DSIs,
                Part2 = S_percentage.of.fixationduration.on.DSIs) %>%
  gather(part, dur_time, Part1:Part2)

#Generate linear model
lmodel1 <- lm(dur_time~PPVT.VIQ + group*part, data=fig3data)
refGrid <- ref.grid(lmodel1)
#Calculate marginal means
marMeans <- lsmeans(refGrid,"part", by="group")
marMeansC <- cld(marMeans)
#Plot marginal means
ggplot(marMeansC, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(0,100)) +
  #scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("2. The percentage of fixation duration on DSIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



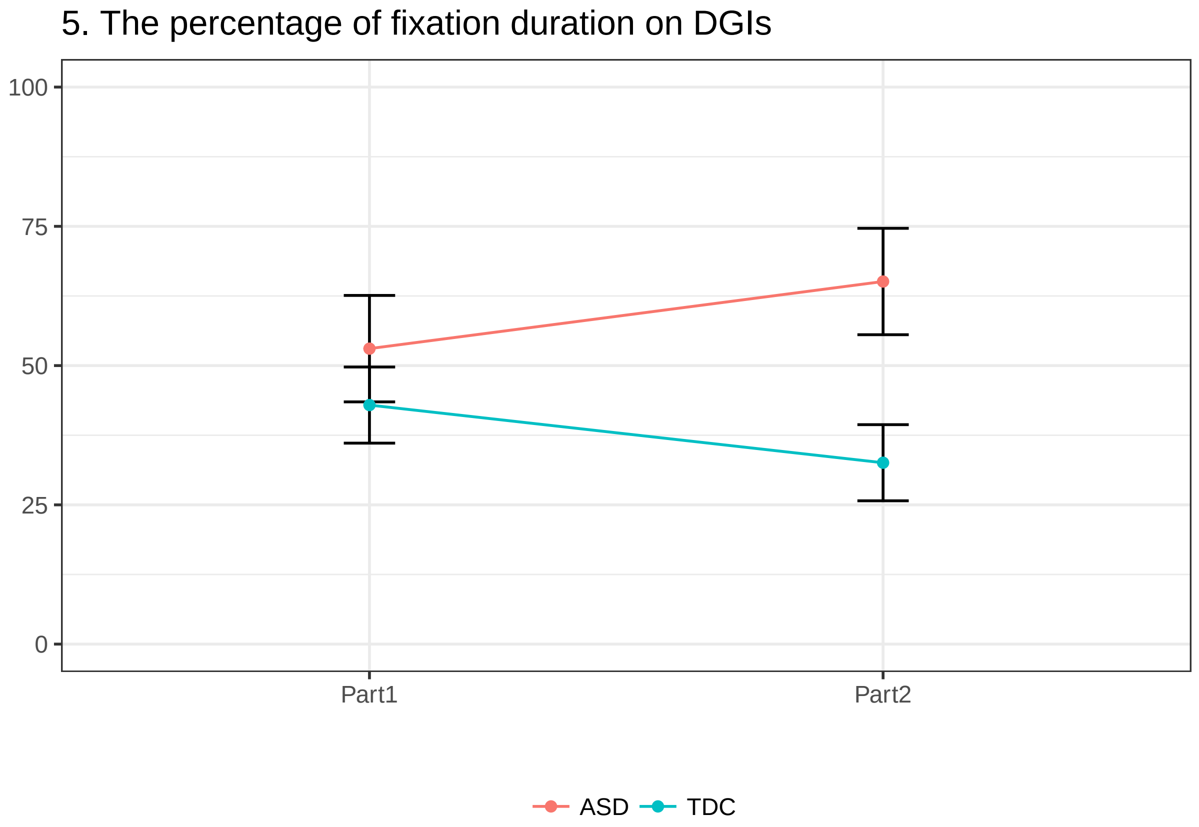
```
#Plot 3: The number of fixations on DSIs
#Select and reformat data
fig3data <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_number.on.DSIs, S_number.on.DSIs) %>%
  dplyr::rename(Part1 = P_number.on.DSIs, Part2 = S_number.on.DSIs) %>%
  gather(part, dur_time, Part1:Part2)
#Generate linear model
lmodel1 <- lm(dur_time~PPVT.VIQ + group*part, data=fig3data)
refGrid <- ref.grid(lmodel1)
#Calculate marginal means
marMeans <- lsmeans(refGrid,"part", by="group")
marMeansC <- cld(marMeans)
#Plot marginal means
ggplot(marMeansC, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(10,60)) +
  #scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("3. The number of fixations on DSIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



```
#Plot 4: Fixation Duration on DGIs
#Select and reformat data
fig3data <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_DGIs.fixationduration, S_fixationduration.on.DGIs) %>%
  dplyr::rename(Part1 = P_DGIs.fixationduration, Part2 = S_fixationduration.on.DGIs) %>%
  gather(part, dur_time, Part1:Part2)
#Generate linear model
lmodel1 <- lm(dur_time~PPVT.VIQ + group*part, data=fig3data)
refGrid <- ref.grid(lmodel1)
#Calculate marginal means
marMeans <- lsmeans(refGrid,"part", by="group")
marMeansC <- cld(marMeans)
#Plot marginal means
ggplot(marMeansC, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(10,70)) +
  #scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("4. Fixation Duration on DGIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```

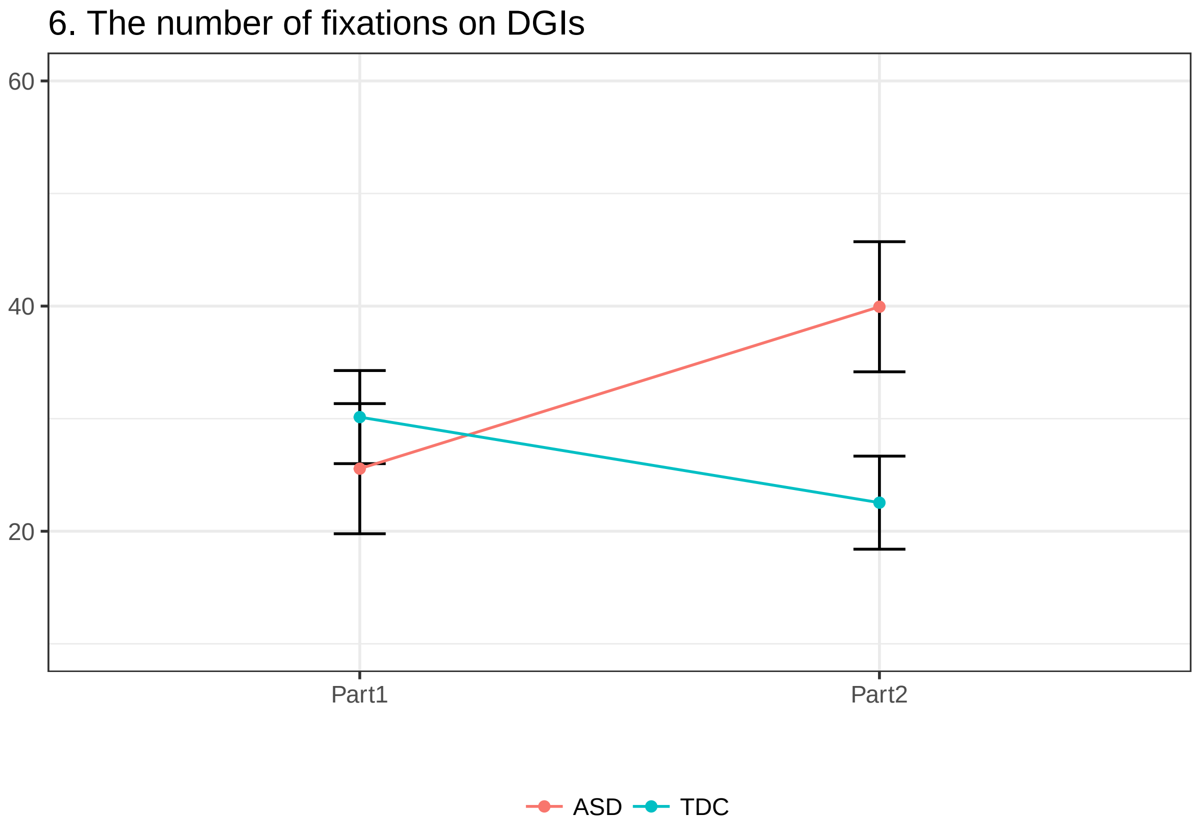


```r
#Plot 5: The percentage of fixation duration on DGIs
#Select and reformat data
fig3data <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_percentage.of.fixationduration.DGIs,
                S_percentage.of.fixationduration.on.DGIs) %>%
  dplyr::rename(Part1 = P_percentage.of.fixationduration.DGIs,
                Part2 = S_percentage.of.fixationduration.on.DGIs) %>%
  gather(part, dur_time, Part1:Part2)
#Generate linear model
lmodel1 <- lm(dur_time~PPVT.VIQ + group*part, data=fig3data)
refGrid <- ref.grid(lmodel1)
#Calculate marginal means
marMeans <- lsmeans(refGrid,"part", by="group")
marMeansC <- cld(marMeans)
#Plot marginal means
ggplot(marMeansC, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(0,100)) +
  #scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("5. The percentage of fixation duration on DGIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



5. The percentage of fixation duration on DGIs

