

Data Analysis - Manually-Scored Data (BORIS)

Daniela C. Rößler, Massimo De Agrò, Kris Kim, Paul S. Shamble

Data analysis - Predator recognition - hand-scored data using BORIS

Load packages

```
library(car) #for anova
library(glmmTMB) #for mixed models
library(DHARMa) #for residual distribution check
library(ggplot2) #for plotting
library(emmeans) #for post-hoc
library(fitdistrplus) #for data distribution
library(dplyr) #for stats
library(lsmeans) #compare slopes
library(svglite) #create svg files
```

Load data

```
master <- read.csv("C:/Users/roess/Desktop/Publications/PredRecognition/Supplements/borisdata.csv")

master$trial <- as.factor(master$trial)
master$day <- as.factor(master$day)
master$subject <- as.factor(master$subject)

#divide by experiment
exp1 <- subset(master, master$exp == 1)
exp2 <- subset(master, master$exp == 2)
```

Experiment 1 - blob vs blobeyes vs shiny vs marp vs phid

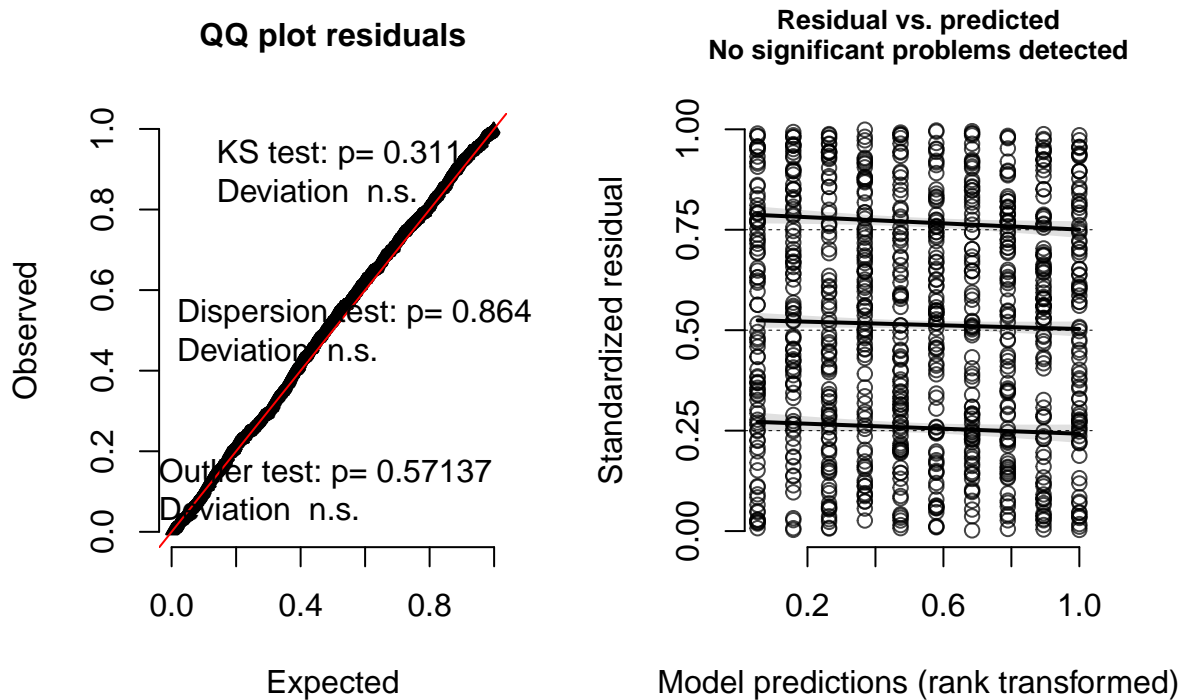
In experiment 1, spiders were tested using 5 different objects (Blob = control, BlobEyes = control + eyes, Marp = Marpissa, Phid = Phidippus, Shiny = 3D model).

Preliminary analysis

All the preliminary analyses will use passing as dependent variable as we consider it to be the more robust and objective measure of recognition of the target. we are going to test all variables against passing regardless of condition.

```
msex <- glmmTMB(passing ~ condition * sex + (1|subject), data = exp1, family=binomial)
simres <- simulateResiduals(msex)
plot(simres)
```

DHARMa residual diagnostics



Sex differences

```
Anova(msex)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: passing
##               Chisq Df Pr(>Chisq)
## condition    177.9853  4    <2e-16 ***
## sex           0.2594  1     0.6106
## condition:sex   3.5150  4     0.4756
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The model is correctly distributed. Since the test is based on the y, we will not repeat the distribution test for every other model that uses passing as dependent variable. There is no differences between sexes.

Day We are going to check whether there was any effect between the two different test days.

```
mday <- glmmTMB(passing ~ condition * day + (1|subject), data=exp1, family = binomial)
Anova(mday)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: passing
##               Chisq Df Pr(>Chisq)
## condition      175.2967  4    <2e-16 ***
## day              1.5482  1     0.2134
## condition:day    3.0741  4     0.5455
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Date We are going to check whether the behavior was balanced over the different dates at which spiders were tested.

```
mdate <- glmmTMB(passing~date + (1|subject), data=exp1, family = binomial)
Anova(mdate)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: passing
##               Chisq Df Pr(>Chisq)
## date 21.428 10      0.0183 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

There is a difference between dates, thus we are going to do a post-hoc test.

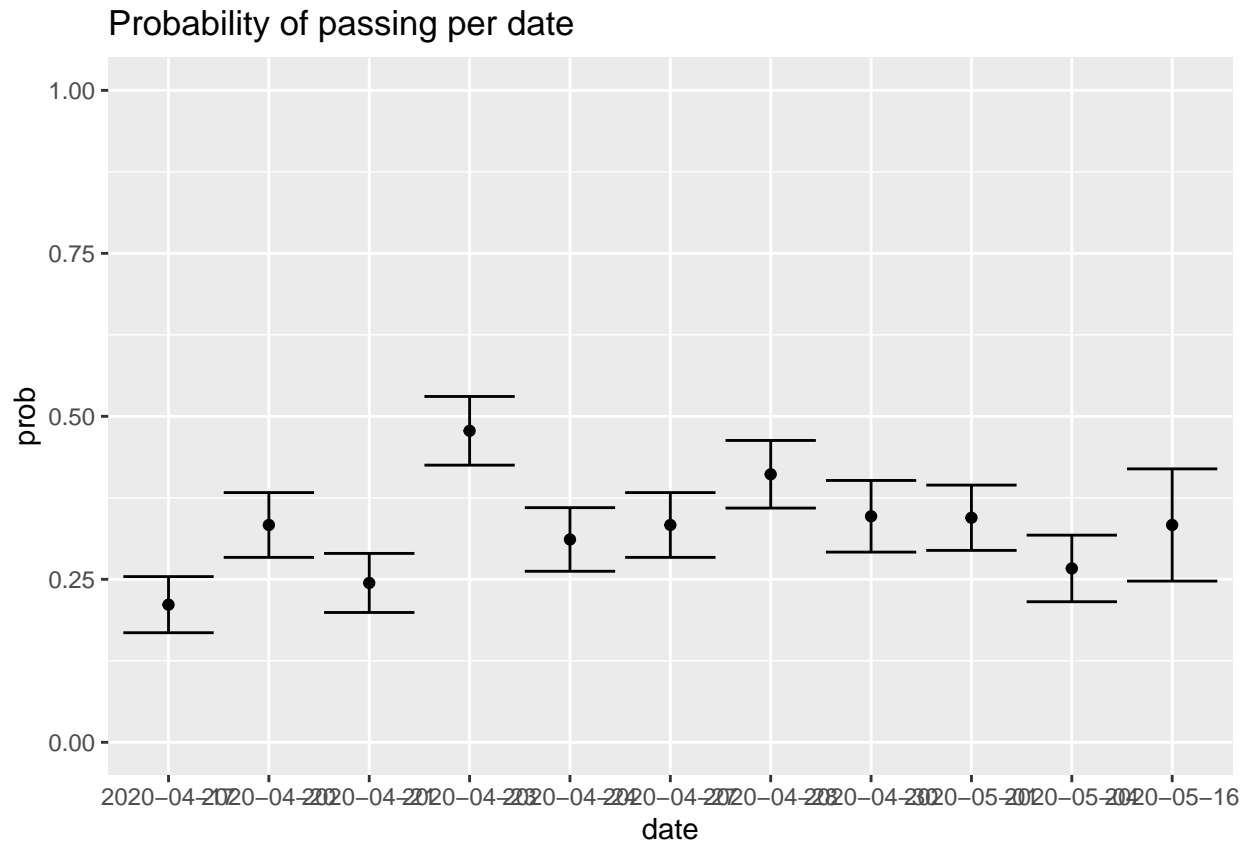
```
e <- emmeans(mdate, ~date, type= "response")
pairs(e, adjust='bonferroni')
```

contrast	odds.ratio	SE	df	t.ratio	p.value
(2020-04-17) / (2020-04-20)	0.535	0.1828	888	-1.830	1.0000
(2020-04-17) / (2020-04-21)	0.827	0.2946	888	-0.533	1.0000
(2020-04-17) / (2020-04-23)	0.292	0.0976	888	-3.686	0.0133
(2020-04-17) / (2020-04-24)	0.593	0.2040	888	-1.520	1.0000
(2020-04-17) / (2020-04-27)	0.535	0.1828	888	-1.830	1.0000
(2020-04-17) / (2020-04-28)	0.383	0.1286	888	-2.857	0.2404
(2020-04-17) / (2020-04-30)	0.504	0.1787	888	-1.932	1.0000
(2020-04-17) / (2020-05-01)	0.509	0.1734	888	-1.982	1.0000
(2020-04-17) / (2020-05-04)	0.736	0.2703	888	-0.835	1.0000
(2020-04-17) / (2020-05-16)	0.535	0.2492	888	-1.343	1.0000
(2020-04-20) / (2020-04-21)	1.545	0.5129	888	1.312	1.0000
(2020-04-20) / (2020-04-23)	0.547	0.1680	888	-1.965	1.0000
(2020-04-20) / (2020-04-24)	1.107	0.3533	888	0.319	1.0000
(2020-04-20) / (2020-04-27)	1.000	0.3162	888	0.000	1.0000
(2020-04-20) / (2020-04-28)	0.716	0.2218	888	-1.078	1.0000
(2020-04-20) / (2020-04-30)	0.942	0.3109	888	-0.180	1.0000
(2020-04-20) / (2020-05-01)	0.952	0.2997	888	-0.157	1.0000
(2020-04-20) / (2020-05-04)	1.375	0.4727	888	0.926	1.0000

