StimuliAndHeuristics

Alex Kale

1/8/2019

Task

We will show people uncertainty visualizations about the probability of different possible margins of victory in an imaginary game or election and ask them:

1. If this game/election happened 100 times, how many times would you expect team A to win?
2. Given $1 to bet on the outcome of this game/election, how much would you bet that team A wins?

Both of these questions assess the ability of the user to judge the common language effect size (CLES) or the proportion of probability mass which favors a victory for team A. However, the second question incorporates an incentive structure whereby the user must bet not too much and not too little, and the optimal bet depends on the odds of victory for team A. Note that in the current proposal team A always has a probability of victory which is less than or equal to 50%.

Incentive Structure

The user bets some portion of their $1 budget in each trial that team A will win the game/election. The payoff of the bet is proportional to the odds of the bet such that a bet on 1:1 odds yields a 50% chance of winnings double the bet amount, a bet on 2:1 odds yields a 33% chance of winnings tripple the bet amount, a bet on 4:1 odds yields a 20% chance of winnings quintupple the bet amount, etc.

Additionally, the amount that users win is subject to a tiered capital gains tax whereby each increment of 50 cents in winnings is taxed 10% more than the previous 50 cents, and all winnings over $2 are taxed 50%. This tiered tax imposes diminishing returns for excessively risky bets. The amount that users do not bet is subject to a flat tax of 25%, which imposes an incentive against risk aversion much like inflation encourages people to invest in the stock market.

The block of code below summarizes the incentive structure for the betting task. The line chart shows the expected utility (in $) of each bet amount for each level of odds of victory (shown in different lines). The peak of each line (representing the optimal bet for each level of odds) is highlighted with a red point. A key feature of this task design is that the expected utility for the bet is a linear function of the probability that team A wins the game (i.e., CLES). Thus, the proposed decision task relies on the same uncertainty information as the proposed perceptual judgment, allowing us to separately investigate perception and decision-making within one experiment.

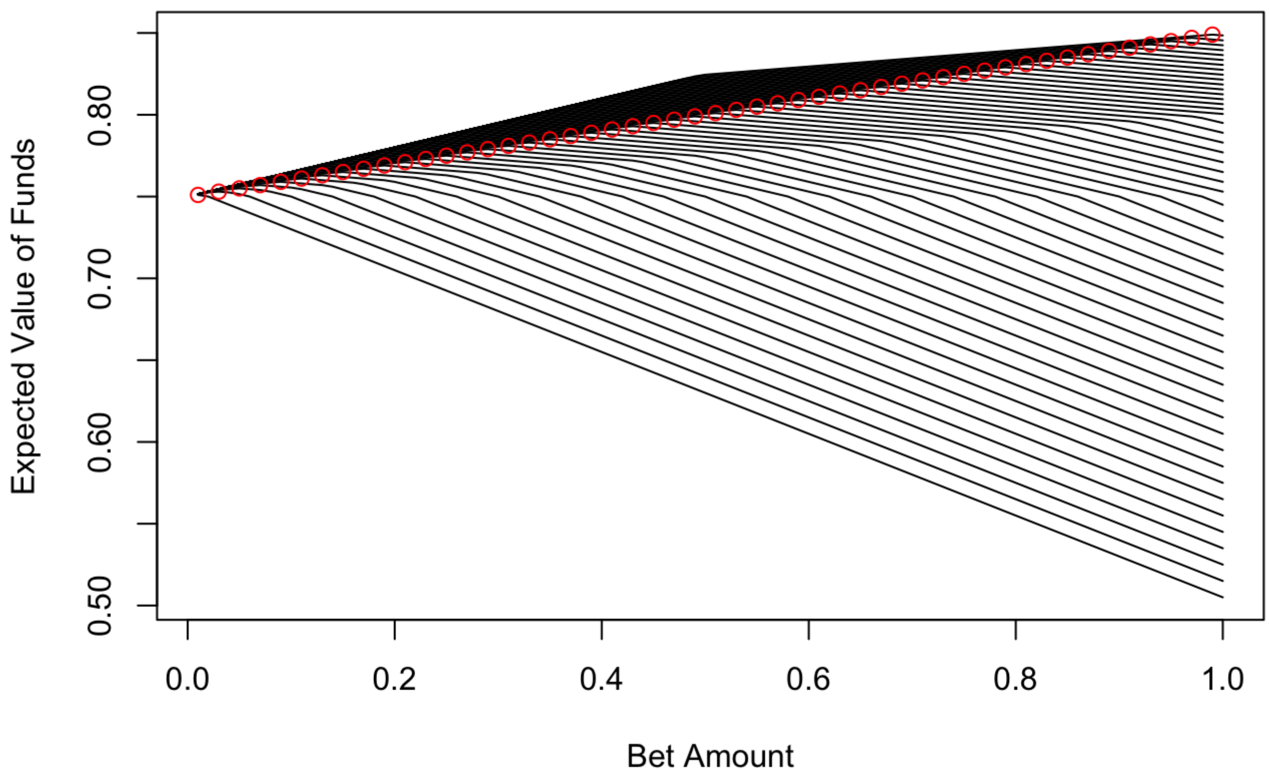
```r
# set range of possible bets based on given budget and minimum bet
budget <- 1
min_bet <- 0.01
possible_bets <- seq(from=min_bet, to=budget, by=0.01)

# create a tiered capital gains tax
tax_winnings <- function(winnings) {
  tiers <- append(seq(0, 2, by = 0.5), Inf)
  rates <- seq(0, .5, by = .1)
  taxed_winnings <- sum(diff(c(0, pmin(winnings, tiers))) * (1-rates))
  return(taxed_winnings)
}

# set cost of not betting
loss_rate <- 0.25

# calculate expected payout for different odds
odds_to_check <- ppoints(50)
payoff <- matrix(NaN, nrow = length(possible_bets), ncol = length(odds_to_check))
net_pay <- matrix(NaN, nrow = length(possible_bets), ncol = length(odds_to_check))
for (i in 1:length(possible_bets)) {
  for (j in 1:length(odds_to_check)) {
    payoff[i, j] <- odds_to_check[j] * tax_winnings(possible_bets[i] / odds_to_check[j])
    net_pay[i, j] <- (1 - loss_rate)*(budget - possible_bets[i]) + payoff[i, j]
  }
}

# plot expected payout for each possible bet
plot(possible_bets,net_pay[,1],type="l",xlab="Bet Amount",ylab="Expected Value of Funds"
,ylim=range(net_pay))
for (i in 2:length(odds_to_check)) {
  lines(possible_bets,net_pay[,i])
}
# determine the best bet at each level of odds
best_pay <- apply(net_pay, 2, max)
best_bets <- seq(-1, 0, length.out = length(best_pay))
for (i in 1:length(best_pay)) {
  best_bets[i] <- possible_bets[which(net_pay[,i]==best_pay[i])]
}
points(best_bets, best_pay, col = "red")
```