```
#Plot 6: The number of fixations on DGIs
#Select and reformat data
fig3data <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_number.on.DGIs, S_number.on.DGIs) %>%
  dplyr::rename(Part1 = P_number.on.DGIs, Part2 = S_number.on.DGIs) %>%
  gather(part, dur_time, Part1:Part2)
#Generate linear model
lmodel1 <- lm(dur_time~PPVT.VIQ + group*part, data=fig3data)
#Calculate marginal means
refGrid <- ref.grid(lmodel1)
marMeans <- lsmeans(refGrid,"part", by="group")
marMeansC <- cld(marMeans)
#Plot marginal means
ggplot(marMeansC, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(10,60)) +
  #scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("6. The number of fixations on DGIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```

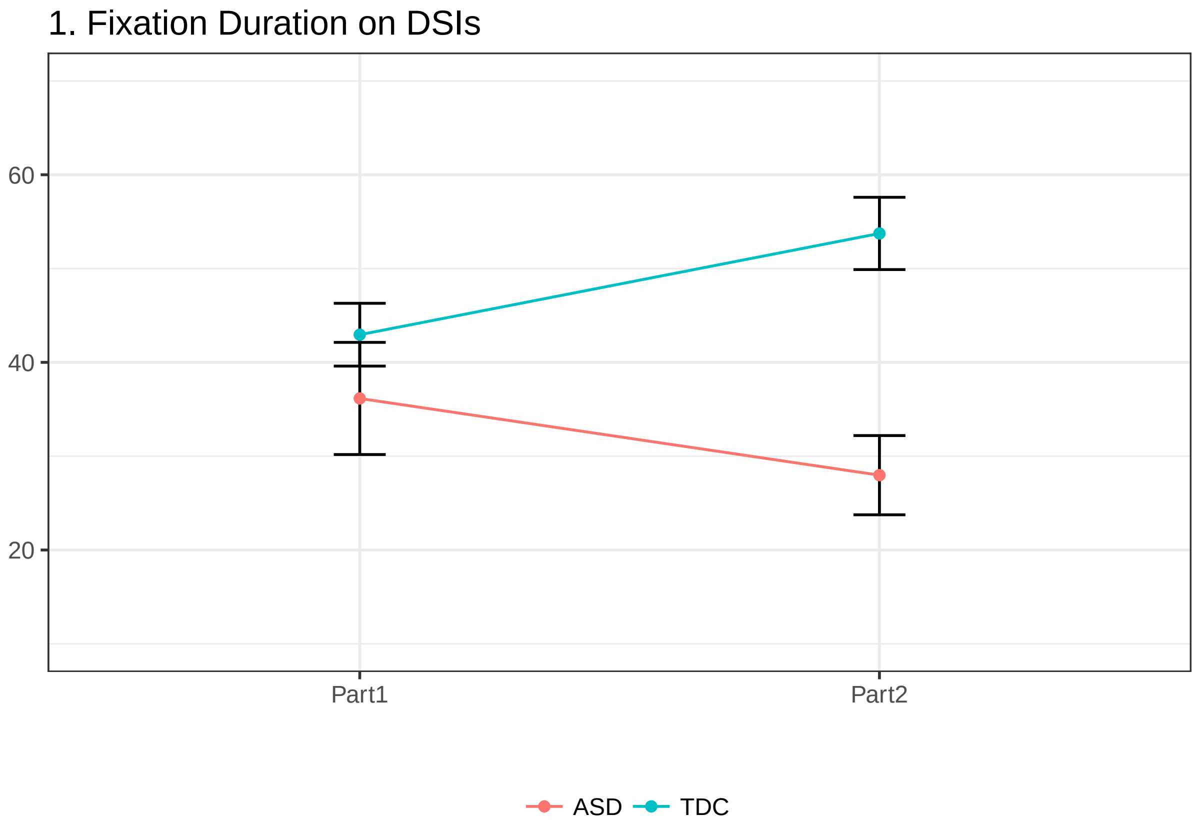


Second Try: Plot raw means without covariates.

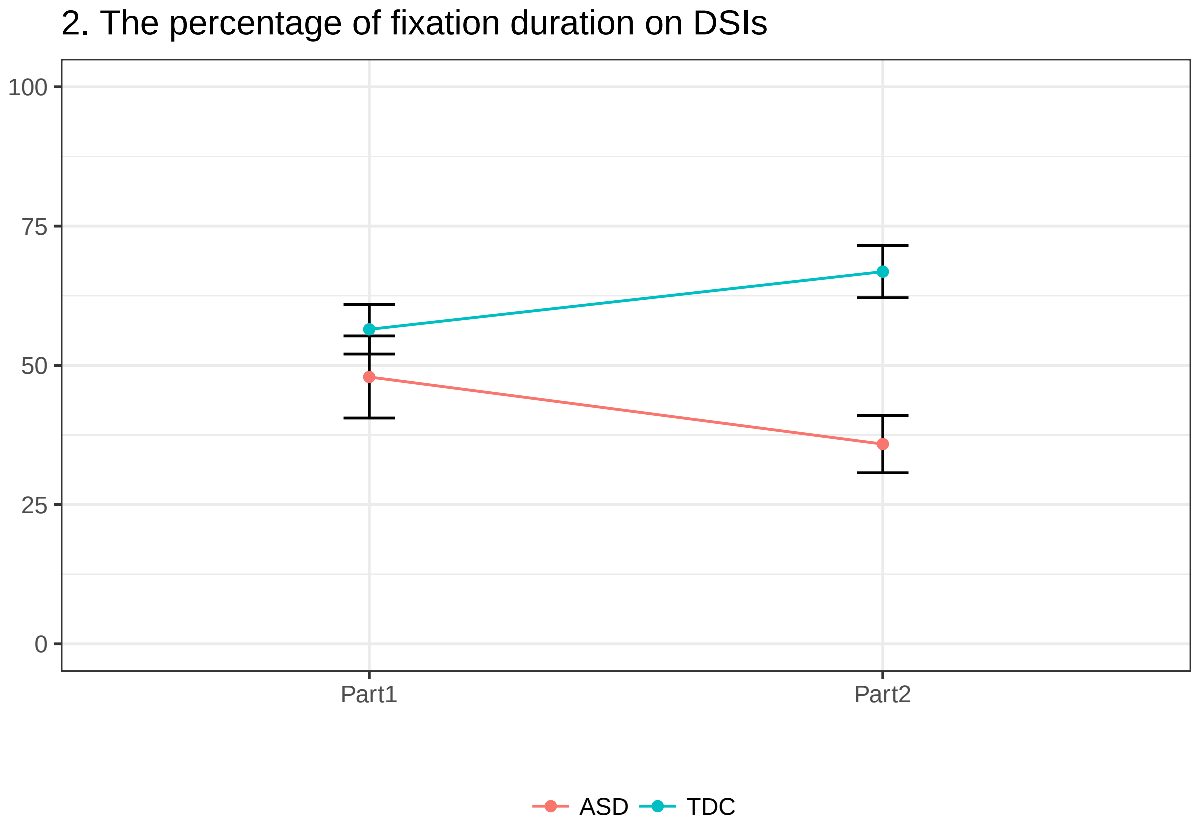
These graphs were close to the paper's, but didn't match exactly.

```
#From R Cookbook
#Used for plotting raw means
# http://www.cookbook-r.com/Graphs/Plotting_means_and_error_bars_(ggplot2)/
## Gives count, mean, standard deviation, standard error of the mean, and confidence interval (default 9
##   data: a data frame.
##   measurevar: the name of a column that contains the variable to be summariezed
##   groupvars: a vector containing names of columns that contain grouping variables
##   na.rm: a boolean that indicates whether to ignore NA's
##   conf.interval: the percent range of the confidence interval (default is 95%)
summarySE <- function(data=NULL, measurevar, groupvars=NULL, na.rm=FALSE,
                      conf.interval=.95, .drop=TRUE) {


    # New version of length which can handle NA's: if na.rm==T, don't count them
    length2 <- function (x, na.rm=FALSE) {
        if (na.rm) sum(!is.na(x))
        else       length(x)
    }

    # This does the summary. For each group's data frame, return a vector with
    # N, mean, and sd
    datac <- ddply(data, groupvars, .drop=.drop,
      .fun = function(xx, col) {
        c(N    = length2(xx[[col]], na.rm=na.rm),
          mean = mean   (xx[[col]], na.rm=na.rm),
          sd    = sd     (xx[[col]], na.rm=na.rm)
        )
      },
      measurevar
    )

    # Rename the "mean" column
    datac <- plyr::rename(datac, c("mean" = measurevar))

    datac$se <- datac$sd / sqrt(datac$N)  # Calculate standard error of the mean

    # Confidence interval multiplier for standard error
    # Calculate t-statistic for confidence interval:
    # e.g., if conf.interval is .95, use .975 (above/below), and use df=N-1
    ciMult <- qt(conf.interval/2 + .5, datac$N-1)
    datac$ci <- datac$se * ciMult

    return(datac)
}
```

