

# Statapult

E. Croffoot-Suede, S. Kim

4/6/2022

## Designs

There were two possible designs for the data of this form: a  $2_V^{5-1}$  design, and a  $2_{III}^{5-2}$  design.

### The Resolution III Design

The  $2_{III}^{5-2}$  design is a quarter-fraction design which uses five factors over 8 runs. This design uses the constraints  $D \equiv AB$  and  $E \equiv AC$ . Its defining relation is  $I \equiv ABD \equiv ACE \equiv BCDE$ . We will refer to this resolution III design as design 1, notated as  $d_1$ .

### The Resolution V Design

The  $2_V^{5-1}$  design is a half-fraction design which uses five factors over 16 runs. This design uses the constraint  $E \equiv ABCD$ . Its defining relation is  $I \equiv ABCDE$ . We will refer to this resolution V design as design 2, notated as  $d_2$ .

## Minimum Aberration Model Robustness

Minimum aberration is a model robustness metric used to select a design shape for a dataset. It uses the wordlength vectors from two or more designs to determine the model with the minimum aberration. Less aberration is desirable.

The wordlength for Design 1 is  $W = (0, 1, 0)$ , whereas the wordlength for Design 2 is  $W = (2, 1, 0)$ , based on the defining relation for each of the designs.

Given the  $d_1$  design of  $2^{5-2}$  (quarter-fraction) and the  $d_2$  design of  $2^{5-1}$  (half-fraction), let  $r$  be the smallest integer such that  $A_r(d_1) \neq A_r(d_2)$ , where  $A_r(\cdot)$  is one element of the design's wordlength pattern.

Design  $d_1$  has less aberration than design  $d_2$ , since  $A_r(d_1) = 0 < 2 = A_r(d_2)$ . Therefore,  $d_1$  is desired over  $d_2$  due to its lower aberration.

## Clear and Strong Clear Effects for Model Robustness

Determining model clear and strong clear factors can also be used to determine model robustness. It uses the number of clear effects and strong clear effects from two or more designs to determine the model with the most strong clear, then the most clear effects. More strong clear effects are desirable. If all models have the same number of strong clear effects or no models have any strong clear effects, then more clear effects are desirable.

We define a main effect or two-factor interaction as clear if none of its aliases are main effects or two-factor interactions, and we define it as strong clear if none of its aliases are main effects, two-factor interactions, or three-factor interactions.

To determine strong clear and clear effects, we consider the alias structure. Design  $d_1$  has the defining relation  $I \equiv ABD \equiv ACE \equiv BCDE$ . We get the following aliases with two-factor interactions.

$$A \equiv BD \equiv CE (\equiv BCDE)$$

$$B \equiv AD (\equiv ACE \equiv CDE)$$

$$C \equiv AE (\equiv ABCD \equiv BDE)$$

$$D \equiv AB (\equiv ACDE \equiv BCE)$$

$$E \equiv AC (\equiv ABDE \equiv BCD)$$

Observe how all main effects have an alias with a two-factor interaction. Therefore, none of the effects are strong clear.

Design  $d_2$  has the defining relation  $I \equiv ABCDE$ . For each main effect, there are no aliases with two- or three-factor interactions. Therefore, each main effect is considered strong clear.

Therefore, design  $d_2$  is more desirable due to its high number of strong clear factors.

Fries and Hunter (1980) established that not all  $2^{k-p}$  designs of maximum resolution are equally desirable, and they introduced the minimum aberration criterion for further discriminating designs of the same resolution.

## Data

We began by collecting our data. The data were collected on April 7, 2022 using a “statapult.” Out of six possible factors, we chose to analyze five of them, leaving the last factor (POST) at the low level for all runs. The factors are as follows.