```
## (2020-04-20) / (2020-05-16)      1.000 0.4472 888  0.000 1.0000
## (2020-04-21) / (2020-04-23)      0.354 0.1144 888 -3.213 0.0749
## (2020-04-21) / (2020-04-24)      0.716 0.2398 888 -0.997 1.0000
## (2020-04-21) / (2020-04-27)      0.647 0.2148 888 -1.312 1.0000
## (2020-04-21) / (2020-04-28)      0.463 0.1509 888 -2.362 1.0000
## (2020-04-21) / (2020-04-30)      0.610 0.2104 888 -1.434 1.0000
## (2020-04-21) / (2020-05-01)      0.616 0.2036 888 -1.466 1.0000
## (2020-04-21) / (2020-05-04)      0.890 0.3187 888 -0.326 1.0000
## (2020-04-21) / (2020-05-16)      0.647 0.2966 888 -0.950 1.0000
## (2020-04-23) / (2020-04-24)      2.026 0.6289 888  2.274 1.0000
## (2020-04-23) / (2020-04-27)      1.830 0.5626 888  1.965 1.0000
## (2020-04-23) / (2020-04-28)      1.311 0.3941 888  0.899 1.0000
## (2020-04-23) / (2020-04-30)      1.724 0.5544 888  1.694 1.0000
## (2020-04-23) / (2020-05-01)      1.741 0.5331 888  1.811 1.0000
## (2020-04-23) / (2020-05-04)      2.516 0.8447 888  2.748 0.3363
## (2020-04-23) / (2020-05-16)      1.830 0.8070 888  1.370 1.0000
## (2020-04-24) / (2020-04-27)      0.903 0.2882 888 -0.319 1.0000
## (2020-04-24) / (2020-04-28)      0.647 0.2022 888 -1.393 1.0000
## (2020-04-24) / (2020-04-30)      0.851 0.2832 888 -0.484 1.0000
## (2020-04-24) / (2020-05-01)      0.860 0.2732 888 -0.476 1.0000
## (2020-04-24) / (2020-05-04)      1.242 0.4303 888  0.625 1.0000
## (2020-04-24) / (2020-05-16)      0.903 0.4058 888 -0.227 1.0000
## (2020-04-27) / (2020-04-28)      0.716 0.2218 888 -1.078 1.0000
## (2020-04-27) / (2020-04-30)      0.942 0.3109 888 -0.180 1.0000
## (2020-04-27) / (2020-05-01)      0.952 0.2997 888 -0.157 1.0000
## (2020-04-27) / (2020-05-04)      1.375 0.4727 888  0.926 1.0000
## (2020-04-27) / (2020-05-16)      1.000 0.4472 888  0.000 1.0000
## (2020-04-28) / (2020-04-30)      1.316 0.4258 888  0.848 1.0000
## (2020-04-28) / (2020-05-01)      1.329 0.4097 888  0.921 1.0000
## (2020-04-28) / (2020-05-04)      1.920 0.6484 888  1.931 1.0000
## (2020-04-28) / (2020-05-16)      1.396 0.6180 888  0.754 1.0000
## (2020-04-30) / (2020-05-01)      1.010 0.3320 888  0.030 1.0000
## (2020-04-30) / (2020-05-04)      1.459 0.5201 888  1.060 1.0000
## (2020-04-30) / (2020-05-16)      1.061 0.4850 888  0.130 1.0000
## (2020-05-01) / (2020-05-04)      1.445 0.4951 888  1.074 1.0000
## (2020-05-01) / (2020-05-16)      1.051 0.4690 888  0.111 1.0000
## (2020-05-04) / (2020-05-16)      0.727 0.3397 888 -0.682 1.0000
##
## P value adjustment: bonferroni method for 55 tests
## Tests are performed on the log odds ratio scale
```

It seems like the 23 of april is different in terms of retreat amount from the 17 and the 21. We will plot the data to visualize it.

```
toplotdate <- as.data.frame(e)
ggplot(toplotdate, aes(x=date, y=prob))+
  geom_point()+
  geom_errorbar(aes(ymin=prob-SE,ymax=prob+SE))+
  ylim(0,1)+
  ggtitle("Probability of passing per date")
```



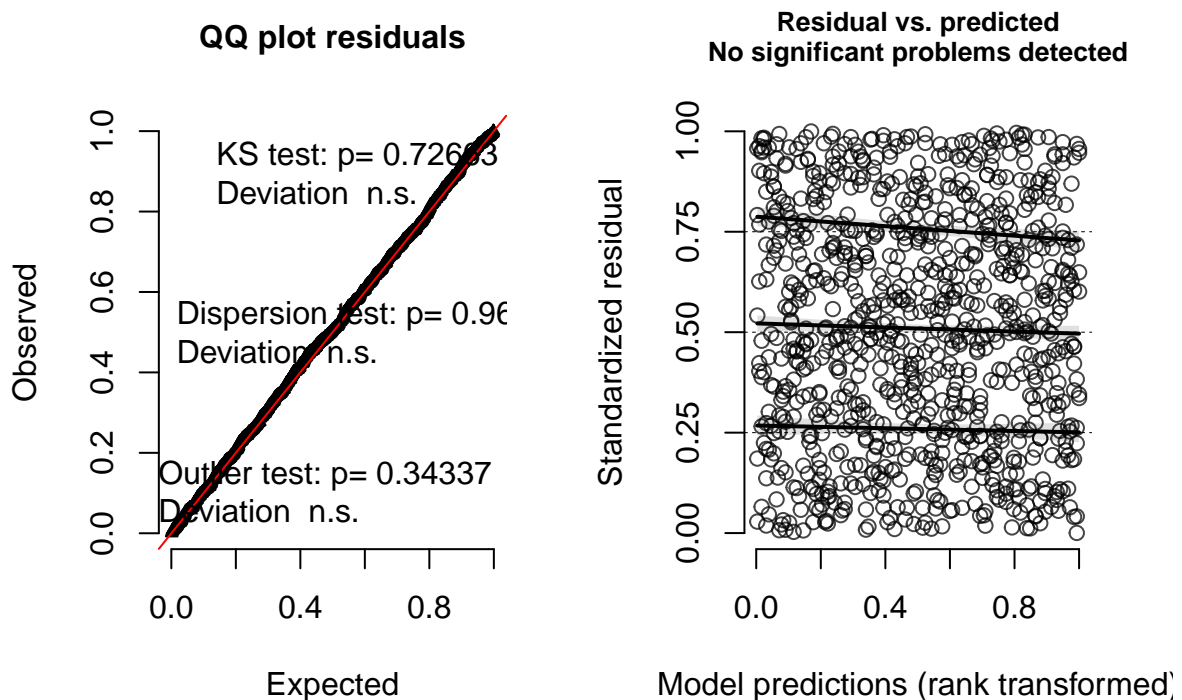
It appears that the difference is significant against day 17th and 21st. The 23rd has a lower percentage of retreats in general. It just so happens that days 17 and 21 are the highest, so that the difference only remains there. In general, this difference should not be concerning, since the conditions are balanced over all dates.

Main analysis

Passing the target We are now testing the potential effect of condition and order of object presentation on the probability to pass the object. We use subject as a random factor.

```
mpass <- glmmTMB(passing~condition * test * day + (1|subject), data= exp1, family= binomial)
simres <- simulateResiduals(mpass)
plot(simres)
```

DHARMA residual diagnostics



`Anova(mpass)`

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: passing
##               Chisq Df Pr(>Chisq)
## condition      158.7771  4    < 2e-16 ***
## test           2.5863  1    0.10779
## day            1.2615  1    0.26136
## condition:test    10.6176  4    0.03122 *
## condition:day     3.5979  4    0.46315
## test:day          0.1669  1    0.68284
## condition:test:day 0.8590  4    0.93036
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The distribution is correct. We see a significant effect of condition as well as an effect in the interaction of condition and presentation order. We will do a post-hoc test to see the differences in conditions. The post-hoc will not contain test since is a continuous variable and cannot be inserted in estimated mean analysis.

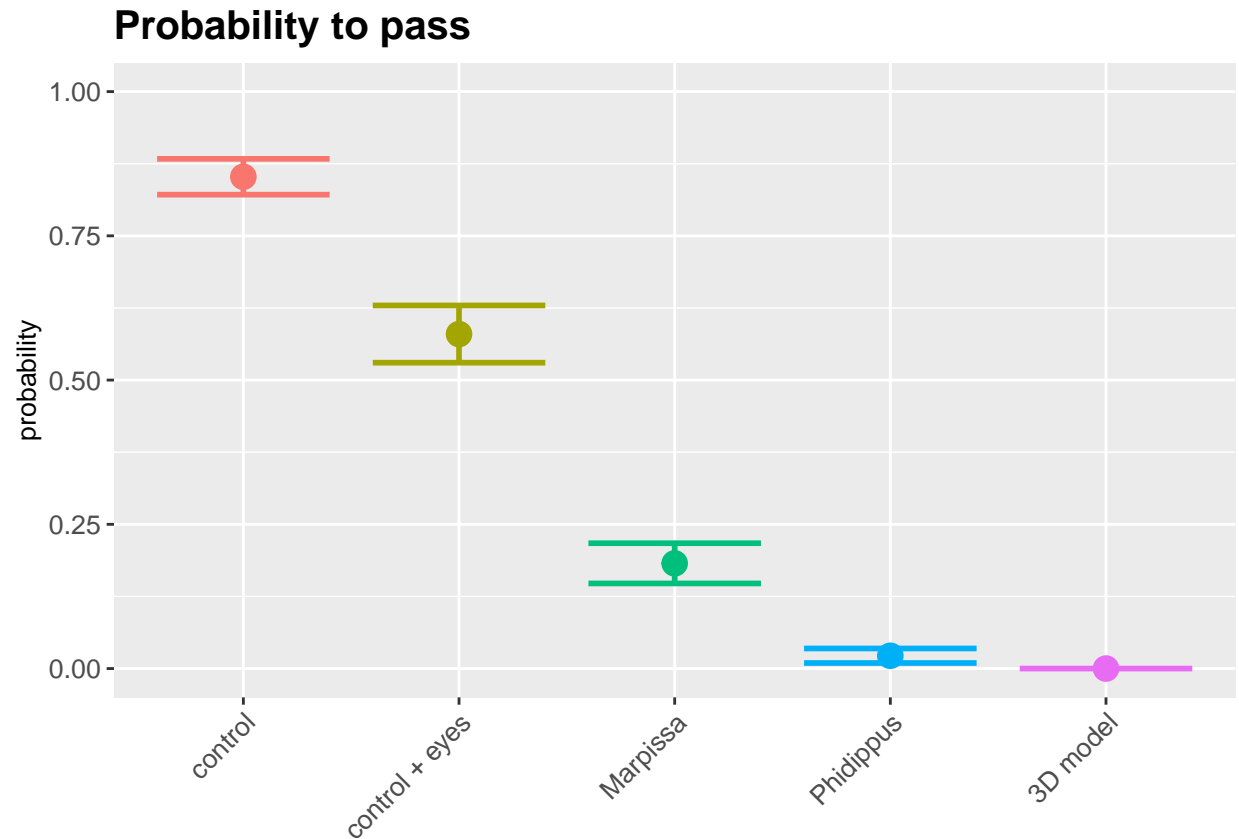
```
toplotpass <- emmeans(mpass, ~condition, type="response")
pairs(toplotpass, adjust = 'bonferroni')
```

```
## contrast      odds.ratio      SE df t.ratio p.value
```

```
## Blob / BlobEyes      4.00e+00 1.00e+00 879  5.334 <.0001
## Blob / Marp          2.60e+01 8.00e+00 879 10.756 <.0001
## Blob / Phid          2.55e+02 1.57e+02 879  9.016 <.0001
## Blob / Shiny         2.15e+16 3.35e+21 879  0.000 1.0000
## BlobEyes / Marp      6.00e+00 2.00e+00 879  6.988 <.0001
## BlobEyes / Phid      6.10e+01 3.60e+01 879  6.928 <.0001
## BlobEyes / Shiny     5.14e+15 8.00e+20 879  0.000 1.0000
## Marp / Phid          1.00e+01 6.00e+00 879  3.832 0.0014
## Marp / Shiny         8.30e+14 1.29e+20 879  0.000 1.0000
## Phid / Shiny         8.44e+13 1.31e+19 879  0.000 1.0000
##
## Results are averaged over the levels of: day
## P value adjustment: bonferroni method for 10 tests
## Tests are performed on the log odds ratio scale
```

All pairwise comparisons are significant except for Marpissa and Shiny as well as Phidippus and Shiny, also Blob and Shiny are not significant which can only be explained by a complete data separation (no spider ever passed this condition). We will plot the model:

```
toplotpass <- as.data.frame(toplotpass)
ggplot(toplotpass, aes(condition, prob, color= condition))+
  geom_point(cex = 4)+
  geom_errorbar(aes(ymin=prob-SE,ymax=prob+SE), cex = 1, width= .8)+
  ylim(0,1)+
  labs(title = "Probability to pass", y= "probability")+
  scale_x_discrete(labels = c("control", "control + eyes",
                             "Marpissa", "Phidippus", "3D model"))+
  theme(axis.title.x = element_blank(), axis.text.x = element_text(size=10, angle=45, hjust = 1),
        axis.text.y = element_text(size=10), axis.title.y = element_text(size=10),
        plot.title = element_text(size= 15, face= "bold"), legend.position = "none")+
  ggsave(filename = "pass_boris.svg", width = 8, height = 10)
```

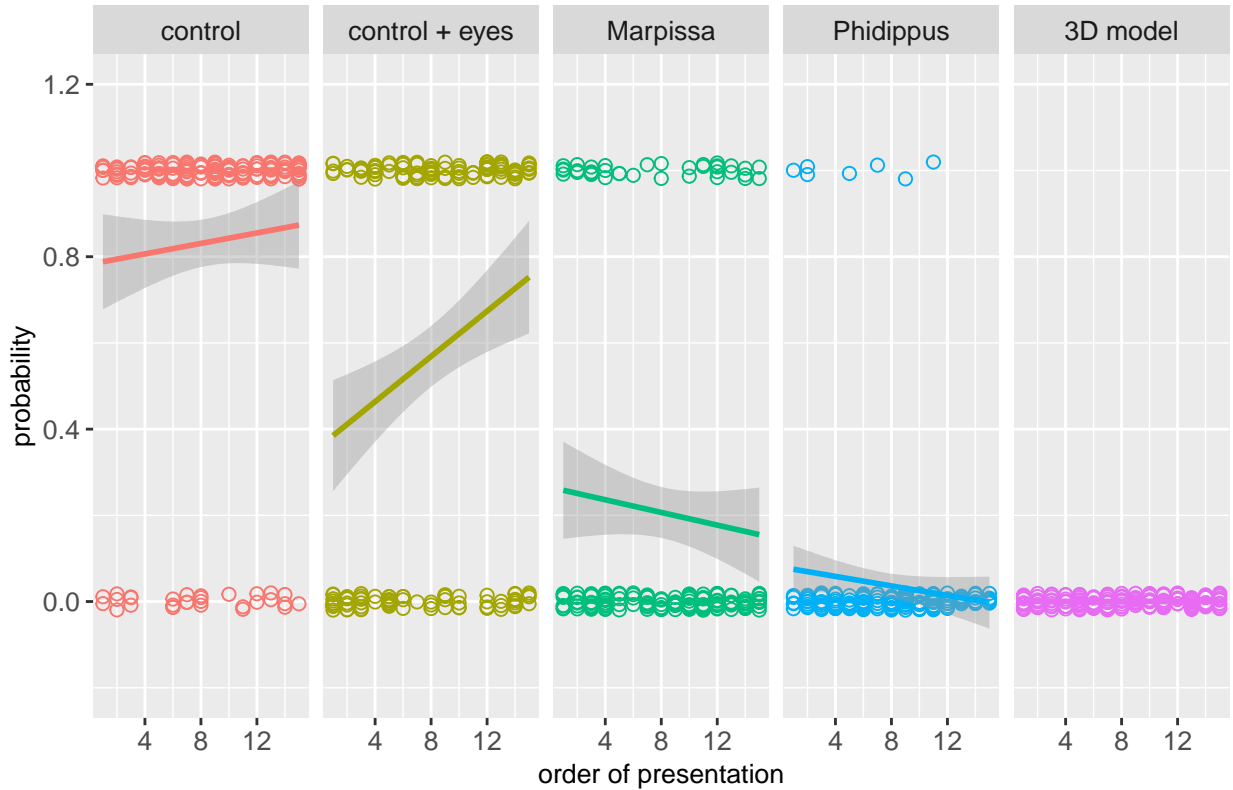


We can clearly see that virtually no spider ever passed the 3D model (Shiny), however, the probability to pass the control lies around 80%. The probability to pass the control with eyes only lies around 55%, which is significantly less than the control. Due to the interaction effect we saw in the model, we will now plot this probability depending on presentation order:

```
new.labels2 <- c("Blob" = "control", "BlobEyes" = "control + eyes",
                 "Marp" = "Marpissa", "Phid" = "Phidippus", "Shiny" = "3D model")

ggplot(exp1, aes(x=test, y=passing, color= condition)) +
  facet_grid(~condition, labeller = labeller(condition = new.labels2)) +
  geom_jitter(width = 0, height = 0.02, shape = 1, size=2) +
  geom_smooth(method="glm") +
  ylim(-0.2, 1.2) +
  labs(title = "Probability to pass depending on order of presentation",
       x = "order of presentation", size= 12, y = "probability", size = 12) +
  theme(strip.text = element_text(size=10), axis.text.x = element_text(size=10),
        axis.text.y = element_text(size=10), axis.title.x = element_text(size=10),
        axis.title.y = element_text(size=10), plot.title = element_text(size= 15, face= "bold"),
        legend.position = "none")
```


Probability to pass depending on order of presentation



We can see that the effect likely stems from the Blobeye condition, showing that passing probability increases significantly with the number of tests for this condition. Next, we will examine the interaction effect of condition and test by analysing the slopes for each condition and using post-hoc correction

```
mlst <- lstrends(mpass, "condition", var= "test")
test(mlst, adjust= "bonferroni")
```

```
## condition test.trend      SE df t.ratio p.value
## Blob      0.0431 5.00e-02 879  0.853  1.0000
## BlobEyes  0.1126 3.70e-02 879  3.059  0.0114
## Marp      -0.0377 4.40e-02 879 -0.853  1.0000
## Phid      -0.1667 1.29e-01 879 -1.296  0.9769
## Shiny     -2.1975 2.26e+04 879  0.000  1.0000
##
## Results are averaged over the levels of: day
## P value adjustment: bonferroni method for 5 tests
```

As expected the effect comes from the control + eyes condition.

Retreat Retreats (meaning moving away from the object after freezing) was observable in the videos. It is a proxy of recognition. We noticed that when detecting a threatening object, spiders freeze and subsequently retreat. Using retreat probability as predicted by condition we can infer the presence of detection.

The model will contain:

- condition, as is obvious

- test, which indicates the order in which the objects are presented within an experiment
- day, the test day, either 1 or 2

```
mret <-glmmTMB(retreat~condition * test * day + (1|subject), data = exp1, family=binomial)
Anova(mret)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: retreat
##               Chisq Df Pr(>Chisq)
## condition      205.7786  4    < 2e-16 ***
## test            2.1972  1    0.13826
## day             2.0833  1    0.14892
## condition:test   11.0950  4    0.02552 *
## condition:day     1.9562  4    0.74381
## test:day          0.3692  1    0.54344
## condition:test:day 2.2475  4    0.69035
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

There is a strong effect of condition, meaning that spiders retreat differently for different objects. There is also an effect of the interaction between condition and test, meaning that the retreat probability changes with time (or presentation order) but only for one or some conditions; or alternatively, the change in probability with the passage of time changes direction with condition. To better see the effect, we will do a post-hoc test to see the differences in conditions. The post-hoc will not contain test since it is a continuous variable and cannot be inserted in estimated mean analysis. Lastly, we will plot the probability of retreat for each condition.

```
e <- emmeans(mret,~condition, type = "response")
pairs(e, adjust = 'bonferroni')
```

```
## contrast      odds.ratio      SE df t.ratio p.value
## Blob / BlobEyes  0.122194 0.039783 879 -6.457 <.0001
## Blob / Marp      0.014320 0.005332 879 -11.403 <.0001
## Blob / Phid       0.002705 0.001413 879 -11.320 <.0001
## Blob / Shiny      0.000682 0.000642 879 -7.744 <.0001
## BlobEyes / Marp   0.117189 0.033199 879 -7.568 <.0001
## BlobEyes / Phid   0.022136 0.010133 879 -8.324 <.0001
## BlobEyes / Shiny  0.005585 0.005061 879 -5.724 <.0001
## Marp / Phid       0.188888 0.085549 879 -3.680 0.0025
## Marp / Shiny      0.047654 0.043009 879 -3.373 0.0078
## Phid / Shiny      0.252287 0.243141 879 -1.429 1.0000
##
## Results are averaged over the levels of: day
## P value adjustment: bonferroni method for 10 tests
## Tests are performed on the log odds ratio scale
```

Here we see: * all objects have a significantly higher retreat probability than Blob * all objects but Blob have a significantly higher retreat probability than Blob with eyes * both Phid and Shiny have a higher retreat probability than Marp. * There is no difference between Phid and Shiny

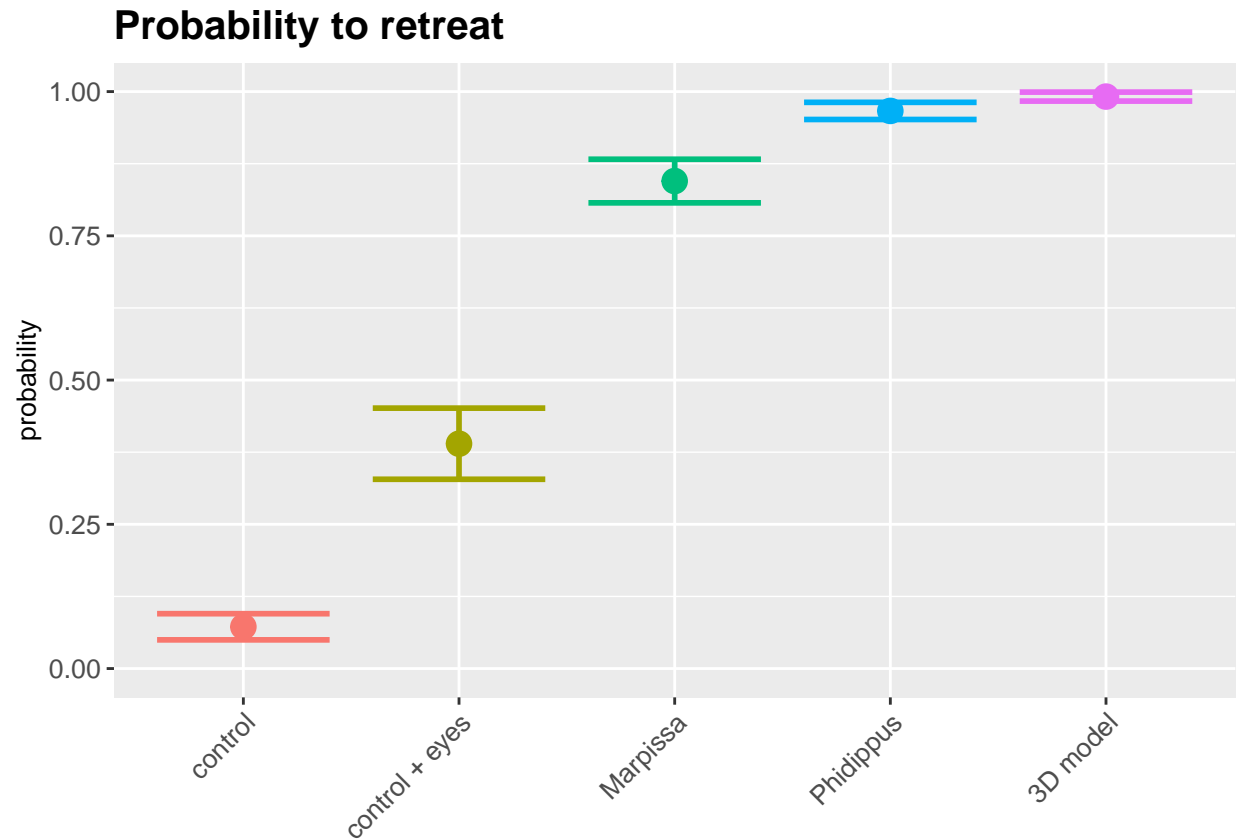
Next, we will examine the interaction effect of condition and test by analysing the slopes for each condition and using post-hoc correction

```
mlst <- lstrends(mret, "condition", var= "test")
test(mlst, adjust= "bonferroni")
```

```
## condition test.trend      SE df t.ratio p.value
## Blob          0.01519 0.0641 879  0.237  1.0000
## BlobEyes     -0.12998 0.0397 879 -3.272  0.0056
## Marp          0.00752 0.0465 879  0.162  1.0000
## Phid          0.15747 0.0967 879  1.629  0.5182
## Shiny        -0.11295 0.1787 879 -0.632  1.0000
##
## Results are averaged over the levels of: day
## P value adjustment: bonferroni method for 5 tests
```