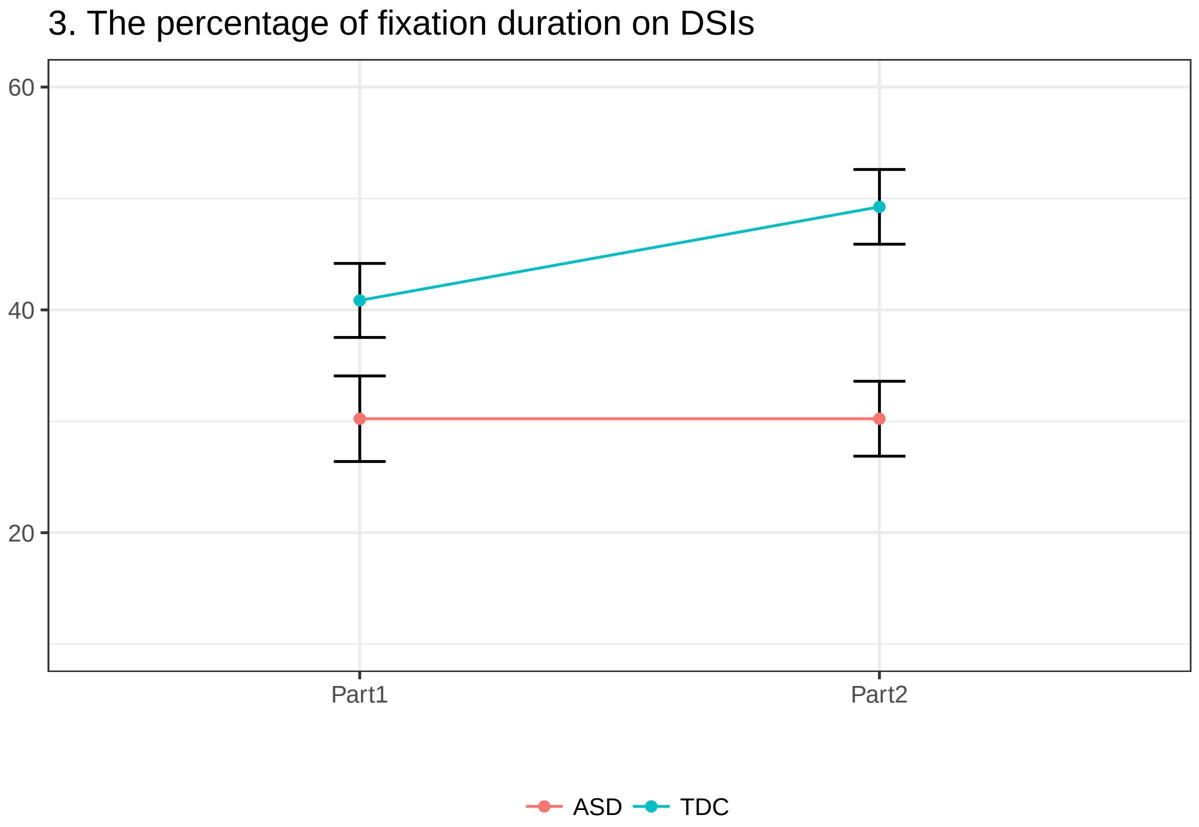
```
#Plots of raw means without covariates
#Plot 1
#Select and reformat data
fig2data %>%
  dplyr::select(group, P_DSIs.fixationduration, S_fixationduration.on.DSIs) %>%
  dplyr::rename(Part1 = P_DSIs.fixationduration, Part2 = S_fixationduration.on.DSIs) %>%
  gather(part, dur_time, Part1:Part2) %>%
  #Calculate means and standard error
  summarySE(measurevar="dur_time", groupvars=c("group","part")) %>%
  #Plot means and standar error
  ggplot(aes(x=part, y=dur_time, colour=group, group=group)) +
  geom_errorbar(aes(ymin=dur_time-se, ymax=dur_time+se), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(10,70)) +
 # scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("1. Fixation Duration on DSIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



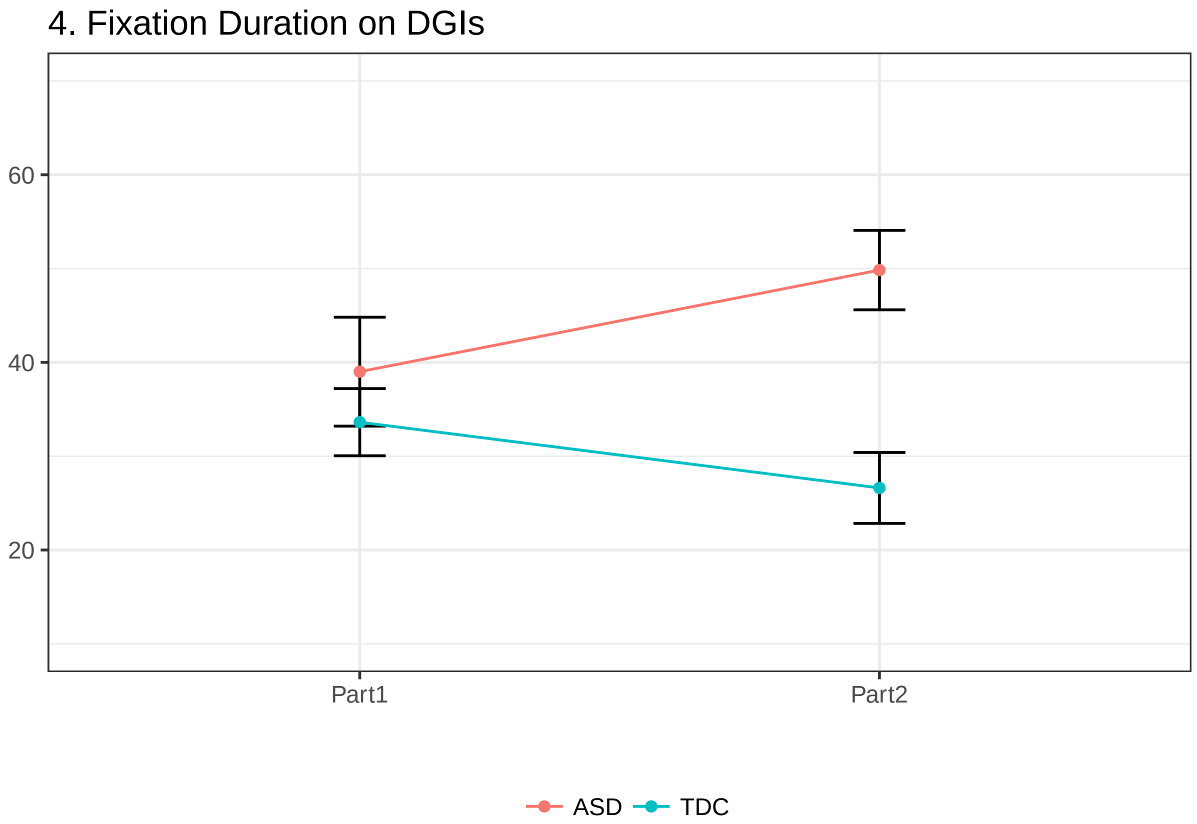
```
#Plot 2
#Select and reformat data
fig2data %>%
  dplyr::select(group, P_percentageof.fixationduration.DSIs,
                 S_percentage.of.fixationduration.on.DSIs) %>%
  dplyr::rename(Part1 = P_percentageof.fixationduration.DSIs,
                 Part2 = S_percentage.of.fixationduration.on.DSIs) %>%
  gather(part, dur_time, Part1:Part2) %>%
  #Calculate means and standard error
  summarySE(measurevar="dur_time", groupvars=c("group","part")) %>%
  #Plot means and standard error
  ggplot(aes(x=part, y=dur_time, colour=group, group=group)) +
  geom_errorbar(aes(ymin=dur_time-se, ymax=dur_time+se), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(0,100)) +
 # scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("2. The percentage of fixation duration on DSIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