Reading Difference Distributions

Data for Stimuli

We will vary the mean and standard deviation of the distribution users are judging so that we measure different levels of uncertainty (e.g., low and high) and different odds of victory for team A. For instance, below we consider 1:1, 2:1, 4:1, 8:1, and 16:1 odds that team A will win. The point of these conditions is that the mean margin of victory for team A depends on both the variance and the odds of victory for team A, such that the location of the mean alone should provide a poor cue for the task.

Here's some data spanning the different combinations of conditions that I propose to test.

```
# set up conditions dataframe
condition <- c("low var, 1:1 odds", "low var, 2:1 odds", "low var, 4:1 odds", "low var,
 8:1 odds", "low var, 16:1 odds",
               "high var, 1:1 odds", "high var, 2:1 odds", "high var, 4:1 odds", "high v
ar, 8:1 odds", "high var, 16:1 odds")
std <- sort(rep(c(2,5),5)) # different levels of uncertainty about the margin of victory
odds <- rep(c(1/2,1/3,1/5,1/9,1/17),2) # probability of team A winning
conds_df <- data.frame(
    "condition"=condition,
    "sd"=std,
    "odds_of_victory"=odds
)

# add column for the mean
conds_df$mean <- - (conds_df$sd * qnorm(conds_df$odds_of_victory)) # see https://www.joh
ndcook.com/quantiles_parameters.pdf

# print
conds_df
```

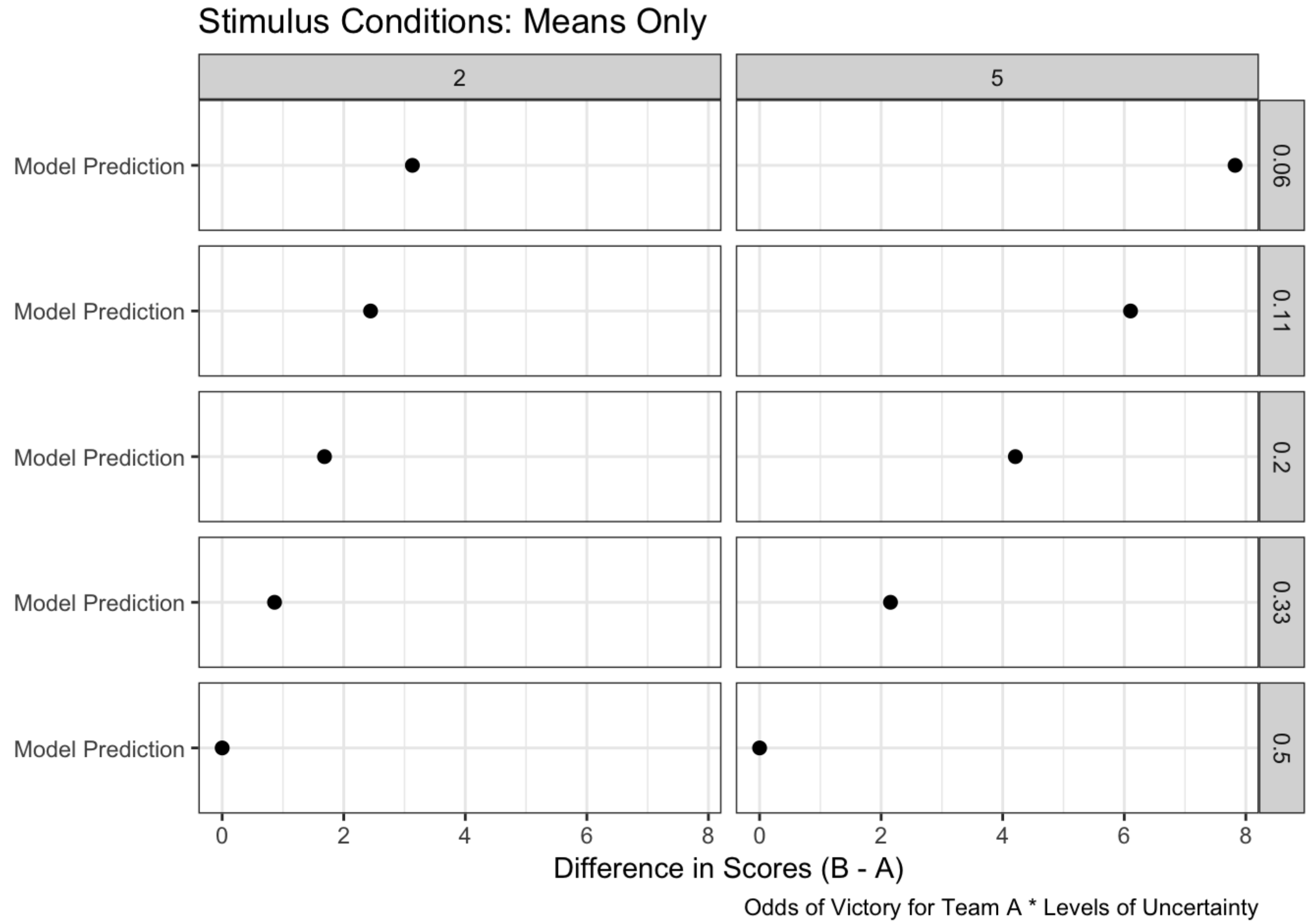
```
##              condition sd odds_of_victory      mean
## 1    low var, 1:1 odds  2      0.50000000 0.0000000
## 2    low var, 2:1 odds  2      0.33333333 0.8614546
## 3    low var, 4:1 odds  2      0.20000000 1.6832425
## 4    low var, 8:1 odds  2      0.11111111 2.4412807
## 5   low var, 16:1 odds  2      0.05882353 3.1294529
## 6   high var, 1:1 odds  5      0.50000000 0.0000000
## 7   high var, 2:1 odds  5      0.33333333 2.1536365
## 8   high var, 4:1 odds  5      0.20000000 4.2081062
## 9   high var, 8:1 odds  5      0.11111111 6.1032017
## 10 high var, 16:1 odds  5      0.05882353 7.8236324
```

These data are visualized in the stimuli below.