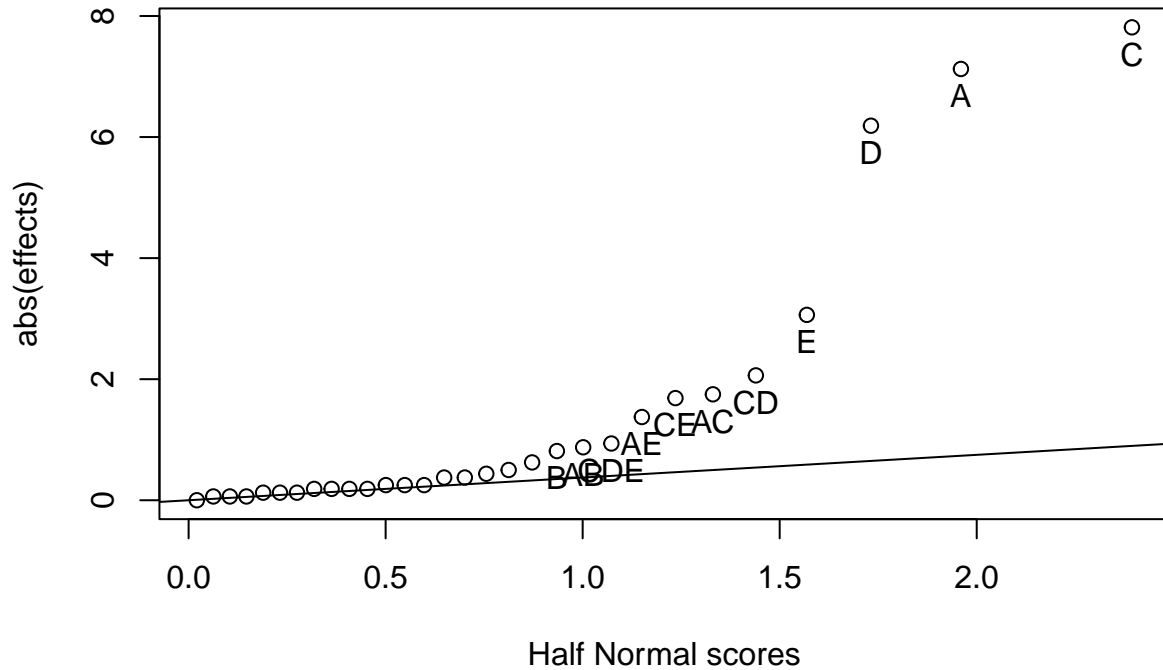
- **arm** - rubber band attachment point on the arm: 1, 3
- **ball** - ball type: ping-pong ball (low), wiffle ball (high)
- **position** - ball position: 1, 3
- **height** - arm draw back height: 1, 3
- **tilt** - catapult tilt: 1, 3

The last factor (**post**) was the rubber band attachment point on the post. It was left at the high level (1) for all runs.

## Analysis

We started by analyzing the full  $2^5$  fractional design. To determine significant effects, we used a half-normal plot on the full dataset with no replicates.

```
myout = lm(distance ~ A*B*C*D*E, data = statapult)
effects = myout$coefficients[2:31]
halfnorm(effects, labs=names(effects), alpha = 0.10, refline = "TRUE")
```



```
## zscore= 0.02089009 0.06270678 0.1046335 0.146745 0.1891184 0.2318344 0.2749777
0.3186394 0.3629173 0.4079187 0.4537622 0.5005801 0.5485223 0.5977601
0.6484922 0.7009514 0.755415 0.8122178 0.871771 0.9345893 1.001331 1.072861
1.150349 1.23544 1.330562 1.439531 1.56892 1.731664 1.959964 2.39398
effp=
1.251781e-15 0.0625 0.0625 0.0625 0.125 0.125 0.125 0.1875 0.1875 0.1875
0.1875 0.25 0.25 0.25 0.375 0.375 0.4375 0.5 0.625 0.8125 0.875 0.9375 1.375
1.6875 1.75 2.0625 3.0625 6.1875 7.125 7.8125
```

Based on the half-normal plot, we determined that all main effects  $A$ ,  $B$ ,  $C$ ,  $D$ , and  $E$  ought to be put into the model, along with the two-way interactions  $AB$ ,  $AC$ ,  $AE$ ,  $CD$ , and  $CE$ , and the three-way interaction  $CDE$ . The three-way interaction term includes the main effects  $C$ ,  $D$ , and  $E$ , and the two-way interactions  $AC$ ,  $AE$ ,  $CD$ , and  $CE$ . The two-way interaction term  $AB$  includes the main effects  $A$ , and  $B$ .