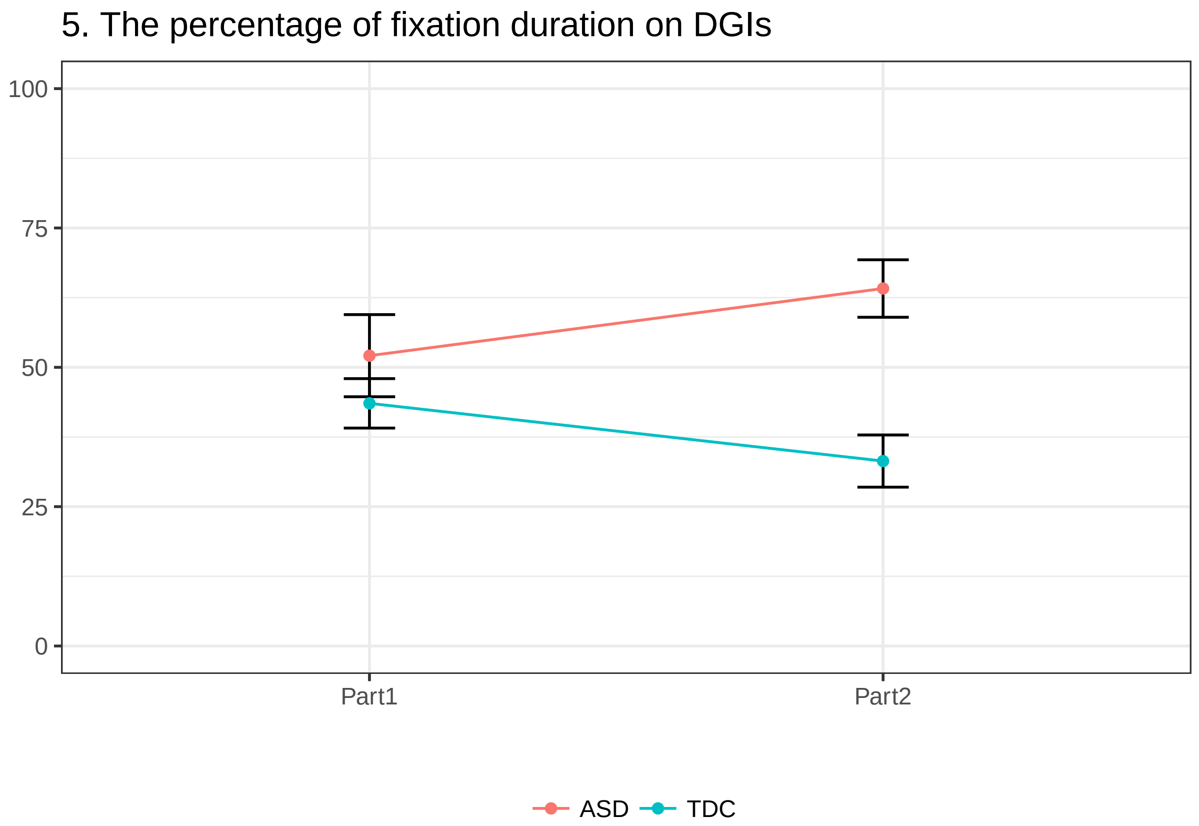
```
#Plot 3
#Select and reformat data
fig2data %>%
  dplyr::select(group, P_number.on.DSIs, S_number.on.DSIs) %>%
  dplyr::rename(Part1 = P_number.on.DSIs, Part2 = S_number.on.DSIs) %>%
  gather(part, dur_time, Part1:Part2) %>%
  #Calculate means and standard error
  summarySE(measurevar="dur_time", groupvars=c("group","part")) %>%
  #Plot means and standard error
  ggplot(aes(x=part, y=dur_time, colour=group, group=group)) +
  geom_errorbar(aes(ymin=dur_time-se, ymax=dur_time+se), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(10,60)) +
 # scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("3. The percentage of fixation duration on DSIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



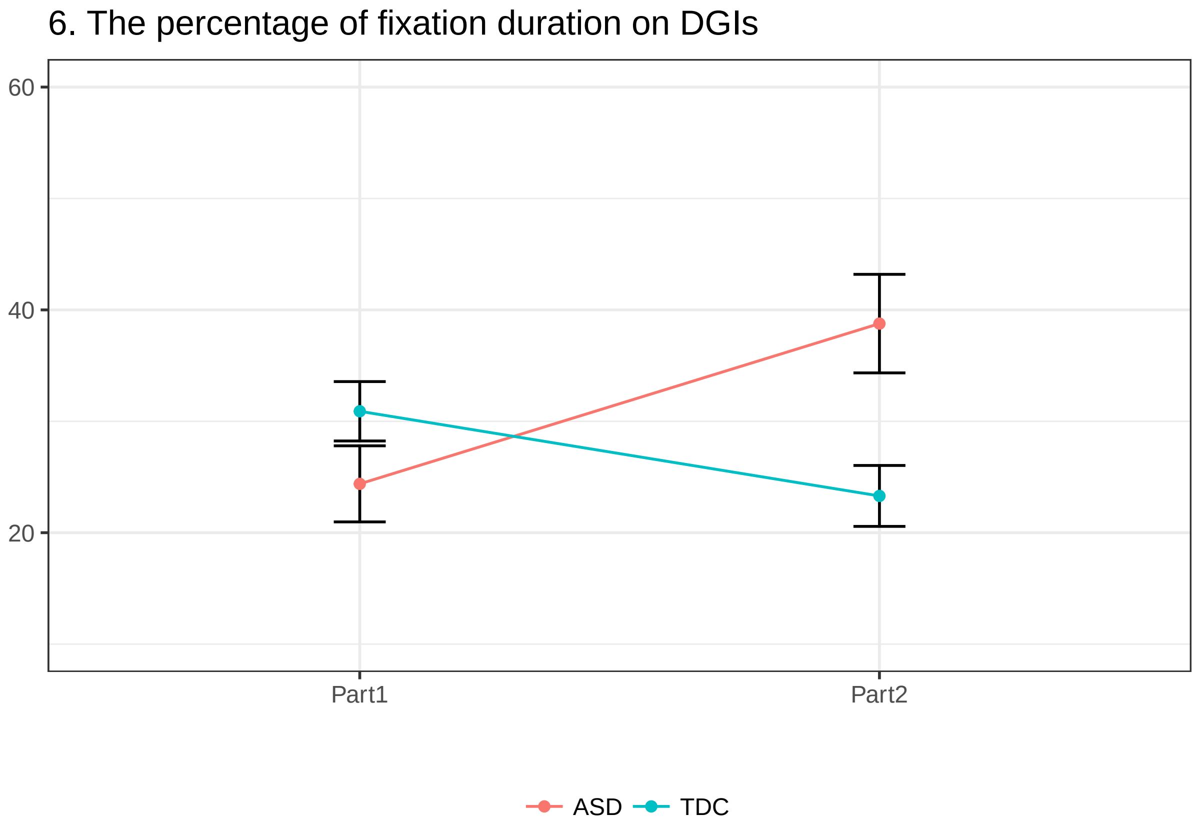
```
#Plot 4
#Select and reformat data
fig2data %>%
  dplyr::select(group, P_DGIs.fixationduration, S_fixationduration.on.DGIs) %>%
  dplyr::rename(Part1 = P_DGIs.fixationduration, Part2 = S_fixationduration.on.DGIs) %>%
  gather(part, dur_time, Part1:Part2) %>%
  #Calculate means and standard error
  summarySE(measurevar="dur_time", groupvars=c("group","part")) %>%
  #Plot means and standard error
  ggplot(aes(x=part, y=dur_time, colour=group, group=group)) +
  geom_errorbar(aes(ymin=dur_time-se, ymax=dur_time+se), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(10,70)) +
 # scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("4. Fixation Duration on DGIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



```
#Plot 5
#Select and reformat data
fig2data %>%
  dplyr::select(group, P_percentage.of.fixationduration.DGIs,
                S_percentage.of.fixationduration.on.DGIs) %>%
  dplyr::rename(Part1 = P_percentage.of.fixationduration.DGIs,
                Part2 = S_percentage.of.fixationduration.on.DGIs) %>%
  gather(part, dur_time, Part1:Part2) %>%
  #Calculate means and standard error
  summarySE(measurevar="dur_time", groupvars=c("group","part")) %>%
  #Plot means and standard error
  ggplot(aes(x=part, y=dur_time, colour=group, group=group)) +
  geom_errorbar(aes(ymin=dur_time-se, ymax=dur_time+se), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(0,100)) +
 # scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("5. The percentage of fixation duration on DGIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