Visualizations and Heuristics

We will test users' ability to judge CLES and use that information to inform their decision about the bet amount based on the information encoded in one of a set of visualizations. Each visualization we test will be associated with a specific hypothesis about the heuristic that participants are likely to use in order to read CLES from the visualization. I am thinking of these heuristics as a form of *representativeness*, whereby the visual form of the visualization is treated as an explicit representation of the reliability of the effect.

Means Only

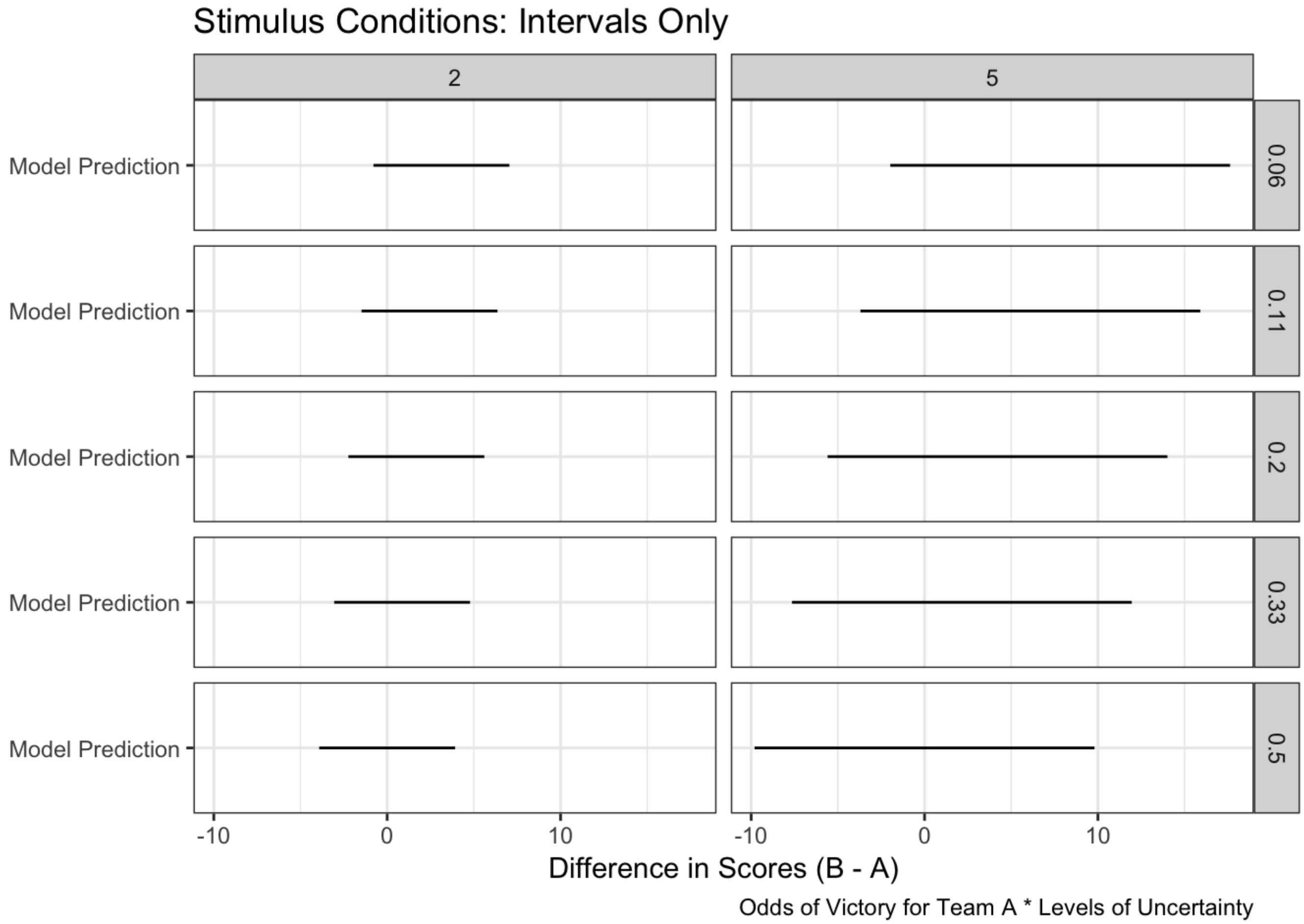


When relying on means alone to make judgments about CLES, users have no uncertainty information. As Jessica proposed in the analysis of her 2015 data she shared with me, we should model the perceived reliability of the effect from means alone as a function of the position of the mean. Specifically:

$$PerceivedPr(A > B) \propto 50 - 50 * \frac{\mu_B - \mu_A}{\max(\mid \mu_B - \mu_A \mid)}$$

This predicts that a user will underestimate the probability that team A wins as the mean difference between scores for teams A and B is more negative and overestimate Pr(A > B) as the difference $(B - A)$ becomes more positive. This should result poor betting at extreme odds.