We will now produce two plots: one showing just the differences in conditions, and one highlighting the presentation order effect.

```
toplotret <- as.data.frame(e)
ggplot(toplotret, aes(x=condition,y=prob, color= condition))+
  geom_point(cex = 4)+
  geom_errorbar(aes(ymin=prob-SE,ymax=prob+SE), cex = 1, width= .8)+
  ylim(0,1)+
  labs(title = "Probability to retreat", y= "probability")+
  scale_x_discrete(labels = c("control", "control + eyes", "Marpissa",
                             "Phidippus", "3D model"))+
  theme(axis.title.x = element_blank(), axis.text.x = element_text(size=10, angle=45, hjust = 1),
        axis.text.y = element_text(size=10), axis.title.y = element_text(size=10),
        plot.title = element_text(size= 15, face= "bold"), legend.position = "none")
```

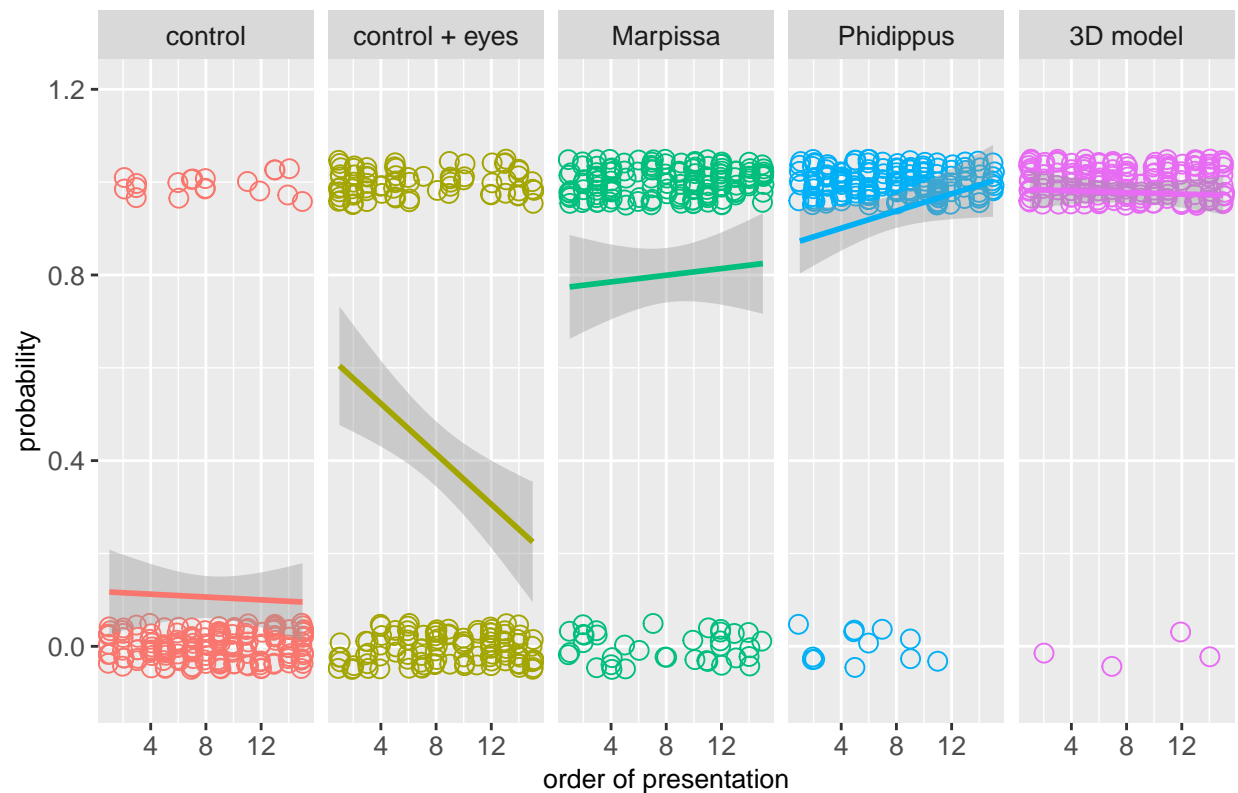


```
ggsave("toplotret.svg")
```

```
new.labels <- c("Blob" = "control", "BlobEyes" = "control + eyes",
               "Marp" = "Marpissa", "Phid" = "Phidippus", "Shiny" = "3D model")

ggplot(exp1, aes(x=test, y=retreat, color=condition)) +
  facet_grid(~condition, labeller = labeller(condition = new.labels)) +
  geom_jitter(width = 0.1, height = 0.05, shape=1, size=3) +
  geom_smooth(method="glm") +
  ylim(-0.1, 1.2) +
  labs(title = "Probability to retreat depending on order of presentation",
       x = "order of presentation", size = 12, y = "probability", size = 12) +
  theme(strip.text = element_text(size=10), axis.text.x = element_text(size=10),
        axis.text.y = element_text(size=10), axis.title.x = element_text(size=10),
        axis.title.y = element_text(size=10), plot.title = element_text(size=15, face="bold"),
        legend.position = "none")
```

Probability to retreat depending on order of presentation



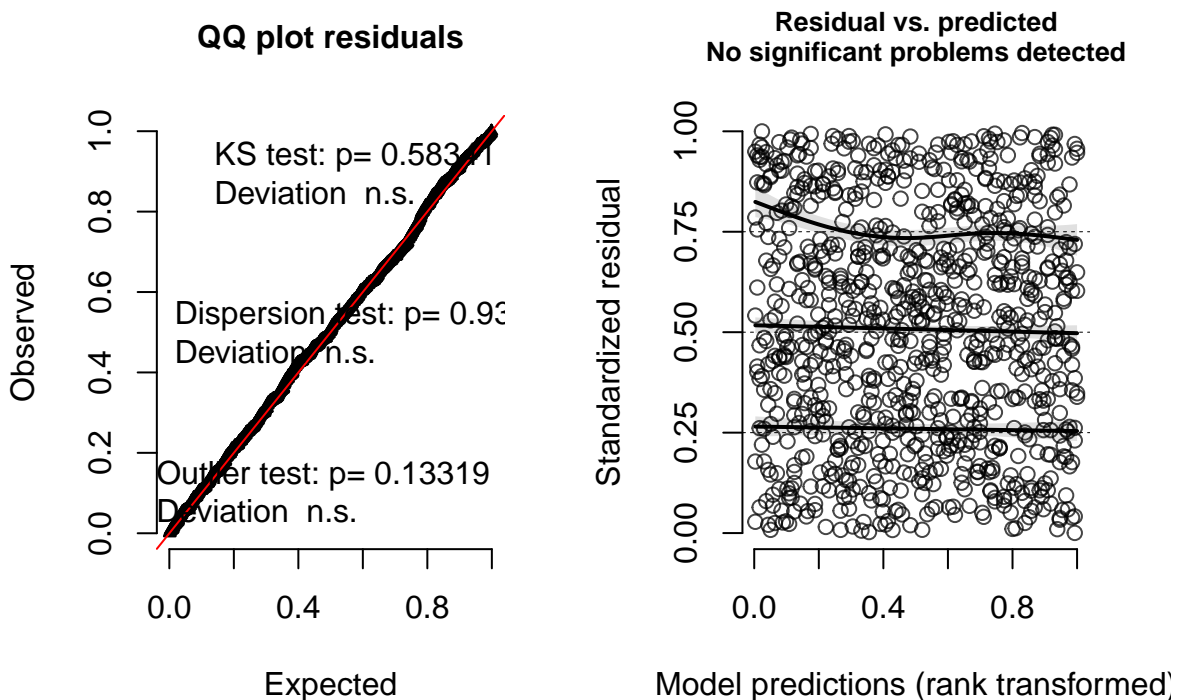
```
ggsave("exp1.svg")
```

We can see that retreat supports the same findings as the variable “passing” which we assume to be more robust and less subjective.

Jump We expect jump to be in line with the earlier analyses. For completeness we will follow the same steps for this variable.

```
mj <- glmmTMB(jump~condition * test * day +(1|subject), data = exp1, family = binomial)
simres <- simulateResiduals(mj)
plot(simres)
```

DHARMA residual diagnostics



```
Anova(mj)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: jump
##               Chisq Df Pr(>Chisq)
## condition      200.4201  4    < 2e-16 ***
## test            0.9791  1    0.32241
## day             1.6773  1    0.19528
## condition:test   12.5474  4    0.01371 *
## condition:day     8.4590  4    0.07614 .
## test:day         0.0024  1    0.96124
## condition:test:day 0.5524  4    0.96820
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The distribution is correct. The effect remains the same as before. We now do a post-hoc test.

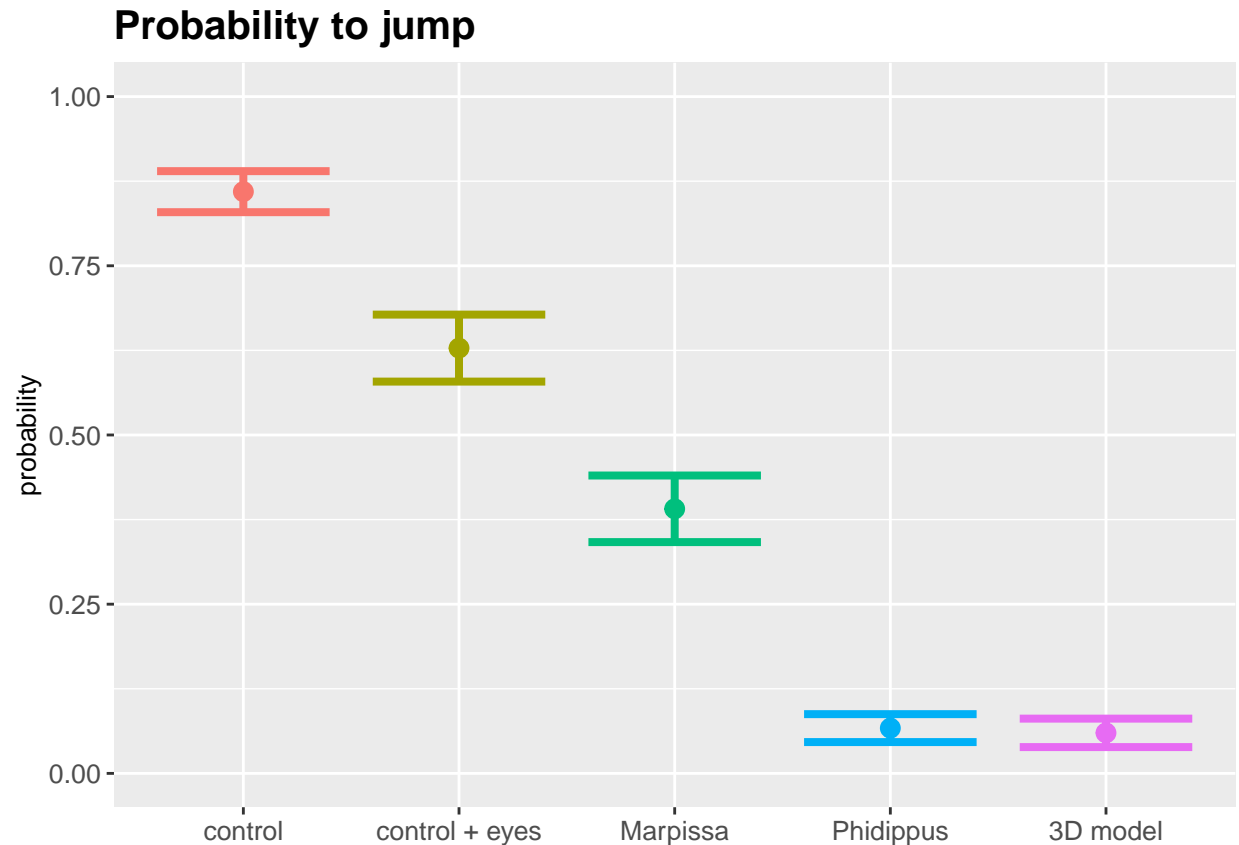
```
e <- emmeans(mj, ~condition, type="response")
pairs(e, adjust = 'bonferroni')
```

```
## contrast      odds.ratio      SE df t.ratio p.value
## Blob / BlobEyes      3.62  0.981 879  4.745 <.0001
## Blob / Marp          9.53  2.591 879  8.296 <.0001
```

```
## Blob / Phid      85.36 32.526 879 11.670 <.0001
## Blob / Shiny     96.09 40.256 879 10.897 <.0001
## BlobEyes / Marp   2.63  0.617 879  4.134 0.0004
## BlobEyes / Phid  23.59  8.282 879  9.002 <.0001
## BlobEyes / Shiny 26.55 10.413 879  8.362 <.0001
## Marp / Phid      8.95  3.070 879  6.392 <.0001
## Marp / Shiny     10.08  3.881 879  6.000 <.0001
## Phid / Shiny      1.13  0.513 879  0.260 1.0000
##
## Results are averaged over the levels of: day
## P value adjustment: bonferroni method for 10 tests
## Tests are performed on the log odds ratio scale
```

The results look the same. We will generate the two plots for completeness (probability and order of presentation).