The resulting full model with all terms is

$$Y_{\text{Distance}} = C * D * E + A * B + A * E + A * C + C * E$$

To analyze the data, we start by testing for overall significance using a F-test for the model. To do so, we fit two linear models: one with the response (`distance`) by the full model and one with the response by 1 to make our null model.

```
stata.full.model = lm(distance ~ C*D*E + A*B + A*E + A*C + C*E, data = statapult)
stata.red.model = lm(distance ~ 1, data = statapult)
```

Then, we tested the full model against the null model using analysis of variance.

```
anova(stata.red.model, stata.full.model)
```

```
## Analysis of Variance Table
##
## Model 1: distance ~ 1
## Model 2: distance ~ C * D * E + A * B + A * E + A * C + C * E
##   Res.Df    RSS Df Sum of Sq    F    Pr(>F)
## 1      31 5610.9
## 2      19   47.4 12    5563.5 185.94 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We found that the full model is significantly different with a  $p$ -value of at least  $2.2 \times 10^{-16}$ .

Based on this, we ran a F-test for three-way interactions.

```
anova(stata.full.model)
```

```
## Analysis of Variance Table
##
## Response: distance
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## C           1 1953.13 1953.13  783.3113 < 2.2e-16 ***
## D           1 1225.12 1225.12  491.3430 4.873e-15 ***
## E           1  300.12  300.12  120.3668 1.159e-09 ***
## A           1 1624.50 1624.50  651.5145 3.629e-16 ***
## B           1   21.13   21.13    8.4723 0.008965 **
## C:D          1  136.13  136.13   54.5937 5.345e-07 ***
## C:E          1   91.13   91.13   36.5462 8.156e-06 ***
## D:E          1    1.12    1.12    0.4512 0.509857
## A:B          1   24.50   24.50    9.8259 0.005457 **
## E:A          1   60.50   60.50   24.2639 9.384e-05 ***
## C:A          1   98.00   98.00   39.3034 5.094e-06 ***
## C:D:E        1   28.13   28.13   11.2797 0.003300 **
## Residuals  19   47.38    2.49
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We found that the three-way interaction  $C * D * E$  was significantly different with a  $F$ -value of 11.2797 on one degree of freedom and a  $p$ -value of  $p = 0.003300 < 0.05 = \alpha$ .

We found that the two-way interaction  $A * B$  was significantly different with a  $F$ -value of 9.8259 on one degree of freedom and a  $p$ -value of  $p = 0.005457 < 0.05 = \alpha$ .

Based on this, we generated contrasts on the three-way interaction cells.

```
lsm_CDE.int <- lsmeans(stata.full.model , ~ C:D:E)
summary(contrast(lsm_CDE.int,
                 method = "pairwise",
                 adjust = "Bonferroni"),
        infer = c(T,T),
        level = .95,
        side = "two.sided")
```