Intervals Only

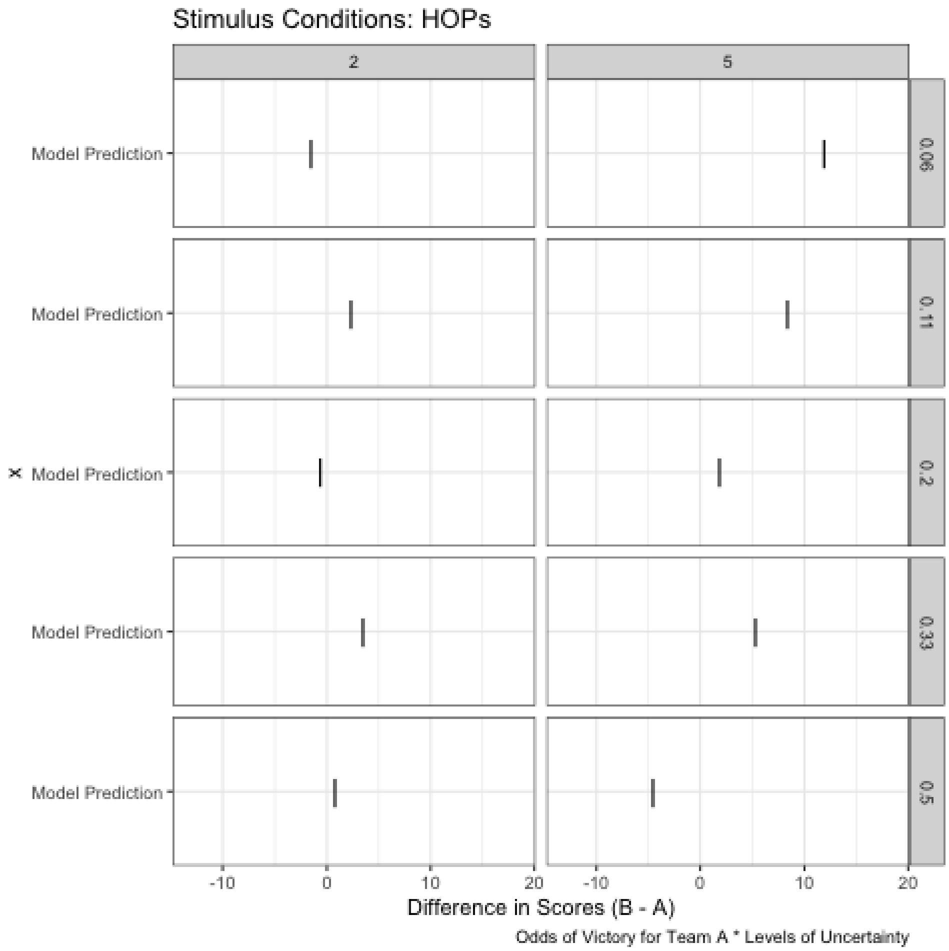


When relying on intervals to make judgments about CLES, I predict that users will use the proportion of the interval which crosses zero as a cue. This needs to be a piecewise function because the interpretation of the interval overlap depends on which group mean is larger (i.e., the sign of the difference). Where the mean score for team A is larger than the mean score for team B (i.e., negative difference), interval overlap is a cue to the degree to which $A > B$ is uncertain. However, where the mean score for team A is smaller than the mean score for team B (i.e., positive difference), interval overlap is a cue to the degree to which $A > B$ is possible.

$$PerceivedPr(A > B) \propto \begin{cases} 100 - 50 * \frac{Interval>0}{IntervalLength} & A \geq B \\ \\ 50 * \frac{Interval<0}{IntervalLength} & A < B \end{cases}$$

HOPs

```
# for HOPs we need to add draws to our dataframe
n <- 100 # number of samples
conds_df$sample_n <- n
conds_df_draws <- conds_df %>% as_tibble() %>%
  mutate(draw=pmap(list(sample_n, mean, sd), rnorm), # get a list of draws from the dist
ribution for each condition
         draw_n=list(seq(1, n))) %>% #number each sample in order to animate multiple vi
ews simultaneously
  unnest() # get back to a tidy format
```



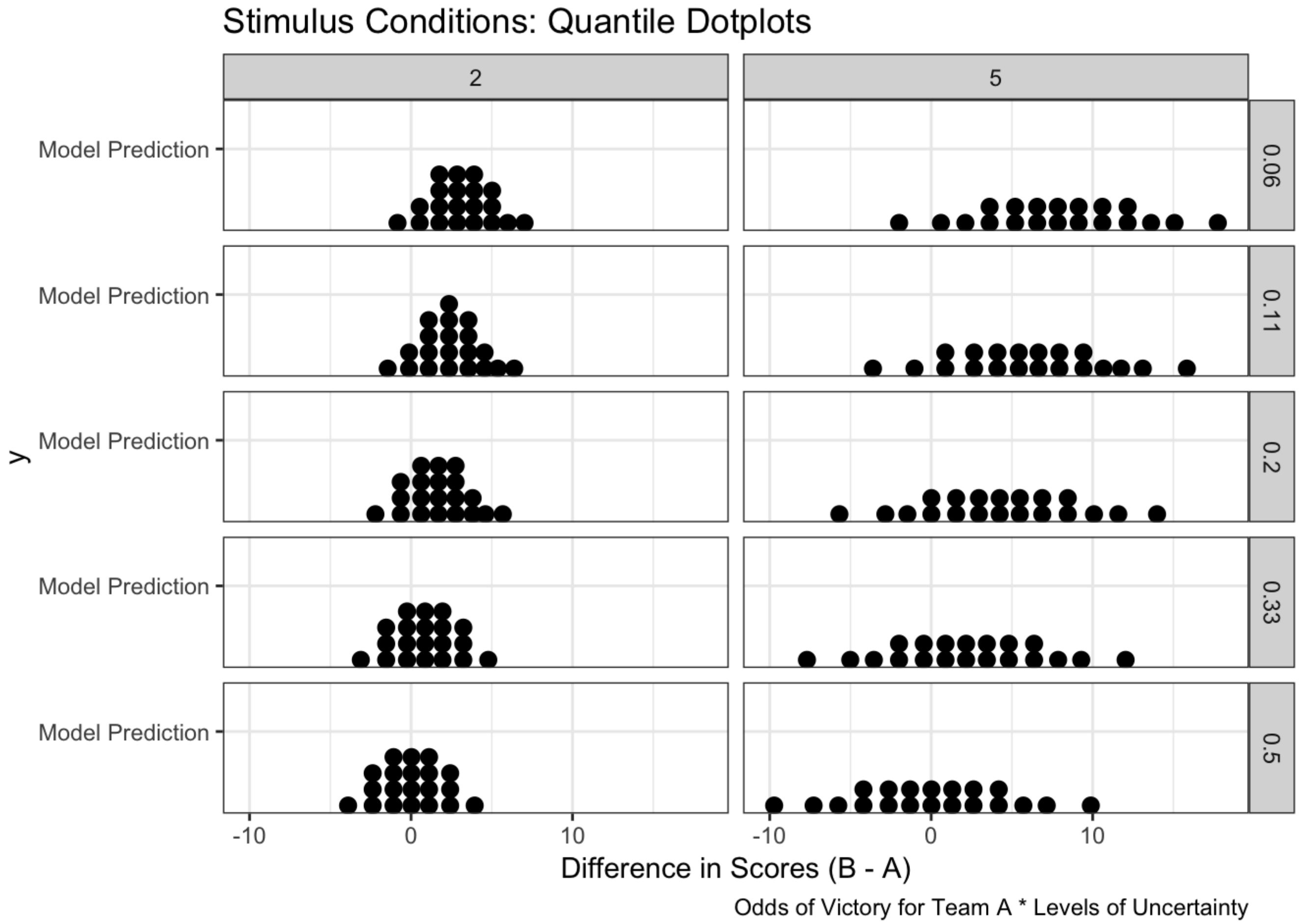
With HOPs, I would expect users to judge CLES based on the proportion of draws crossing the zero line.