```
toplotj <- as.data.frame(e)
ggplot(toplotj, aes(x=condition,y=prob, color=condition))+
  geom_point(cex = 3)+
  geom_errorbar(aes(ymin=prob-SE,ymax=prob+SE), cex = 1.4, width= .8)+
  ylim(0,1)+
  labs(title = "Probability to jump", y= "probability")+
  scale_x_discrete(labels = c("control", "control + eyes",
                             "Marpissa", "Phidippus", "3D model"))+
  theme(axis.title.x = element_blank(), axis.text.x = element_text(size=10),
        axis.text.y = element_text(size=10), axis.title.y = element_text(size=10),
        plot.title = element_text(size= 15, face= "bold"), legend.position = "none")
```



The plot looks almost identical to the probability to pass the object. However, Phidippus and the 3D model show slightly higher probabilities to jump compared to the probability to pass the object, most likely because spiders sometimes jump across the gap before looking at the object.

```
new.labels3 <- c("Blob" = "control", "BlobEyes" = "control + eyes",
                 "Marp" = "Marpissa", "Phid" = "Phidippus", "Shiny" = "3D model")

ggplot(exp1, aes(x=test,y=jump,color=condition))+
  facet_grid(~condition, labeller = labeller(condition = new.labels3))+
  geom_jitter(width = 0.1, height = 0.05, shape=1, size=2)+
  geom_smooth(method="glm")+
  ylim(-0.1,1.2)+
  labs(title = "Probability to jump depending on order of presentation",
       x = "order of presentation", size= 12, y = "probability", size = 12)+
  theme(strip.text = element_text(size=10), axis.text.x = element_text(size=10),
        axis.text.y = element_text(size=10), axis.title.x = element_text(size=10),
        axis.title.y = element_text(size=10), plot.title = element_text(size= 15, face= "bold"),
        legend.position = "none")
```


Probability to jump depending on order of presentation

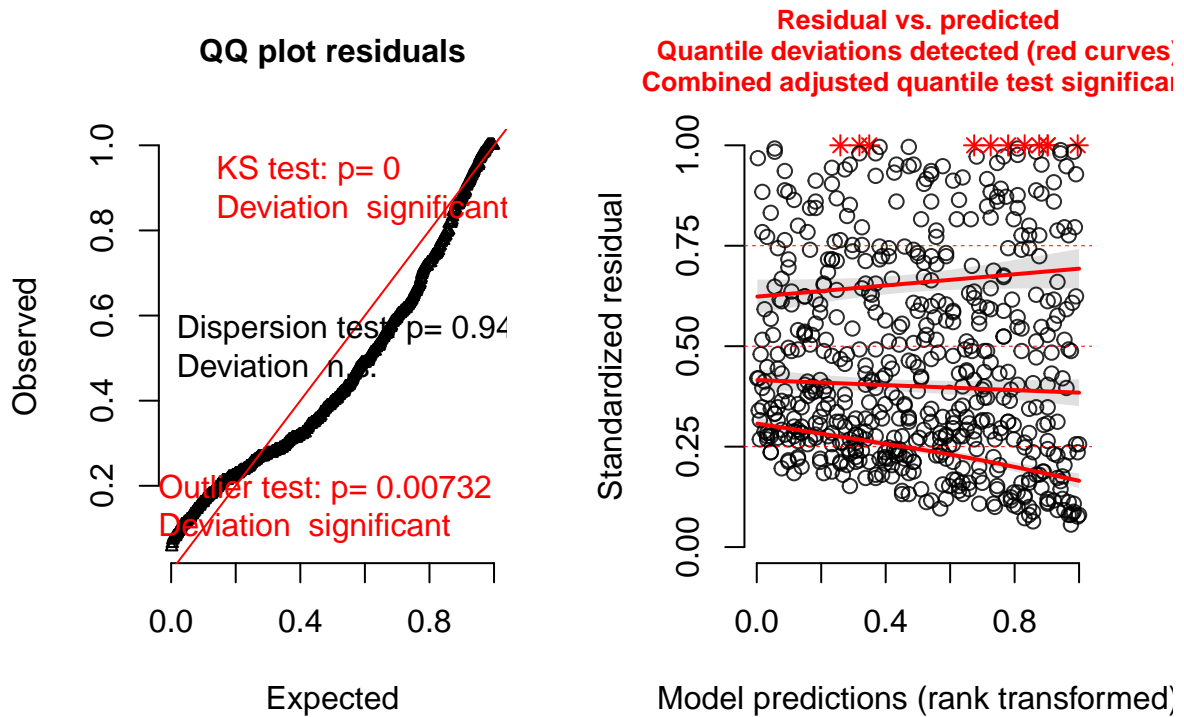


We see the same effect as with passing. The probability of jumping is inversely correlated to the probability of retreating and there is an obvious effect of presentation order for the control + eyes condition.

Freeze duration (reaction time) Freeze duration is a proxy of reaction time. It entails the detection, integration and decision making process until the behavioral response. We use this as a proxy of ease of recognition and potential certainty about the objects threat value. We will explore reactiontime for all five conditions. Attention: reactiontime is only present for trials in which spiders froze, thus the number of datapoints varies between conditions.

```
mreac <- glmmTMB(reactiontime ~ condition * test * day + (1|subject), family= gaussian, data= exp1)
simres <- simulateResiduals(mreac)
plot(simres)
```

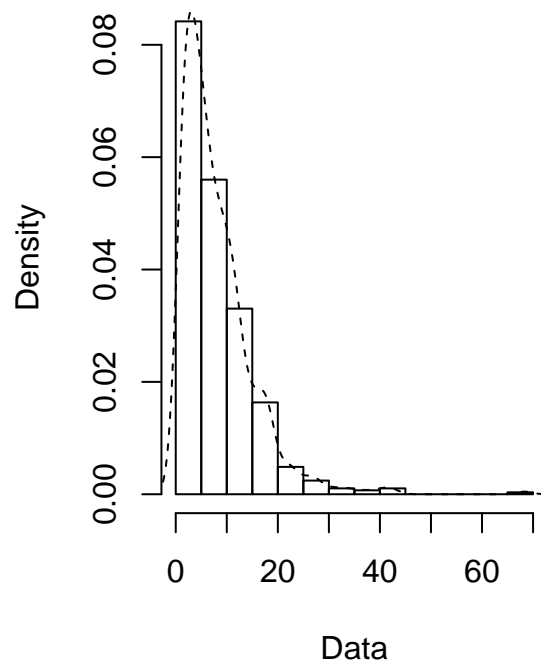
DHARMA residual diagnostics



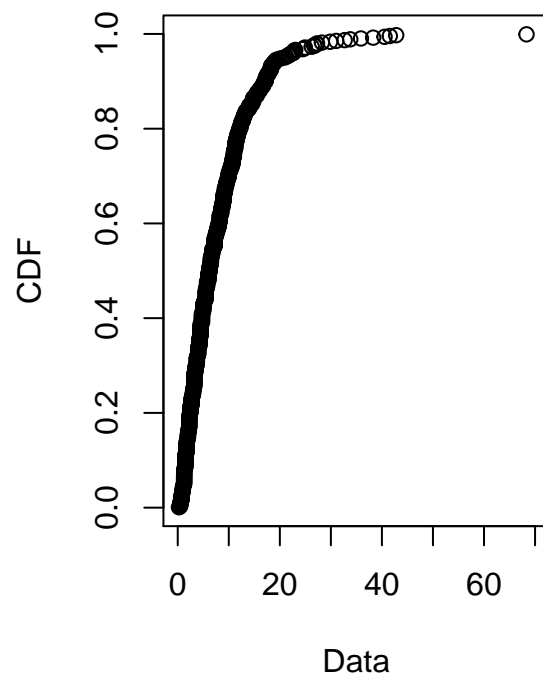
The distribution is not correct. We now check for the right distribution:

```
plotdist(exp1$reactiontime[!is.na(exp1$reactiontime)], histo = TRUE, demp = TRUE)
```

Empirical density

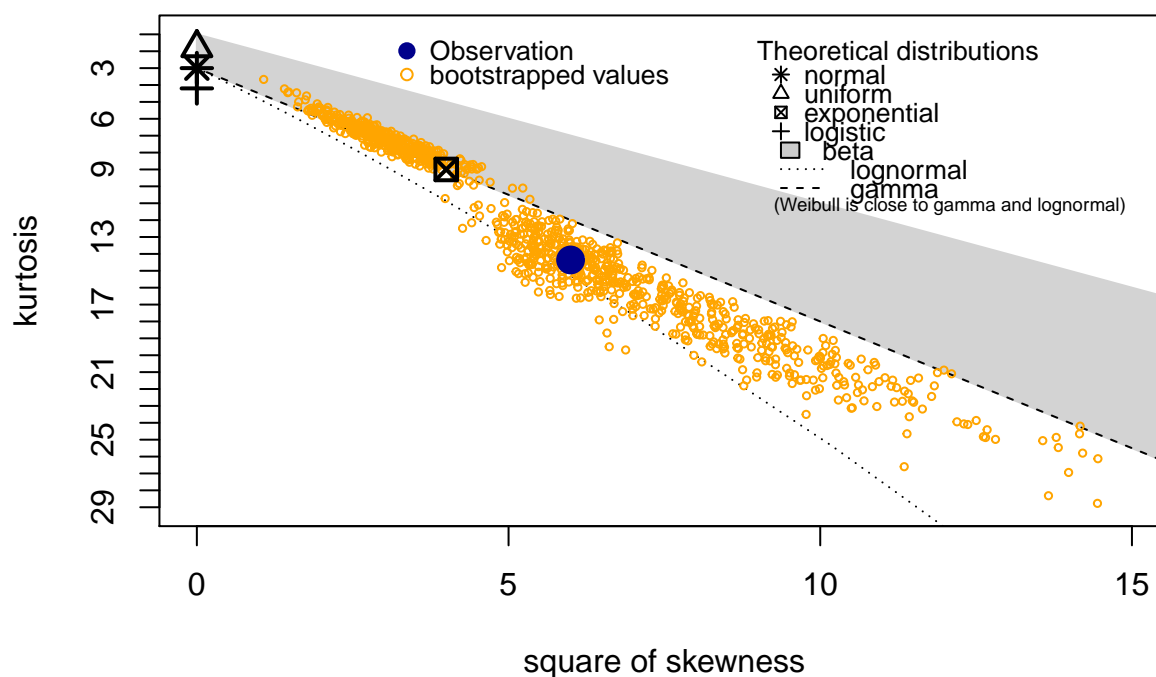


Cumulative distribution