```
#Plot 6
#Select and reformat data
fig2data %>%
  dplyr::select(group, P_number.on.DGIs, S_number.on.DGIs) %>%
  dplyr::rename(Part1 = P_number.on.DGIs, Part2 = S_number.on.DGIs) %>%
  gather(part, dur_time, Part1:Part2) %>%
  #Calculate means and standard error
  summarySE(measurevar="dur_time", groupvars=c("group","part")) %>%
  #Plot means and standard error
  ggplot(aes(x=part, y=dur_time, colour=group, group=group)) +
  geom_errorbar(aes(ymin=dur_time-se, ymax=dur_time+se), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  coord_cartesian(ylim=c(10,60)) +
 # scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("6. The percentage of fixation duration on DGIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```

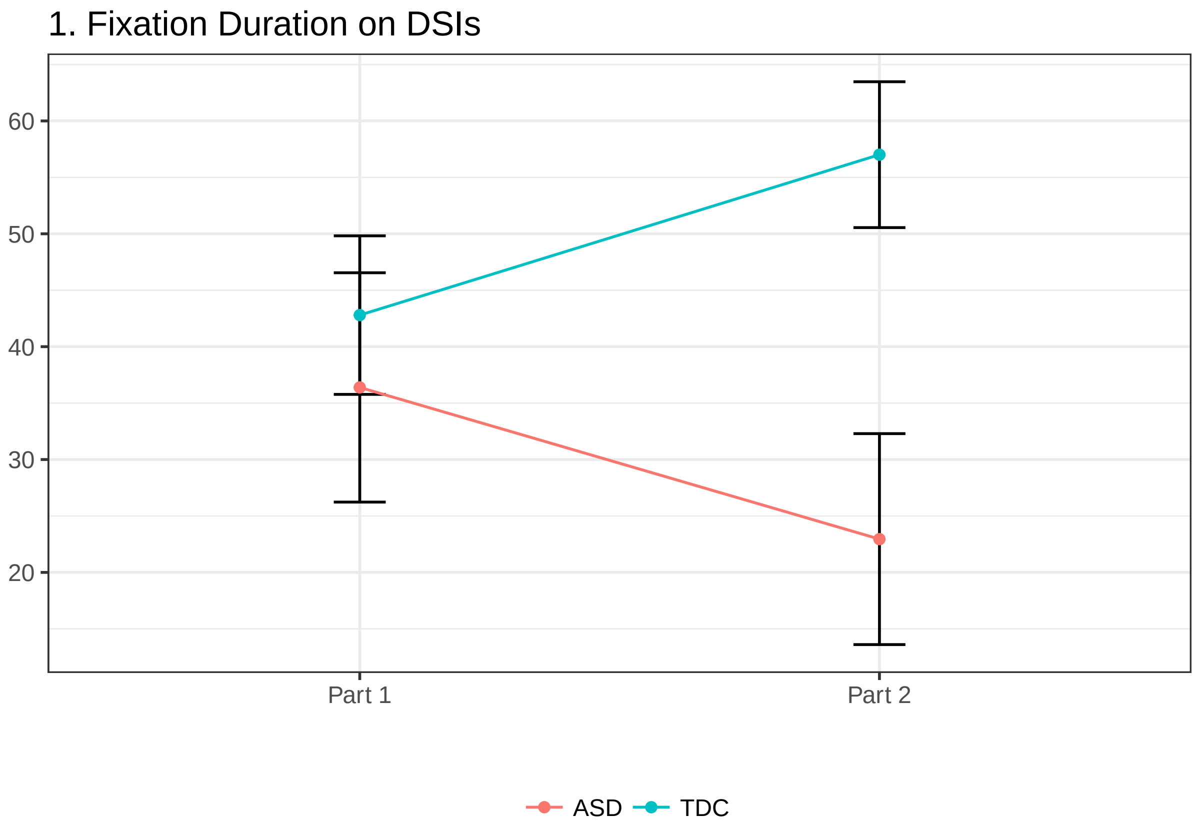


Third try: Generate linear models separately for parts 1 and 2.

This attempt seems the closest to the paper's.