$$PerceivedPr(A > B) \propto 100 * \frac{\Sigma(draws > 0)}{\Sigma(draws)}$$

Given representative sampling this should be pretty close to the ground truth. Although maybe we can test order effects—and thus differentiate the outcome proportion heuristic from the ground truth—by manipulating the series of draws and modeling primacy and recency as different weighting functions over the timeseries of draws.

Quantile Dotplots

```
# as with hops, we need to add to the dataframe to build quantile dotplots
n_dots <- 20 # number of dots
n <- 10000 # number of samples
conds_df$sample_n <- n
conds_df_quantiles <- conds_df %>% as_tibble() %>%
  mutate(draws=pmap(list(sample_n, mean, sd), rnorm), # sample each distribution (as bef
ore)
    quantiles=map(draws, ~ quantile(unlist(.x), ppoints(n_dots))), # use these draws to
 get quantiles
    draws=NULL) %>% # drop draws from the dataframe since these were an intermediate ste
p anyway
  unnest()
```



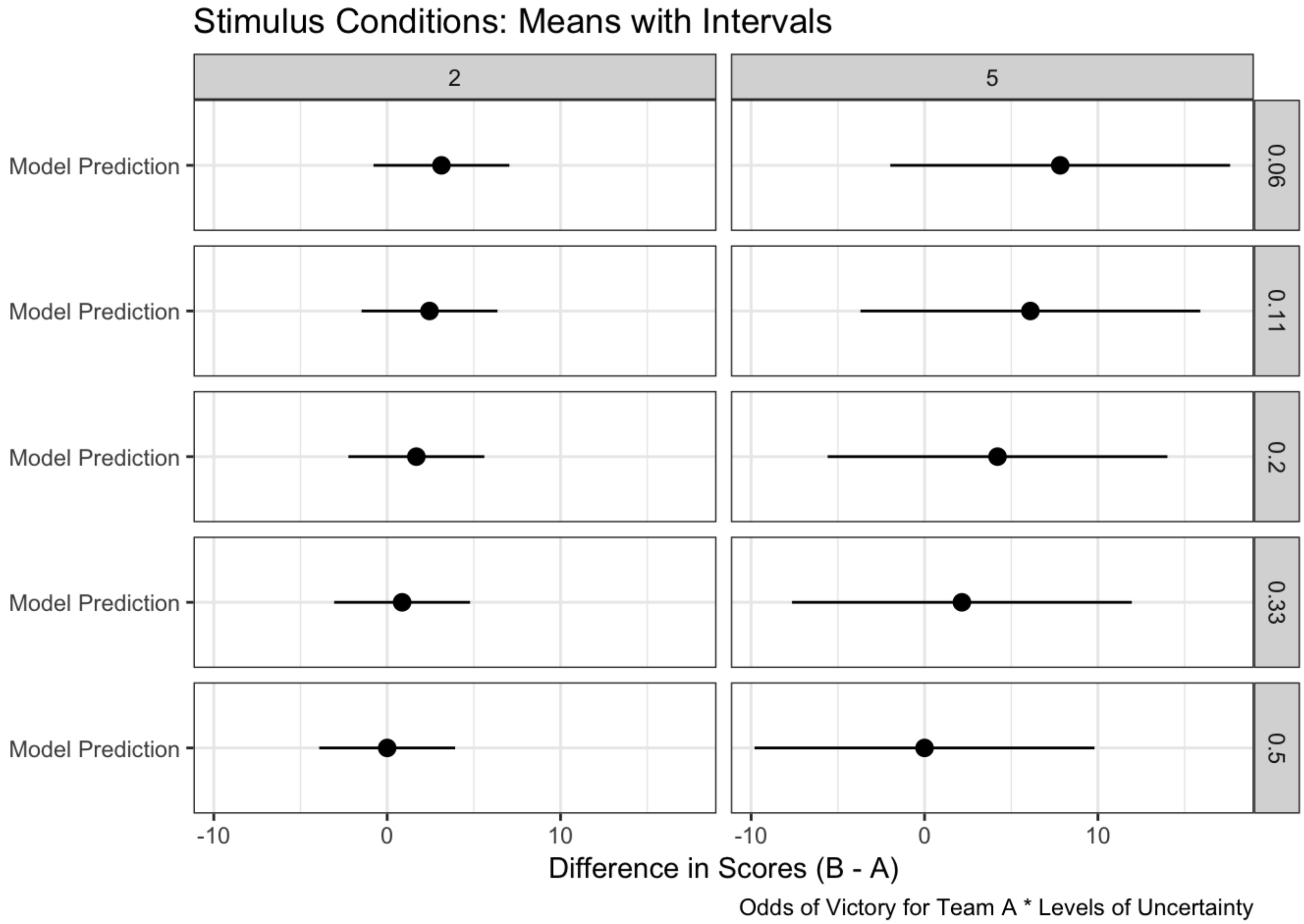
Similar to HOPs, I would expect users of quantile dotplots to judge CLES based on the proportion of dots crossing the zero line. However, just like the interval proportion heuristic, this needs to be a piecewise function to account for the sign of the difference.

$$PerceivedPr(A > B) \propto \begin{cases} 100 - 50 * \frac{\Sigma(Dots>0)}{\Sigma(Dots)} & A \geq B \\ \\ 50 * \frac{\Sigma(Dots<0)}{\Sigma(Dots)} & A < B \end{cases}$$

Again, this should be pretty close to the ground truth. I'm not sure if we will have enough statistical power to test differences between this heuristic and the ground truth which might arise from binning. For instance, using dotplot-20, I would expect CLES judgments to tend toward multiples of 5%, but given other sources of imprecision in responses, it might be difficult to reliably differentiate this rounding from the ground truth.