```
descdist(exp1$reactiontime[!is.na(exp1$reactiontime)], boot = 1000)
```

Cullen and Frey graph

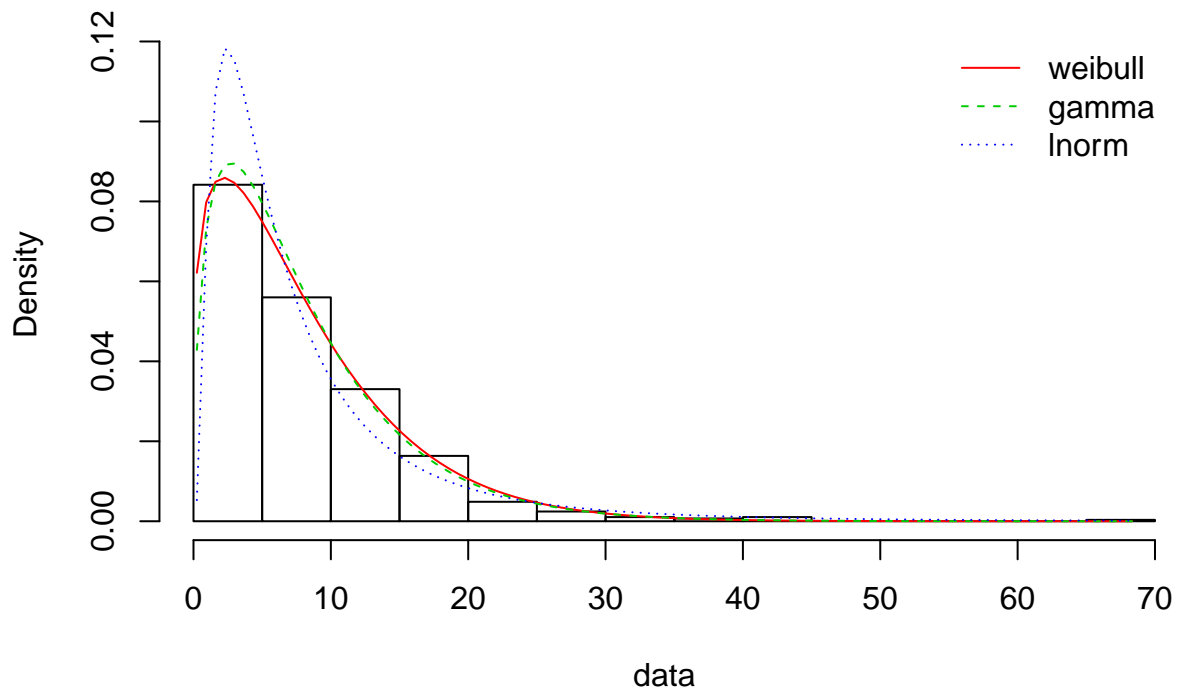


```
## summary statistics
## -----
## min: 0.244 max: 68.324
## median: 6.249
## mean: 8.094591
## estimated sd: 7.190214
## estimated skewness: 2.448497
## estimated kurtosis: 14.3594
```

```
lw <- fitdist(exp1$reactiontime[!is.na(exp1$reactiontime)], "weibull")
lg <- fitdist(exp1$reactiontime[!is.na(exp1$reactiontime)], "gamma")
lln <- fitdist(exp1$reactiontime[!is.na(exp1$reactiontime)], "lnorm")

denscomp(list(lw, lg, lln))
```

Histogram and theoretical densities



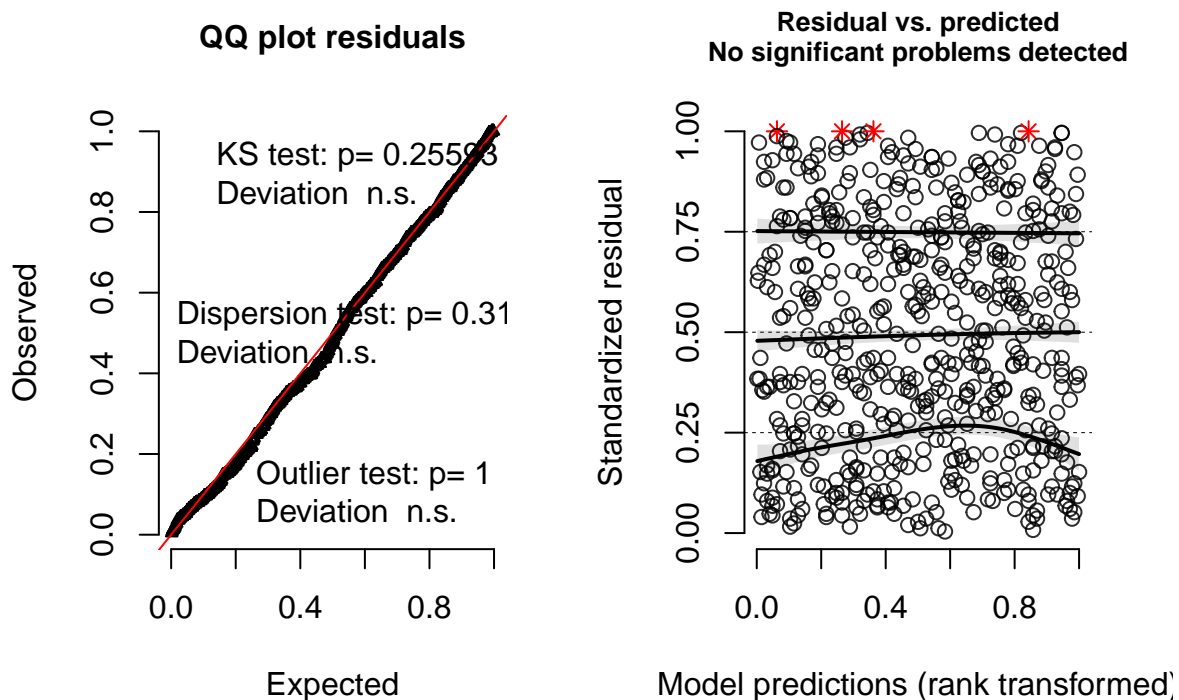
None of the distributions fit our data exactly, the most similar seems to be the lognormal, however is not a perfect fit. The best candidate seem to be Gamma. We will try to calculate the model with Gamma distribution.

```
mreac <- glmmTMB(reactiontime ~ condition + (1|subject), family= Gamma, data= exp1)
```

No model seem to be working. We have tried a lot of different ways of getting this to converge but without any luck. As a last resort we will drop the random effect. We know that this is not ideal, but there seems to be no other way to get any results. We advise caution in interpreting these results.

```
mreac <- glm(reactiontime ~ condition * test * day, family= Gamma, data= exp1)
simres <- simulateResiduals(mreac)
plot(simres)
```

DHARMA residual diagnostics



```
Anova(mreac)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: reactiontime
##          LR Chisq Df Pr(>Chisq)
## condition    23.3340  4  0.0001086 ***
## test          0.4682  1  0.4938204
## day           0.0831  1  0.7731334
## condition:test    2.1204  4  0.7136316
## condition:day   12.1458  4  0.0162995 *
## test:day        1.5010  1  0.2205123
## condition:test:day  6.5000  4  0.1647877
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We found an effect of condition and an interaction effect of condition and day. We now do a post-hoc test:

```
e <- emmeans(mreac, ~condition*day)
contrast(e, list(Blob1vs2 = c(1,0,0,0,0,-1,0,0,0,0),
  BlobEyes1vs2 = c(0,1,0,0,0,0,-1,0,0,0),
  Marp1vs2 = c(0,0,1,0,0,0,0,-1,0,0),
  Phid1vs2 = c(0,0,0,1,0,0,0,0,-1,0),
  Shiny1vs2 = c(0,0,0,0,1,0,0,0,0,-1),
  BlobVsBlobEyes = c(0.5,-0.5,0,0,0,0.5,-0.5,0,0,0),
```

```

BlobVsMarp = c(0.5,0,-0.5,0,0,0.5,0,-0.5,0,0),
BlobVsPhid = c(0.5,0,0,-0.5,0,0.5,0,0,-0.5,0),
BlobVsShiny = c(0.5,0,0,0,-0.5,0.5,0,0,0,-0.5),
BlobEyesVsMarp = c(0,0.5,-0.5,0,0,0,0.5,-0.5,0,0),
BlobEyesVsPhid = c(0,0.5,0,-0.5,0,0,0.5,0,-0.5,0),
BlobEyesVsShiny = c(0,0.5,0,0,-0.5,0,0.5,0,0,-0.5),
MarpVsPhid = c(0,0,0.5,-0.5,0,0,0,0.5,-0.5,0),
MarpVsShiny = c(0,0,0.5,0,-0.5,0,0,0.5,0,-0.5),
PhidVsShiny = c(0,0,0,0.5,-0.5,0,0,0,0.5,-0.5)),
adjust = 'bonferroni')

```

```

## contrast      estimate      SE  df z.ratio p.value
## Blob1vs2      -0.03920 0.0702 Inf  -0.559  1.0000
## BlobEyes1vs2  -0.06229 0.0296 Inf  -2.105  0.5292
## Marp1vs2       0.02488 0.0152 Inf   1.641  1.0000
## Phid1vs2      -0.02896 0.0157 Inf  -1.848  0.9699
## Shiny1vs2      0.02900 0.0208 Inf   1.395  1.0000
## BlobVsBlobEyes 0.03112 0.0381 Inf   0.817  1.0000
## BlobVsMarp     0.05394 0.0359 Inf   1.503  1.0000
## BlobVsPhid     0.04412 0.0360 Inf   1.227  1.0000
## BlobVsShiny    -0.00127 0.0366 Inf  -0.035  1.0000
## BlobEyesVsMarp 0.02281 0.0166 Inf   1.372  1.0000
## BlobEyesVsPhid 0.01300 0.0167 Inf   0.776  1.0000
## BlobEyesVsShiny -0.03239 0.0181 Inf  -1.791  1.0000
## MarpVsPhid     -0.00981 0.0109 Inf  -0.900  1.0000
## MarpVsShiny    -0.05520 0.0129 Inf  -4.290  0.0003
## PhidVsShiny    -0.04539 0.0130 Inf  -3.486  0.0073
##
## Note: contrasts are still on the inverse scale
## P value adjustment: bonferroni method for 15 tests