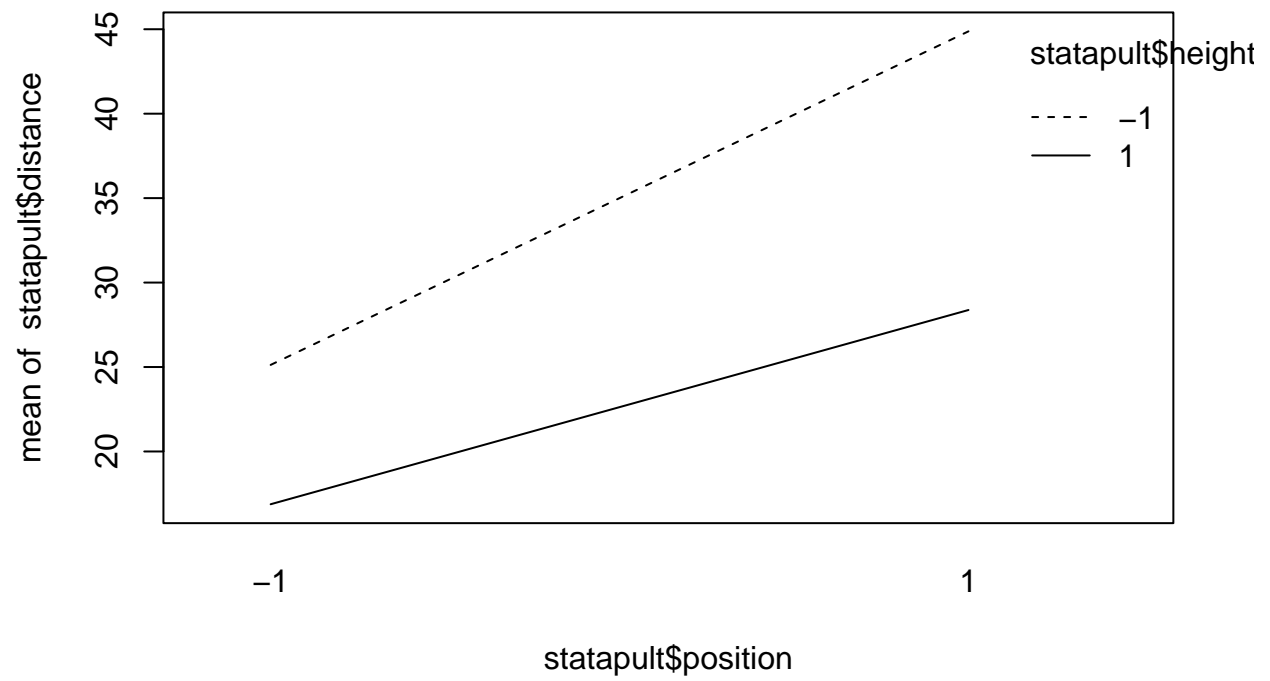
```
##   contrast                estimate    SE df lower.CL upper.CL t.ratio p.value
## (-1 -1 -1) - (1 -1 -1)      -25.00 1.12 19  -29.052  -20.95  -22.390 <.0001
## (-1 -1 -1) - (-1 1 -1)       6.75 1.12 19   2.698   10.80   6.045 0.0002
## (-1 -1 -1) - (1 1 -1)       -6.25 1.12 19  -10.302   -2.20  -5.598 0.0006
## (-1 -1 -1) - (-1 -1 1)       1.25 1.12 19   -2.802    5.30   1.120 1.0000
```

```
## (-1 -1 -1) - (1 -1 1) -13.25 1.12 19 -17.302 -9.20 -11.867 <.0001
## (-1 -1 -1) - (-1 1 1) 11.00 1.12 19 6.948 15.05 9.852 <.0001
## (-1 -1 -1) - 1 1 1 1.00 1.12 19 -3.052 5.05 0.896 1.0000
## (1 -1 -1) - (-1 1 -1) 31.75 1.12 19 27.698 35.80 28.436 <.0001
## (1 -1 -1) - (1 1 -1) 18.75 1.12 19 14.698 22.80 16.793 <.0001
## (1 -1 -1) - (-1 -1 1) 26.25 1.12 19 22.198 30.30 23.510 <.0001
## (1 -1 -1) - (1 -1 1) 11.75 1.12 19 7.698 15.80 10.523 <.0001
## (1 -1 -1) - (-1 1 1) 36.00 1.12 19 31.948 40.05 32.242 <.0001
## (1 -1 -1) - 1 1 1 26.00 1.12 19 21.948 30.05 23.286 <.0001
## (-1 1 -1) - (1 1 -1) -13.00 1.12 19 -17.052 -8.95 -11.643 <.0001
## (-1 1 -1) - (-1 -1 1) -5.50 1.12 19 -9.552 -1.45 -4.926 0.0026
## (-1 1 -1) - (1 -1 1) -20.00 1.12 19 -24.052 -15.95 -17.912 <.0001
## (-1 1 -1) - (-1 1 1) 4.25 1.12 19 0.198 8.30 3.806 0.0334
## (-1 1 -1) - 1 1 1 -5.75 1.12 19 -9.802 -1.70 -5.150 0.0016
## (1 1 -1) - (-1 -1 1) 7.50 1.12 19 3.448 11.55 6.717 0.0001
## (1 1 -1) - (1 -1 1) -7.00 1.12 19 -11.052 -2.95 -6.269 0.0001
## (1 1 -1) - (-1 1 1) 17.25 1.12 19 13.198 21.30 15.449 <.0001
## (1 1 -1) - 1 1 1 7.25 1.12 19 3.198 11.30 6.493 0.0001
## (-1 -1 1) - (1 -1 1) -14.50 1.12 19 -18.552 -10.45 -12.986 <.0001
## (-1 -1 1) - (-1 1 1) 9.75 1.12 19 5.698 13.80 8.732 <.0001
## (-1 -1 1) - 1 1 1 -0.25 1.12 19 -4.302 3.80 -0.224 1.0000
## (1 -1 1) - (-1 1 1) 24.25 1.12 19 20.198 28.30 21.718 <.0001
## (1 -1 1) - 1 1 1 14.25 1.12 19 10.198 18.30 12.762 <.0001
## (-1 1 1) - 1 1 1 -10.00 1.12 19 -14.052 -5.95 -8.956 <.0001
##
## Results are averaged over the levels of: A, B
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 28 estimates
## P value adjustment: bonferroni method for 28 tests
```