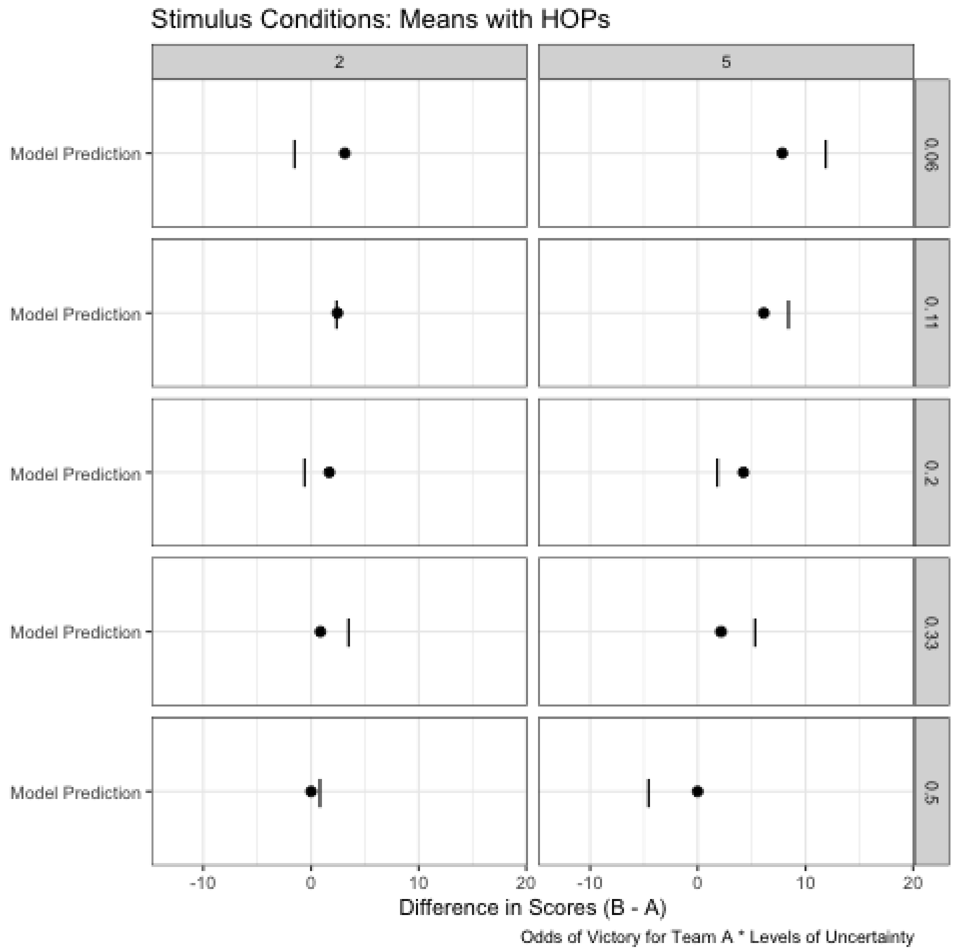
I do think it would be nice to run an experiment with both HOPs and quantile dotplots. Which gets closer to the ground truth? How reliably do users' judgments reflect the ground truth? I think these would be novel comparisons.

Means with Intervals



As we’ve discussed, visualizations which mix the representations above (e.g., means with intervals) might involve the use of different heuristics among different users. I predict that the CLES judgments of some portion of users in this condition will most closely align with the mean magnitude heuristic and others will most closely align with the interval proportion heuristic (see *Modeling Heuristics* below). To the extent that users seem to rely on the mean magnitude heuristic in the presence of uncertainty information, we can make the case that people ignore uncertainty when given a mean to latch onto. We might see this same heuristic reflected to a lesser degree with quantile dotplots or means with HOPs.

Means with HOPs



My notes above about means with intervals sum up my thoughts about this condition as well. I predict that we will see some mixture of heuristics in this condition across users.

Heuristics and Underlying Hypothesis

In a sense, each of the heuristics I propose above are part of a broader hypothesis that judgments based on visualization more often rely on visual-spatial cues than reasoning about the underlying distribution implied by a visualization (e.g., confidence intervals). To the extent that this sort of *what-you-see-is-all-there-is* bias is reflected in our data, I think it really strengthens the empirical case for expressive uncertainty visualizations that can be processed automatically (i.e., Type 1 processing).

Comparing Two Distributions

Jessica suggested that comparing two distributions is an ideal task for two reasons: 1) it is harder than reading the difference distribution which should lead a greater number of users to adopt heuristics; and 2) comparing distributions is the most common way to judge uncertainty in scientific effects.

Data for Stimuli

Again, we will vary the mean and standard deviation of the distribution users are judging so that we measure different levels of uncertainty (e.g., low and high) and different odds of victory for team A (e.g., 1:1, 2:1, 4:1, 8:1, 16:1).

```
# set up conditions dataframe
condition <- rep(c("low var, 1:1 odds", "low var, 2:1 odds", "low var, 4:1 odds", "low v
ar, 8:1 odds", "low var, 16:1 odds",
               "high var, 1:1 odds", "high var, 2:1 odds", "high var, 4:1 odds", "high v
ar, 8:1 odds", "high var, 16:1 odds"), 2)
std_diff <- rep(sort(rep(c(2, 5), 5)), 2) # different levels of uncertainty about the ma
rgin of victory
odds <- rep(c(1/2, 1/3, 1/5, 1/9, 1/17), 4) # probability of team A winning
teamAB <- sort(rep(c("A", "B"), 10))
conds_df2 <- data.frame(
    "condition"=condition,
    "sd_diff"=std_diff,
    "odds_of_victory"=odds,
    "team"=teamAB
)

# add column for the mean difference
conds_df2$mean_diff <- - (conds_df2$sd_diff * qnorm(conds_df2$odds_of_victory))
# conds_df2$mean_diff <- conds_df2$sd_diff * qnorm(conds_df2$odds_of_victory)

# compute the mean of distributions A and B
center <- 50 # set the center point between the score of A and B
conds_df2$mean[conds_df2$team == "A"] <- center - conds_df2$mean_diff[conds_df2$team ==
"A"] / 2
conds_df2$mean[conds_df2$team == "B"] <- center + conds_df2$mean_diff[conds_df2$team ==
"B"] / 2

# compute the sd of distributions A and B, assuming independent and equal variances
conds_df2$sd <- sqrt(conds_df2$sd_diff ^ 2 / 2)

# print
conds_df2
```