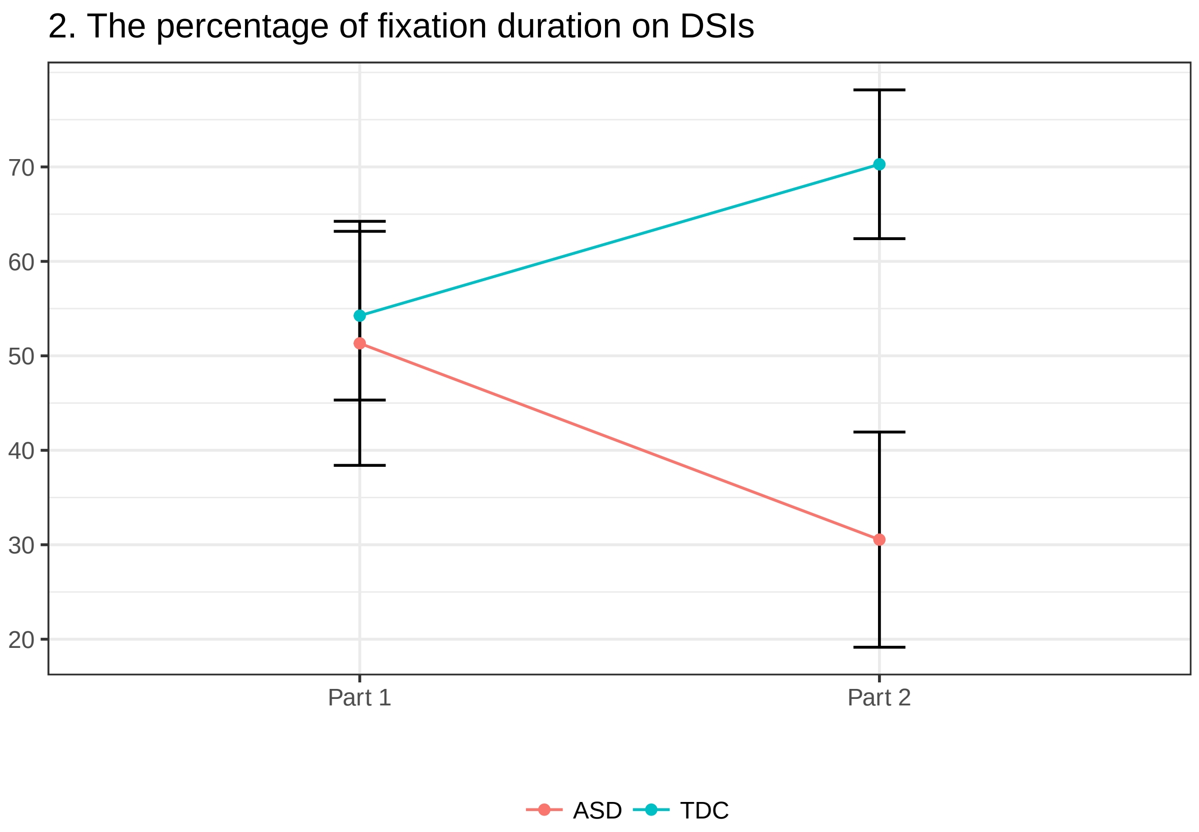
```
#Plot 1
#Do the Part 1 and Part 2 analyses separately, then combine for plot
#Did not match the paper's plot

#Select and reformat data
fig3data1 <- proj_data %>%
  select(group, PPVT.VIQ,P_DSIs.fixationduration, S_fixationduration.on.DSIs)
#Generate linear model for part 1
lmodel1 <- lm(P_DSIs.fixationduration~PPVT.VIQ + group, data=fig3data1)
refGrid1 <- ref.grid(lmodel1)
#Calculate marginal means for part 1
marMeans1 <- lsmeans(refGrid1, "group")
marMeans1c <- cld(marMeans1)
marMeans1c["part"] <- c("Part 1", "Part 1")
#Generate linear model for part 2
lmodel2 <- lm(S_fixationduration.on.DSIs~PPVT.VIQ + group, data=fig3data1)
refGrid2 <- ref.grid(lmodel2)
#Calculate marginal means for part 2
marMeans2 <- lsmeans(refGrid2, "group")
marMeans2c <- cld(marMeans2)
marMeans2c["part"] <- c("Part 2", "Part 2")

#Combine marginal means for both parts
marMeansc <- rbind(marMeans1c, marMeans2c)
```

```r
#Plot marginal means
ggplot(marMeansc, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("1. Fixation Duration on DSIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



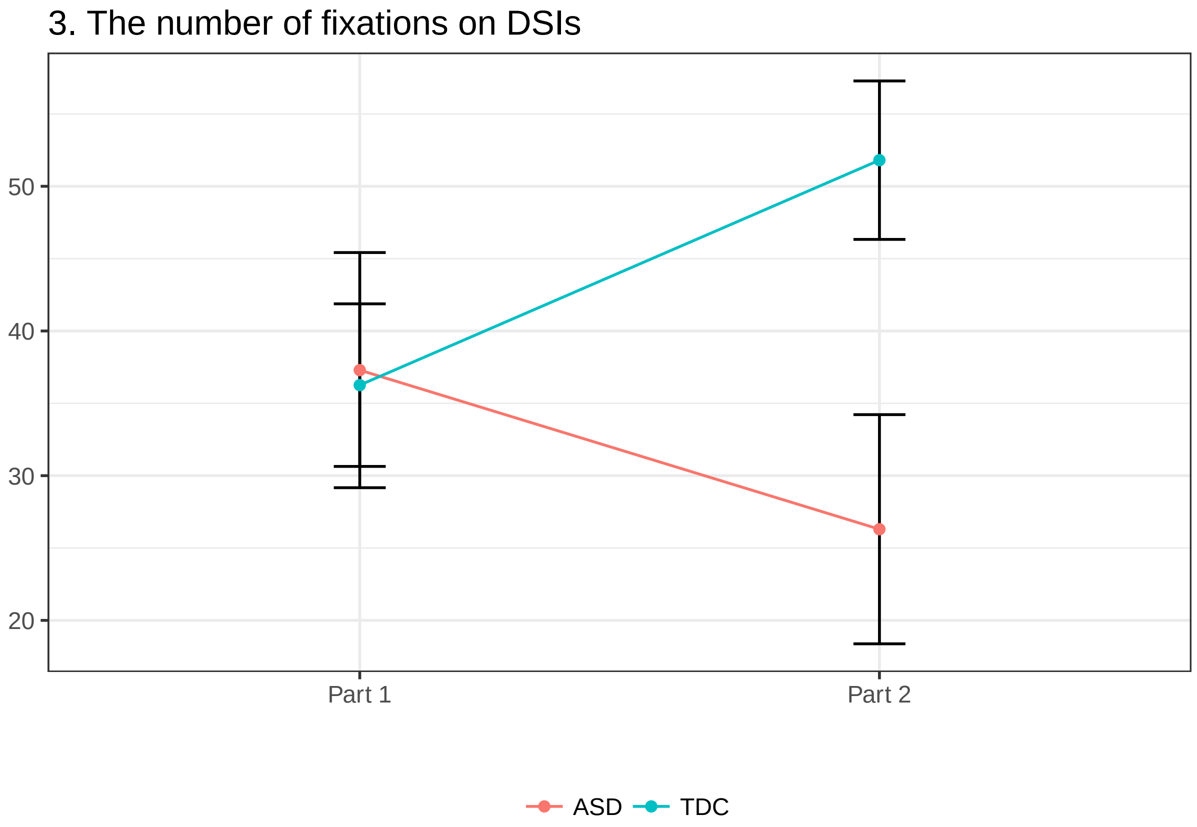
```r
#Plot 2
#Do the Part 1 and Part 2 analyses separately, then combine for plot
#Did not match the paper's plot

#Select and reformat data
fig3data1 <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_percentageof.fixationduration.DSIs,
                S_percentage.of.fixationduration.on.DSIs)
#Generate linear model for part 1
lmodel1 <- lm(P_percentageof.fixationduration.DSIs~PPVT.VIQ + group, data=fig3data1)
refGrid1 <- ref.grid(lmodel1)
#Calculate marginal means for part 1
marMeans1 <- lsmeans(refGrid1, "group")
marMeans1c <- cld(marMeans1)
marMeans1c["part"] <- c("Part 1", "Part 1")
#Generate linear model for part 2
```

```r
lmodel2 <- lm(S_percentage.of.fixationduration.on.DSIs~PPVT.VIQ + group, data=fig3data1)
refGrid2 <- ref.grid(lmodel2)
#Calculate marginal means for part 2
marMeans2 <- lsmeans(refGrid2, "group")
marMeans2c <- cld(marMeans2)
marMeans2c["part"] <- c("Part 2", "Part 2")