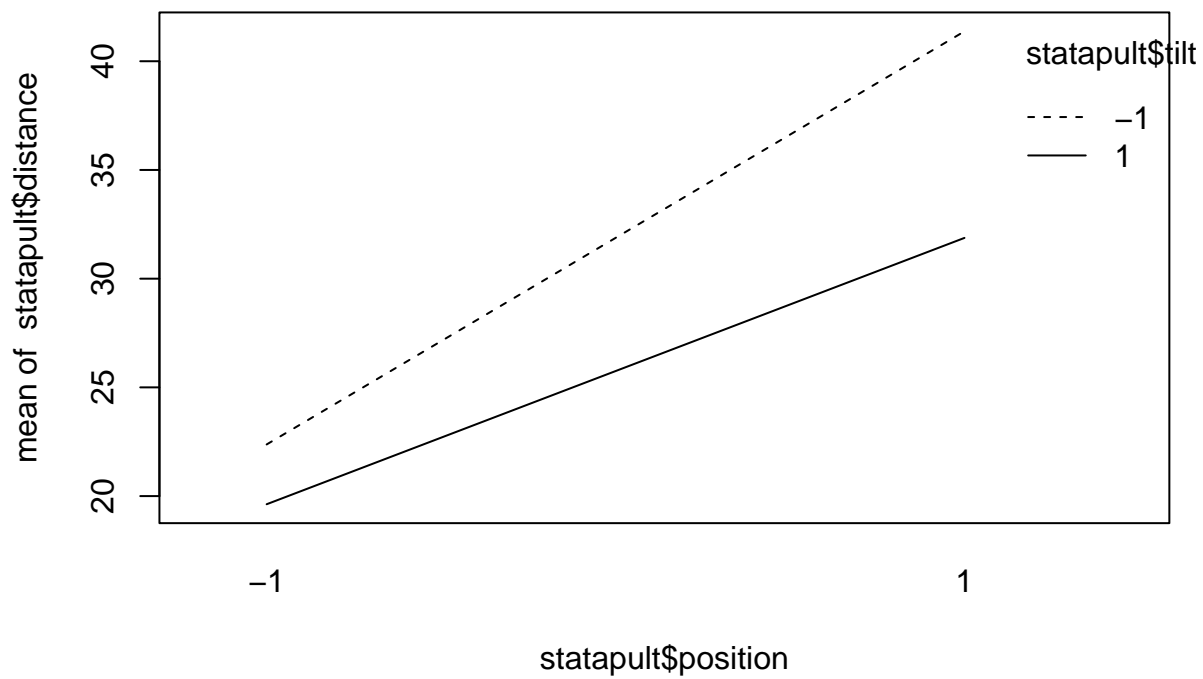
We chose to use Bonferroni's correction method on the contrasts because of its higher power on smaller numbers of means, especially compared to Tukey's correction method.