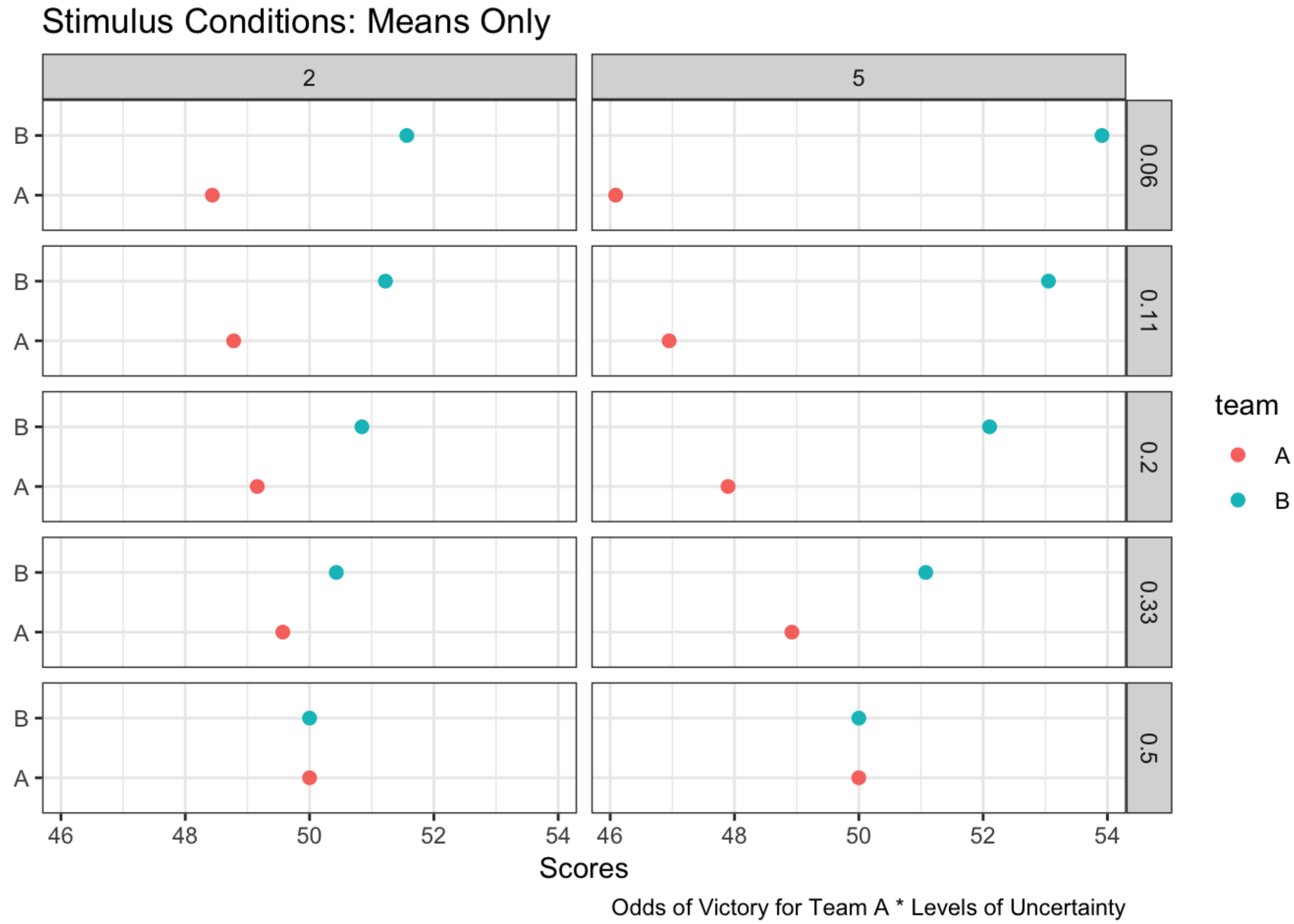
```
##              condition sd_diff odds_of_victory team mean_diff     mean
## 1     low var, 1:1 odds       2      0.50000000    A 0.0000000 50.00000
## 2     low var, 2:1 odds       2      0.33333333    A 0.8614546 49.56927
## 3     low var, 4:1 odds       2      0.20000000    A 1.6832425 49.15838
## 4     low var, 8:1 odds       2      0.11111111    A 2.4412807 48.77936
## 5    low var, 16:1 odds       2      0.05882353    A 3.1294529 48.43527
## 6    high var, 1:1 odds       5      0.50000000    A 0.0000000 50.00000
## 7    high var, 2:1 odds       5      0.33333333    A 2.1536365 48.92318
## 8    high var, 4:1 odds       5      0.20000000    A 4.2081062 47.89595
## 9    high var, 8:1 odds       5      0.11111111    A 6.1032017 46.94840
## 10  high var, 16:1 odds       5      0.05882353    A 7.8236324 46.08818
## 11    low var, 1:1 odds       2      0.50000000    B 0.0000000 50.00000
## 12    low var, 2:1 odds       2      0.33333333    B 0.8614546 50.43073
## 13    low var, 4:1 odds       2      0.20000000    B 1.6832425 50.84162
## 14    low var, 8:1 odds       2      0.11111111    B 2.4412807 51.22064
## 15   low var, 16:1 odds       2      0.05882353    B 3.1294529 51.56473
## 16   high var, 1:1 odds       5      0.50000000    B 0.0000000 50.00000
## 17   high var, 2:1 odds       5      0.33333333    B 2.1536365 51.07682
## 18   high var, 4:1 odds       5      0.20000000    B 4.2081062 52.10405
## 19   high var, 8:1 odds       5      0.11111111    B 6.1032017 53.05160
## 20  high var, 16:1 odds       5      0.05882353    B 7.8236324 53.91182
##          sd
## 1  1.414214
## 2  1.414214
## 3  1.414214
## 4  1.414214
## 5  1.414214
## 6  3.535534
## 7  3.535534
## 8  3.535534
## 9  3.535534
## 10 3.535534
## 11 1.414214
## 12 1.414214
## 13 1.414214
## 14 1.414214
## 15 1.414214
## 16 3.535534
## 17 3.535534
## 18 3.535534
## 19 3.535534
## 20 3.535534
```

These data are visualized in the stimuli below.