```

```

e <- emmeans(mreac, ~condition, type= "response")
toplotreac <- as.data.frame(e)

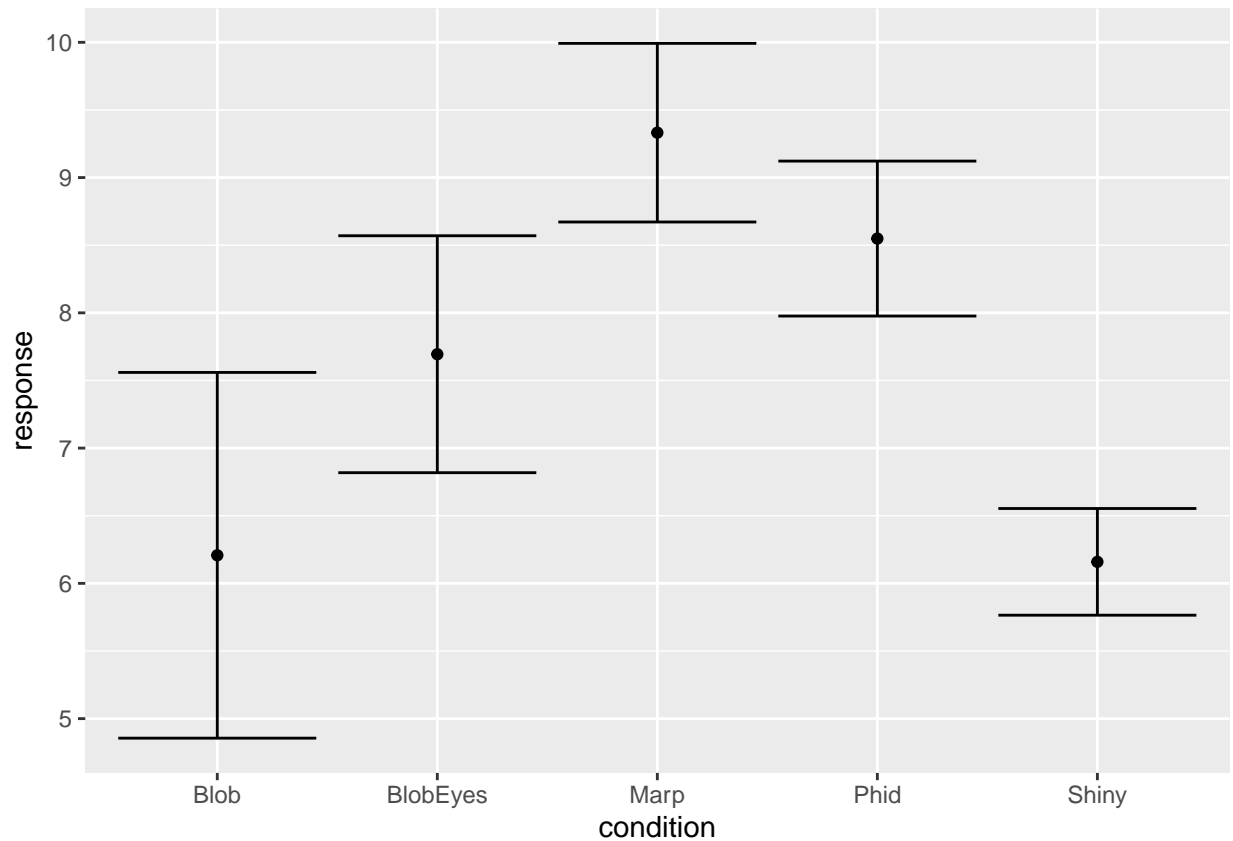
```

We now plot the reaction times for all conditions:

```

ggplot(toplotreac, aes(x = condition, y = response))+
  geom_point()+
  geom_errorbar(aes(ymin = response-SE, ymax = response+SE))

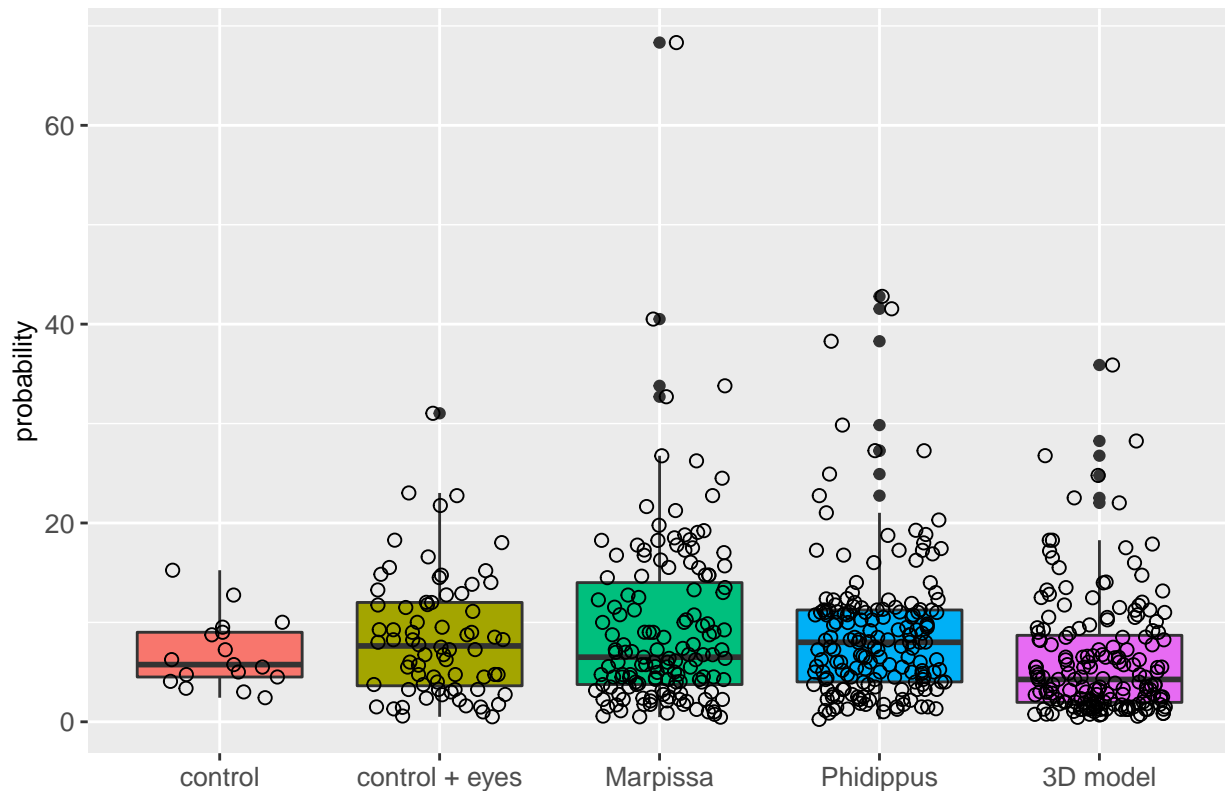
```



Spiders show the longest freezes for Marpissa and Phidippus. Due the likely fewer data points in the Blob (control) condition, there is quite a big variance in the reaction time. We plot the original data points to better visualize this:

```
ggplot(exp1, aes(condition, reactiontime, fill= condition))+
  geom_boxplot()+
  geom_jitter(height = 0, width= 0.3, cex = 2, line = 2, shape=1)+
  labs(title = "Reaction time", y= "probability")+
  scale_x_discrete(labels = c("control", "control + eyes", "Marpissa",
                              "Phidippus", "3D model"))+
  theme(axis.title.x = element_blank(), axis.text.x = element_text(size=10),
        axis.text.y = element_text(size=10), axis.title.y = element_text(size=10),
        plot.title = element_text(size= 15, face= "bold"), legend.position = "none")
```


Reaction time



As expected, there are only few data points for the control condition. The most interesting observation may be the shorter freezes for the 3D model compared to Marpissa and Phidippus.

Experiment 2 - blob vs faceless vs shiny

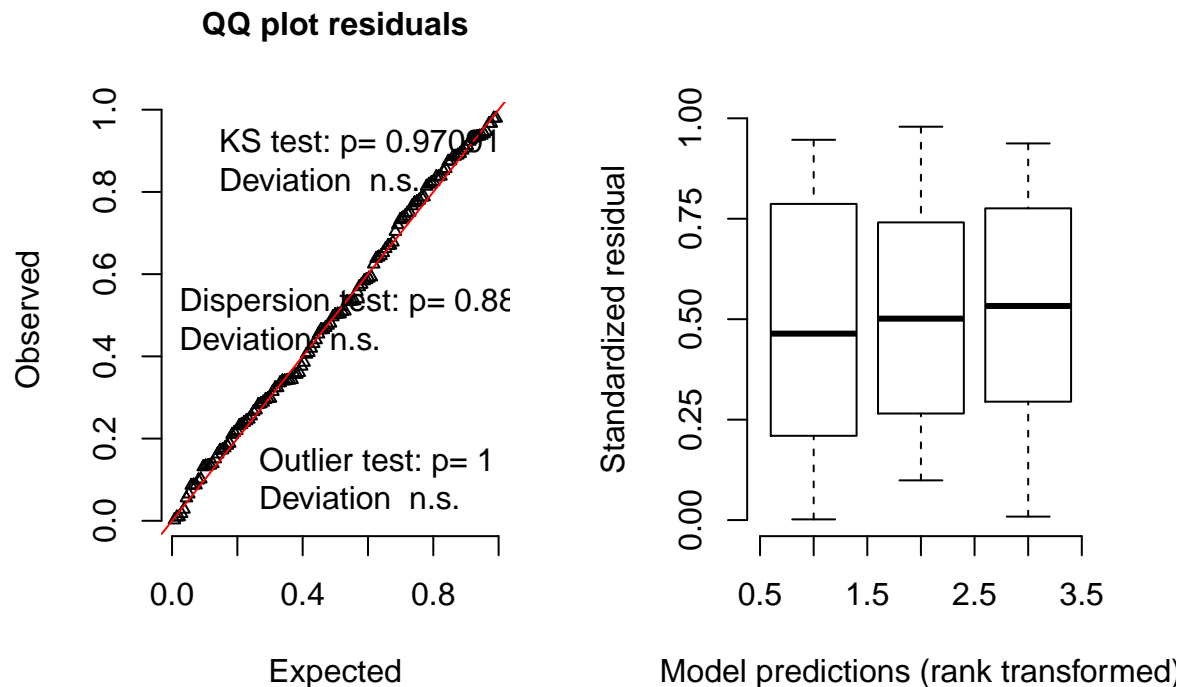
In experiment 2, spiders were tested using 3 different objects (Blob = control, Faceless = 3D model w/o eyes, Shiny = 3D model).

Preliminary analysis

All preliminary analyses will use passing as dependent variable as we consider it to be the more robust and objective measure of recognition of the target. We will test all variables against passing regardless of condition. Not sex, because all tests in exp2 were conducted with adult females.

```
mdate2 <- glmmTMB(passing~date + (1|subject), data = exp2, family=binomial)
simres <- simulateResiduals(mdate2)
plot(simres)
```

DHARMA residual diagnostics



Date

`Anova(mdate2)`

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: passing
##      Chisq Df Pr(>Chisq)
## date 0.4371  2    0.8037
```

Hooray - correct distribution!

Passing the target exp2 We calculate the model for the variable passing exactly as for experiment 1.