Based on this, we found that to maximize distance, we ought to keep position (C) high, tilt (E) low, and height (D) low, based on the interaction plots between position, tilt, and height.

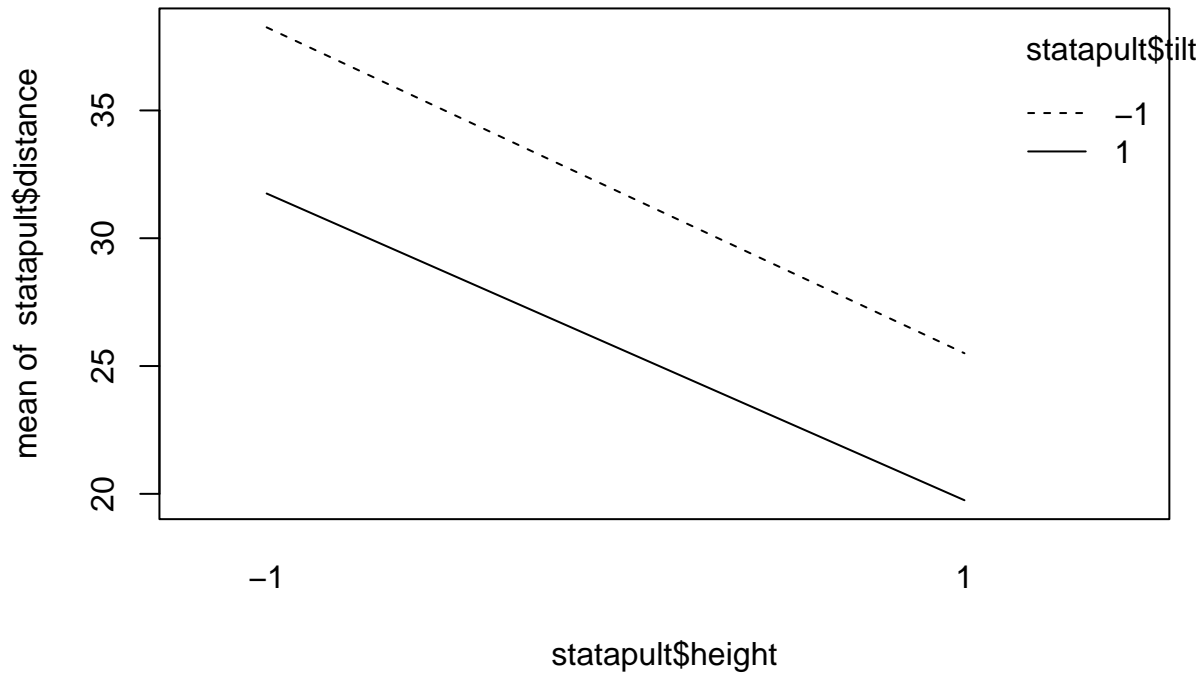
```
interaction.plot(statapult$position, statapult$height, statapult$distance)
```



```
interaction.plot(statapult$position, statapult$tilt, statapult$distance)
```



```
interaction.plot(statapult$height, statapult$tilt, statapult$distance)
```



Since all of the  $p$ -values for these contrasts are below our  $\alpha = 0.05$ , we also determined that this combination of position at the high-level, and tilt and height at the low-level were significantly different from the other combinations.

We also generated contrasts for the two-way  $A * B$  cell. Similarly, we chose to use Bonferroni's correction method on the contrasts over Tukey's correction method.