#Combine marginal means for both parts
marMeansc <- rbind(marMeans1c, marMeans2c)
#Plot marginal means
ggplot(marMeansc, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("2. The percentage of fixation duration on DSIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```

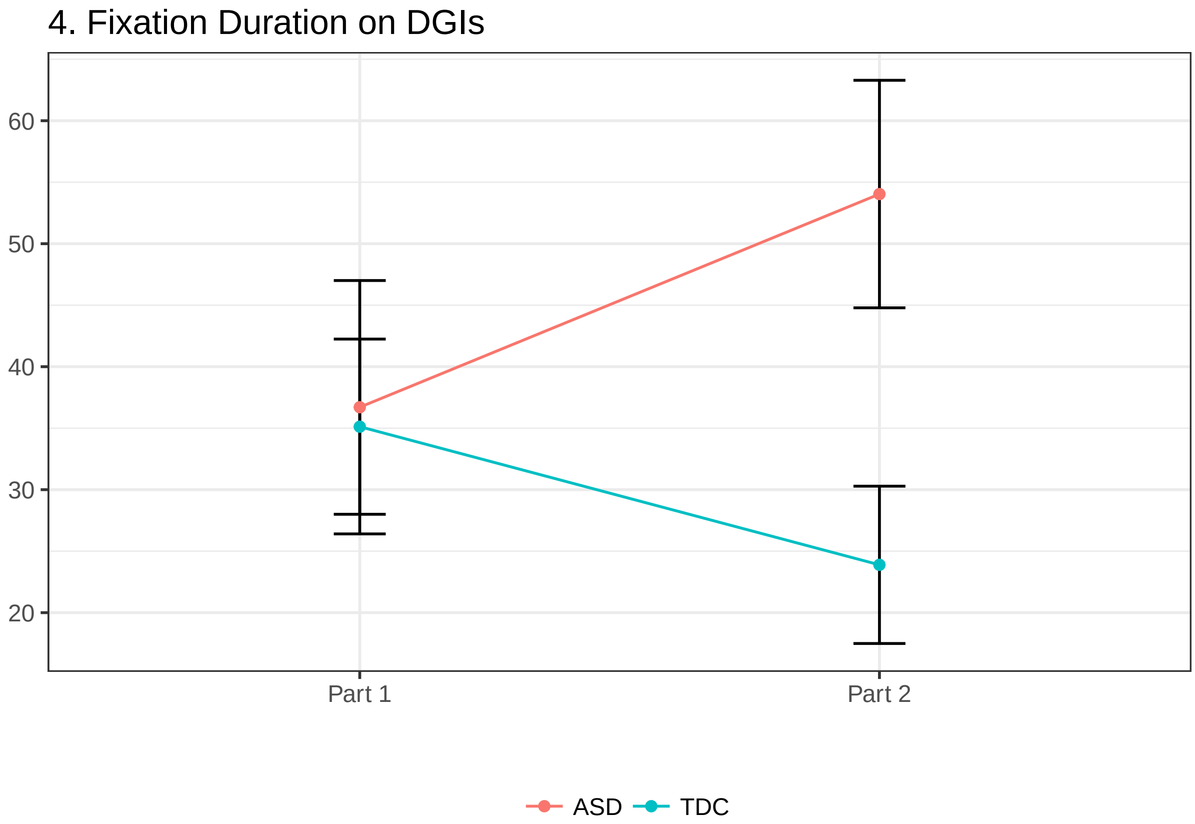


```r
#Plot 3
#Do the Part 1 and Part 2 analyses separately, then combine for plot
#Did not match the paper's plot

#Select and reformat data
fig3data1 <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_number.on.DSIs, S_number.on.DSIs)
```

```r
#Generate linear model for part 1
lmodel1 <- lm(P_number.on.DSIs~PPVT.VIQ + group, data=fig3data1)
refGrid1 <- ref.grid(lmodel1)
#Calculate marginal means for part 1
marMeans1 <- lsmeans(refGrid1, "group")
marMeans1c <- cld(marMeans1)
marMeans1c["part"] <- c("Part 1", "Part 1")
#Generate linear model for part 2
lmodel2 <- lm(S_number.on.DSIs~PPVT.VIQ + group, data=fig3data1)
refGrid2 <- ref.grid(lmodel2)
#Calculate marginal means for part 2
marMeans2 <- lsmeans(refGrid2, "group")
marMeans2c <- cld(marMeans2)
marMeans2c["part"] <- c("Part 2", "Part 2")

#Combine marginal means for both parts
marMeansc <- rbind(marMeans1c, marMeans2c)
#Plot marginal means
ggplot(marMeansc, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("3. The number of fixations on DSIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



```r
#Plot 4
#Do the Part 1 and Part 2 analyses separately, then combine for plot
#Did not match the paper's plot

#Select and reformat data
fig3data1 <- proj_data %>%
  select(group, PPVT.VIQ,P_DGIs.fixationduration, S_fixationduration.on.DGIs)
#Generate linear model for part 1
lmodel1 <- lm(P_DGIs.fixationduration~PPVT.VIQ + group, data=fig3data1)
refGrid1 <- ref.grid(lmodel1)
#Calculate marginal means for part 1
marMeans1 <- lsmeans(refGrid1, "group")
marMeans1c <- cld(marMeans1)
marMeans1c["part"] <- c("Part 1", "Part 1")
#Generate linear model for part 2
lmodel2 <- lm(S_fixationduration.on.DGIs~PPVT.VIQ + group, data=fig3data1)
refGrid2 <- ref.grid(lmodel2)
#Calculate marginal means for part 2
marMeans2 <- lsmeans(refGrid2, "group")
marMeans2c <- cld(marMeans2)
marMeans2c["part"] <- c("Part 2", "Part 2")

#Combine marginal means for both parts
marMeansc <- rbind(marMeans1c, marMeans2c)
#Plot marginal means
ggplot(marMeansc, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("4. Fixation Duration on DGIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



```
#Plot 5
#Do the Part 1 and Part 2 analyses separately, then combine for plot
#Did not match the paper's plot

#Select and reformat data
fig3data1 <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_percentageof.fixationduration.DGIs,
                S_percentage.of.fixationduration.on.DGIs)
```

```
Error in .f(.x[[i]], ...): object 'P_percentageof.fixationduration.DGIs' not found
```

```
#Generate linear model for part 1
lmodel1 <- lm(P_percentageof.fixationduration.DGIs~PPVT.VIQ + group, data=fig3data1)
```