```
mret2 <- glmmTMB(passing ~ condition * test + (1|subject), data= exp2, family= binomial)
```

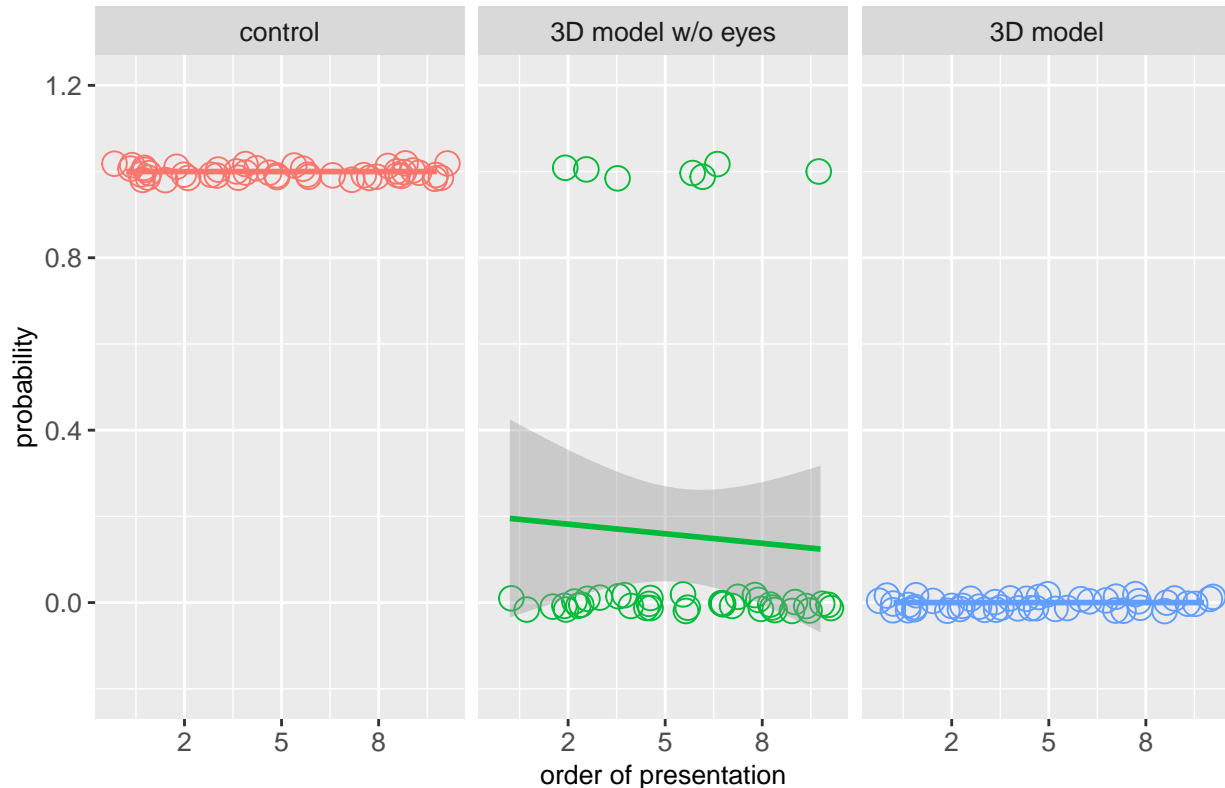
presumably because the data is completely separated for the conditions blob and shiny (meaning no spider passed Shiny but all passed Blob), the model fails to converge. We now plot the data to visualize passing for the three conditions in experiment 2.

```
new.labels2 <- c("Blob" = "control", "Faceless" = "3D model w/o eyes", "Shiny" = "3D model")

ggplot(exp2, aes(x=test,y=passing,color=condition))+
  facet_grid(~condition, labeller = labeller(condition = new.labels2))+
  geom_jitter(width = 0.5, height = 0.02,shape = 1, size=4)+
  geom_smooth(method="glm")+
```

```
ylim(-0.2,1.2)+
  scale_x_continuous(labels = scales::number_format(accuracy = 1))+
  labs(title = "Probability to pass depending on order of presentation", x = "order of presentation", y =
  theme(strip.text = element_text(size=10), axis.text.x = element_text(size=10),
        axis.text.y = element_text(size=10), axis.title.x = element_text(size=10),
        axis.title.y = element_text(size=10), plot.title = element_text(size= 15, face= "bold"),
        legend.position = "none")
```

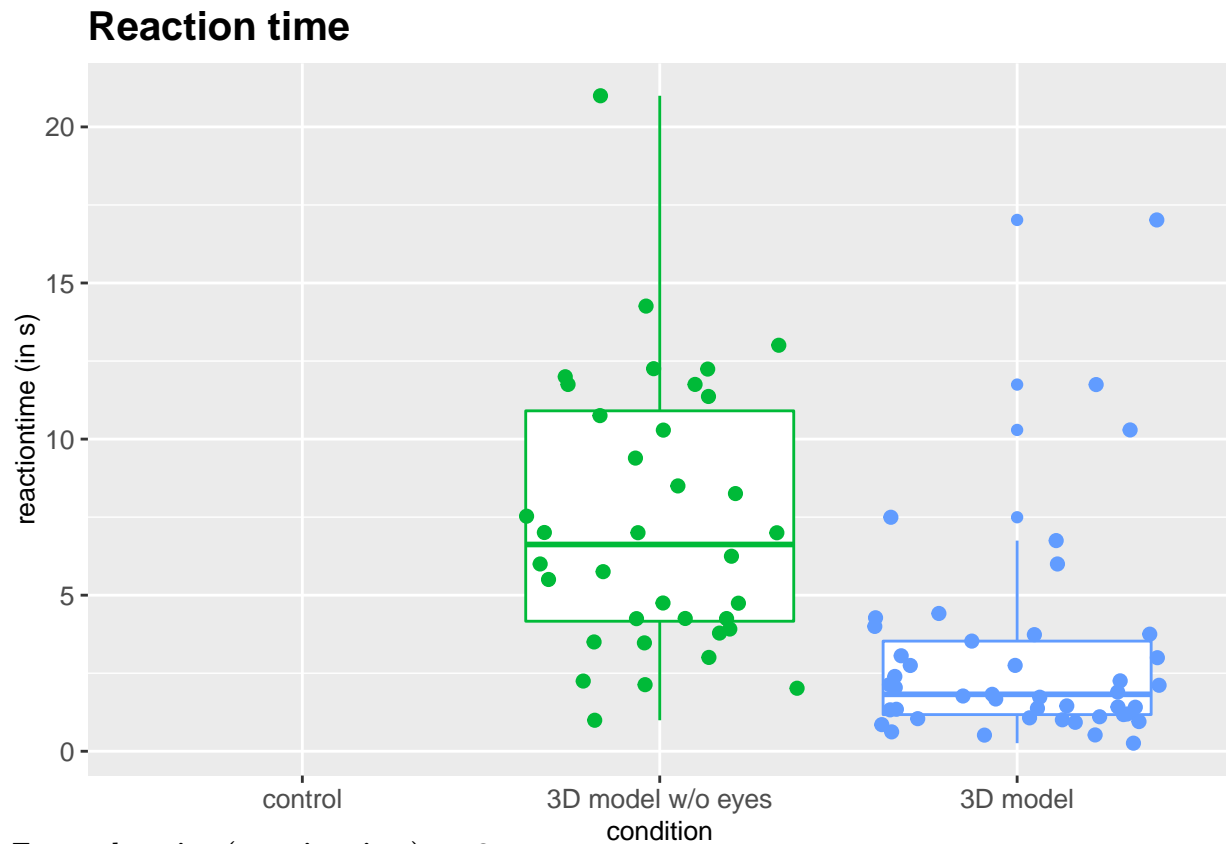
Probability to pass depending on order of presentation



Our hypothesis of data separation is confirmed. We subsequently will not be able to calculate a model for the second experiment based on passing. To get a better idea of the difference between the conditions Faceless and Shiny we will now plot reactiontime for the three conditions of experiment 2. Attention! Reaction time is directly linked to freezes, when there is no freeze, there is no reactiontime, meaning there is no data for the control condition.

```
ggplot(exp2, aes(condition, reactiontime, color= condition))+
  geom_boxplot()+
  geom_jitter(cex= 2)+
  ggtitle("Reaction time")+
  labs(y = "reactiontime (in s)")+
  scale_x_discrete(labels = c("control", "3D model w/o eyes", "3D model"))+
  theme(strip.text = element_text(size=10), axis.text.x = element_text(size=10),
        axis.text.y = element_text(size=10), axis.title.x = element_text(size=10),
```

```
axis.title.y = element_text(size=10), plot.title = element_text(size= 15, face= "bold"),
legend.position = "none")
```



Freeze duration (reaction time) exp2

We can see a clear difference of reaction time (freezing time) between the 3D model without eyes and the 3D model. Spiders freeze more than twice as long for the 3D model without eyes, implying that the recognition process is hindered by the lack of eye features.

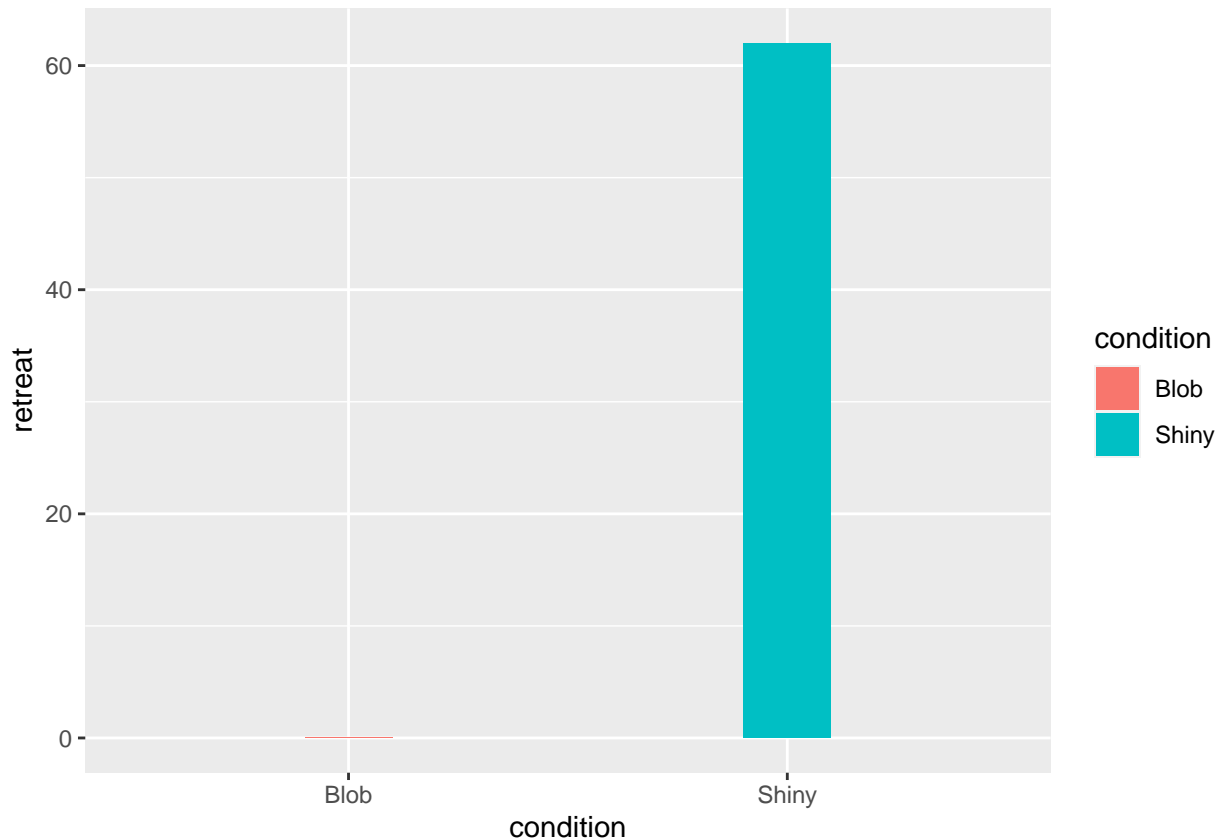
Spiderlings

```
spid <- read.csv("C:/Users/roess/Desktop/Publications/PredRecognition/Spiderlings_Innateness/Spiderlings.csv")
look <- subset(spig, looking == "1")
```

```
tapply(look$retreat, look$condition, table)
```

```
## $Blob
##
## 0
## 67
##
## $Shiny
##
## 0 1
## 11 62
```

```
ggplot(spid, aes(y = retreat, fill = condition))+
  geom_bar(stat = "identity", aes(x = condition), width = .2)+
  ggsave(filename = "spiderlings.svg")
```



```
colMeans(spid[sapply(spid, is.numeric)])
```

```
##      Mom      trial starttime    endtime    retreat
## 38.65625  2.00000      NA      NA      NA
## freezing approach looking lookingamount passing
##      NA      NA      NA      NA      NA
## reactiontime
##      NA
```

```
mean(spid$reactiontime[!is.na(spid$reactiontime)])
```

```
## [1] 5.904323
```

```
mmom <- glmmTMB(retreat ~ Mom + (1|subject), data = spid, family = binomial)
Anova(mmom)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: retreat
##      Chisq Df Pr(>Chisq)
## Mom 0.8042 1 0.3698
```

```
mtrial <- glmmTMB(retreat ~ trial + (1|subject), data = spid, family = binomial)
Anova(mtrial)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: retreat
##           Chisq Df Pr(>Chisq)
## trial 0.0164  1      0.8982
```