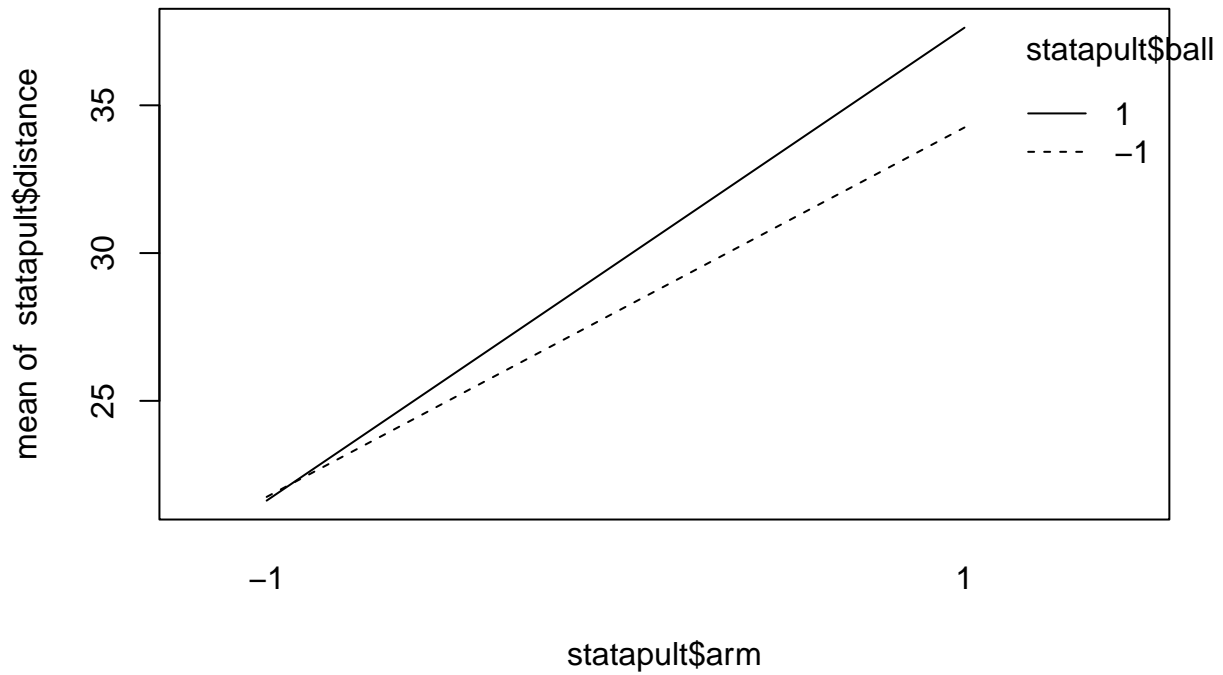
```
lsm_AB.int <- lsmeans(stata.full.model , ~ A:B)
summary(contrast(lsm_AB.int,
  method = "pairwise",
  adjust = "Bonferroni"),
  infer = c(T,T),
  level = .95,
  side = "two.sided")
```

```
## contrast      estimate    SE df lower.CL upper.CL t.ratio p.value
## (-1 -1) - (1 -1)  -12.500 0.79 19    -14.8    -10.18  -15.832  <.0001
## (-1 -1) - (-1 1)   0.125 0.79 19     -2.2     2.45   0.158  1.0000
## (-1 -1) - 1 1     -15.875 0.79 19    -18.2    -13.55 -20.107  <.0001
## (1 -1) - (-1 1)   12.625 0.79 19     10.3    14.95  15.991  <.0001
## (1 -1) - 1 1      -3.375 0.79 19     -5.7    -1.05  -4.275  0.0025
## (-1 1) - 1 1     -16.000 0.79 19    -18.3    -13.68 -20.265  <.0001
##
## Results are averaged over the levels of: C, D, E
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 6 estimates
## P value adjustment: bonferroni method for 6 tests
```



Based on this, we found that to maximize distance, we ought to keep ball and arm at the high level to achieve the highest distance based on the interaction plot.

```
interaction.plot(statapult$arm, statapult$ball, statapult$distance)
```



Since all of the  $p$ -values for these contrasts are below our  $\alpha = 0.05$ , we also determined that this combination of ball and arm at the high-level were significantly different from the other combinations.

## Discussion

## Conclusion