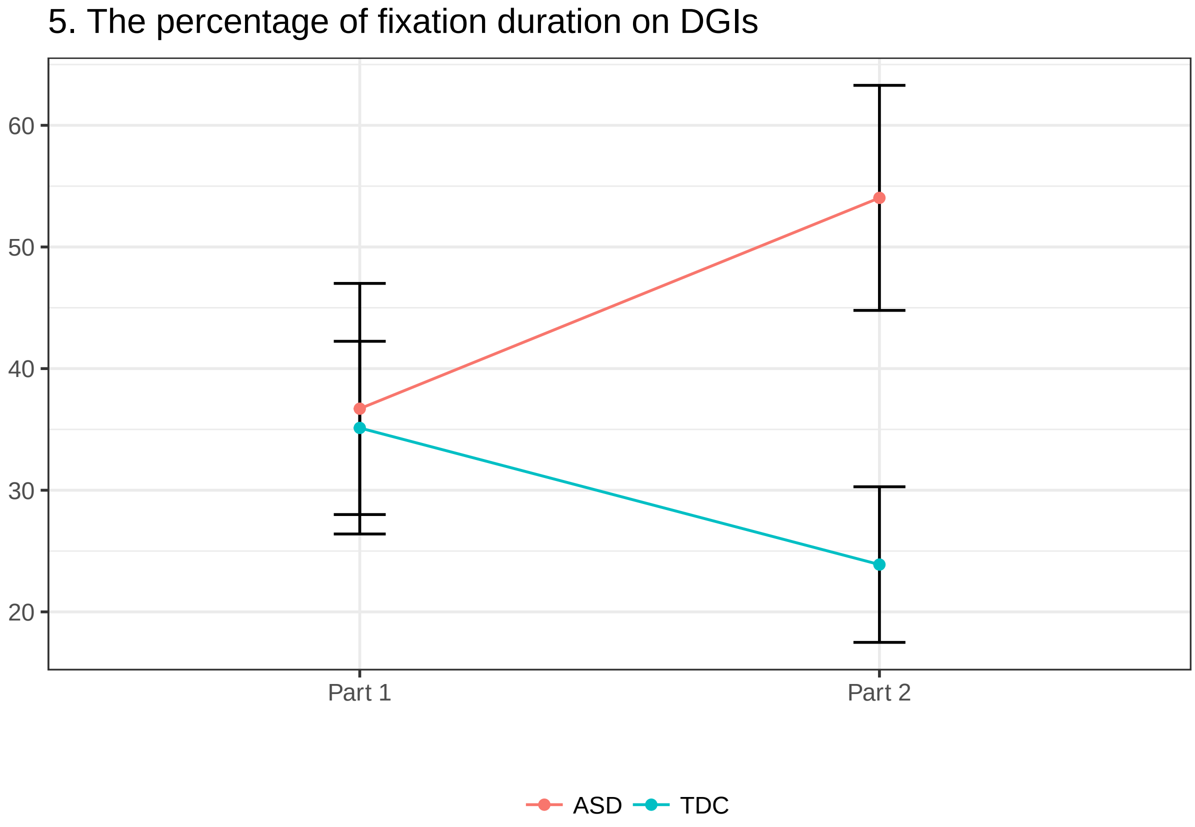
```
Error in eval(predvars, data, env): object 'P_percentageof.fixationduration.DGIs' not found
```

```
refGrid1 <- ref.grid(lmodel1)
#Calculate marginal means for part 1
marMeans1 <- lsmeans(refGrid1, "group")
marMeans1c <- cld(marMeans1)
marMeans1c["part"] <- c("Part 1", "Part 1")
```

```
#Generate linear model for part 2
lmodel2 <- lm(S_percentage.of.fixationduration.on.DGIs~PPVT.VIQ + group, data=fig3data1)
```

```
Error in eval(predvars, data, env): object 'S_percentage.of.fixationduration.on.DGIs' not found
```

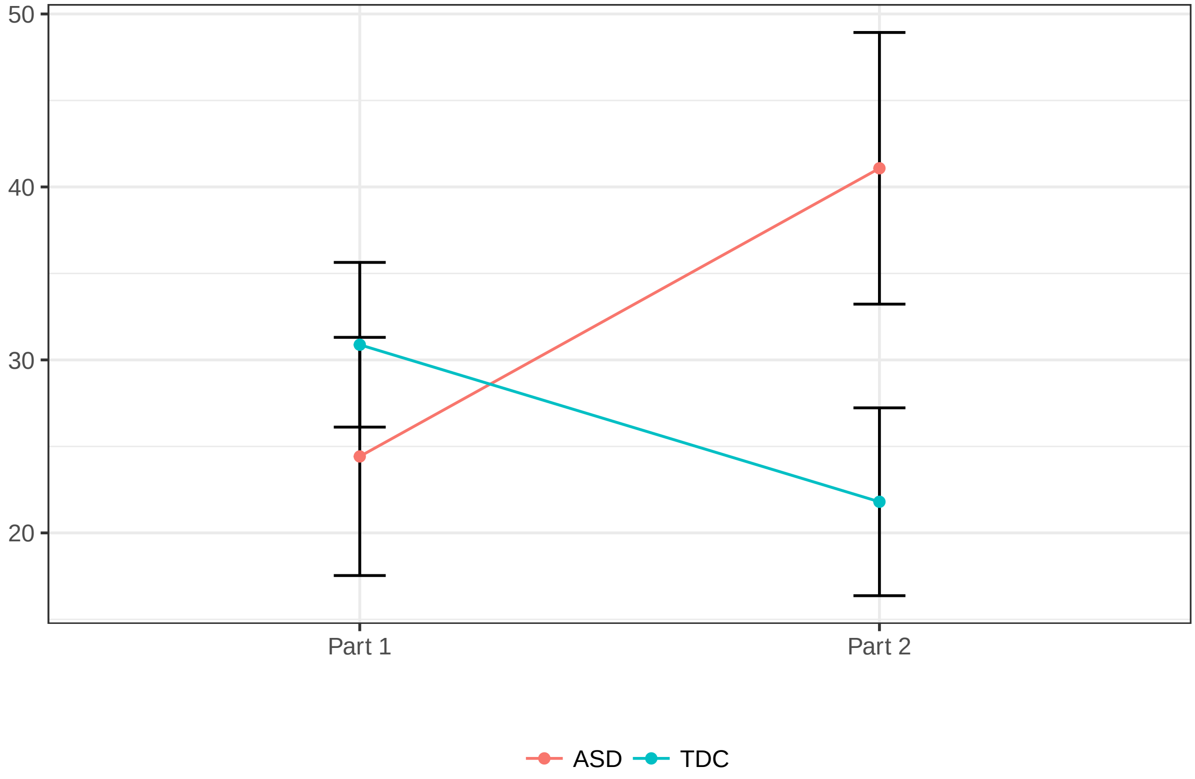
```
refGrid2 <- ref.grid(lmodel2)
#Calculate marginal means for part 2
marMeans2 <- lsmeans(refGrid2, "group")
marMeans2c <- cld(marMeans2)
marMeans2c["part"] <- c("Part 2", "Part 2")

#Combine marginal means for both parts
marMeansc <- rbind(marMeans1c, marMeans2c)
#Plot marginal means
ggplot(marMeansc, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("5. The percentage of fixation duration on DGIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```



```
#Plot 6
#Do the Part 1 and Part 2 analyses separately, then combine for plot
#Did not match the paper's plot
```

```r
#Select and reformat data
fig3data1 <- proj_data %>%
  dplyr::select(group, PPVT.VIQ, P_number.on.DGIs, S_number.on.DGIs)
#Generate linear model for part 1
lmodel1 <- lm(P_number.on.DGIs~PPVT.VIQ + group, data=fig3data1)
refGrid1 <- ref.grid(lmodel1)
#Calculate marginal means for part 1
marMeans1 <- lsmeans(refGrid1, "group")
marMeans1c <- cld(marMeans1)
marMeans1c["part"] <- c("Part 1", "Part 1")
#Generate linear model for part 2
lmodel2 <- lm(S_number.on.DGIs~PPVT.VIQ + group, data=fig3data1)
refGrid2 <- ref.grid(lmodel2)
#Calculate marginal means for part 2
marMeans2 <- lsmeans(refGrid2, "group")
marMeans2c <- cld(marMeans2)
marMeans2c["part"] <- c("Part 2", "Part 2")

#Combine marginal means for both parts
marMeansc <- rbind(marMeans1c, marMeans2c)
#Plot marginal means
ggplot(marMeansc, aes(x=part, y=lsmean, colour=group, group=group)) +
  geom_errorbar(aes(ymin=lsmean-SE, ymax=lsmean+SE), width=.1, colour="black") +
  geom_line() +
  geom_point() +
  xlab("") +
  ylab("") +
  scale_y_continuous(breaks=seq(10, 70, 10)) +
  theme_bw() +
  ggtitle("6. The number of fixations on DGIs") +
  theme(legend.position="bottom", legend.title=element_blank())
```

6. The number of fixations on DGIs



Note

- Figures 3 and 4 are not replicated as there is not enough detail about the predictive model given in the paper to do so.
- Figures 5 through 8 are not replicated as these require specialized eye tracking software (iMap) and raw eye tracking data which is not included in the dataset made available to the public.