Visualizations and Heuristics

We will test users' ability to judge CLES and use that information to inform their decision about the bet amount based on the information encoded in one of a set of visualizations. Each visualization we test will be associated with a specific hypothesis about the heuristic that participants are likely to use in order to read CLES from the visualization. I am thinking of these heuristics as a form of *representativeness*, whereby the visual form of the visualization is treated as an explicit representation of the reliability of the effect.

Means Only



When relying on means alone to make judgments about CLES, users have no uncertainty information. As Jessica proposed in the analysis of her 2015 data she shared with me, we should model the perceived reliability of the effect from means alone as a function of the position of the mean. Specifically:

$$PerceivedPr(A > B) \propto 50 - 50 * \frac{\mu_B - \mu_A}{\max(|\ \mu_B - \mu_A\ |)}$$

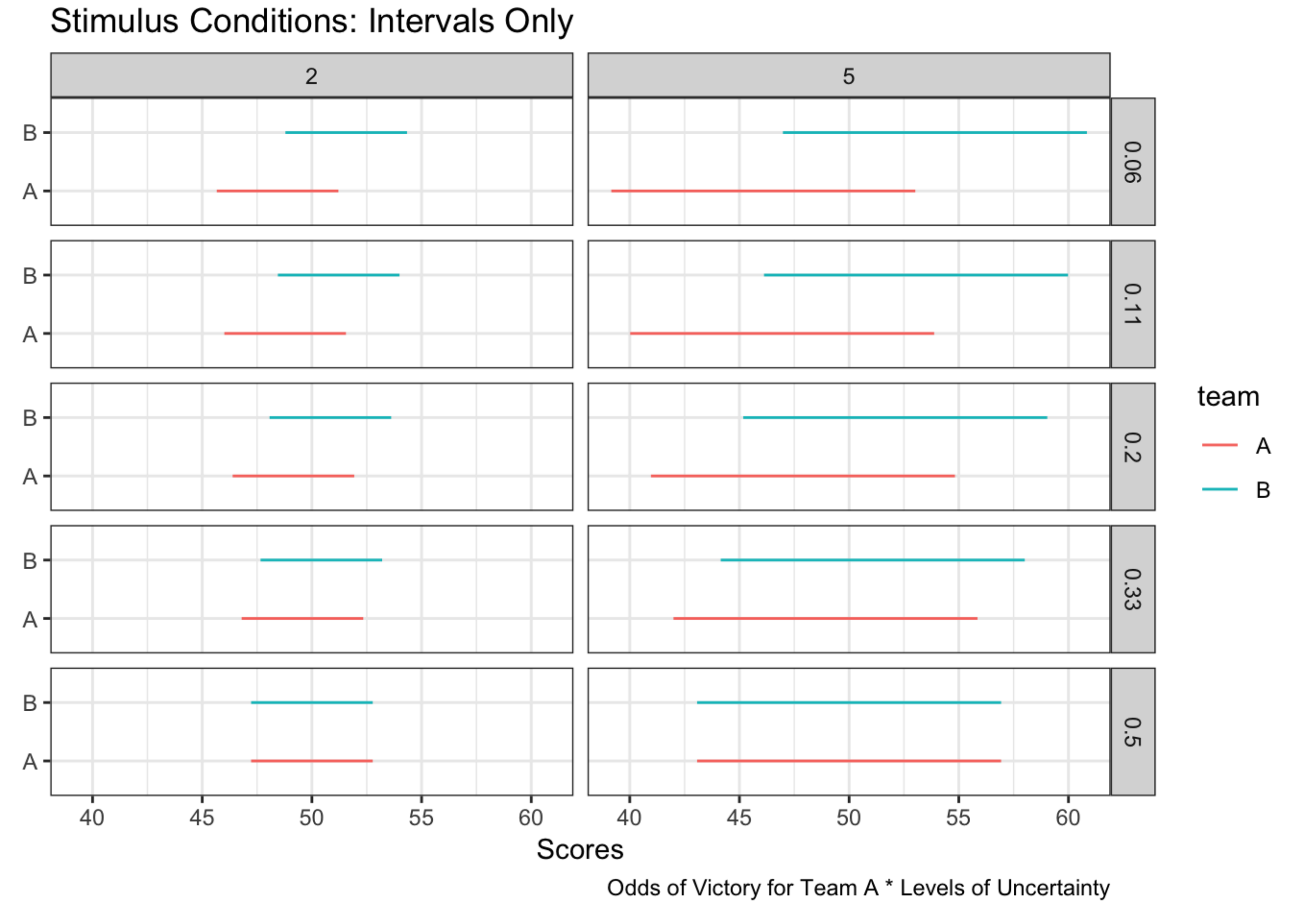
This predicts that a user will underestimate the probability that team A wins as the mean difference between scores $(B - A)$ becomes more negative and overestimate as the difference becomes more positive. This should result in poor betting for extreme odds.

Alternatively, the heuristic might be relative to the length of the axis such that:

$$PerceivedPr(A > B) \propto 50 - 50 * \frac{\mu_B - \mu_A}{AxisRange}$$

This would yield even more dramatic underestimation of $Pr(A > B)$ at large mean differences.

Intervals Only



When relying on intervals to make judgments about CLES, Jessica proposed multiple possible heuristics based on interval overlap:

$$PerceivedPr(A > B) \propto \begin{cases} 100 - 50 * \frac{IntervalOverlap}{\mu(IntervalLength)} & A \geq B \\ 50 * \frac{IntervalOverlap}{\mu(IntervalLength)} & A < B \end{cases}$$

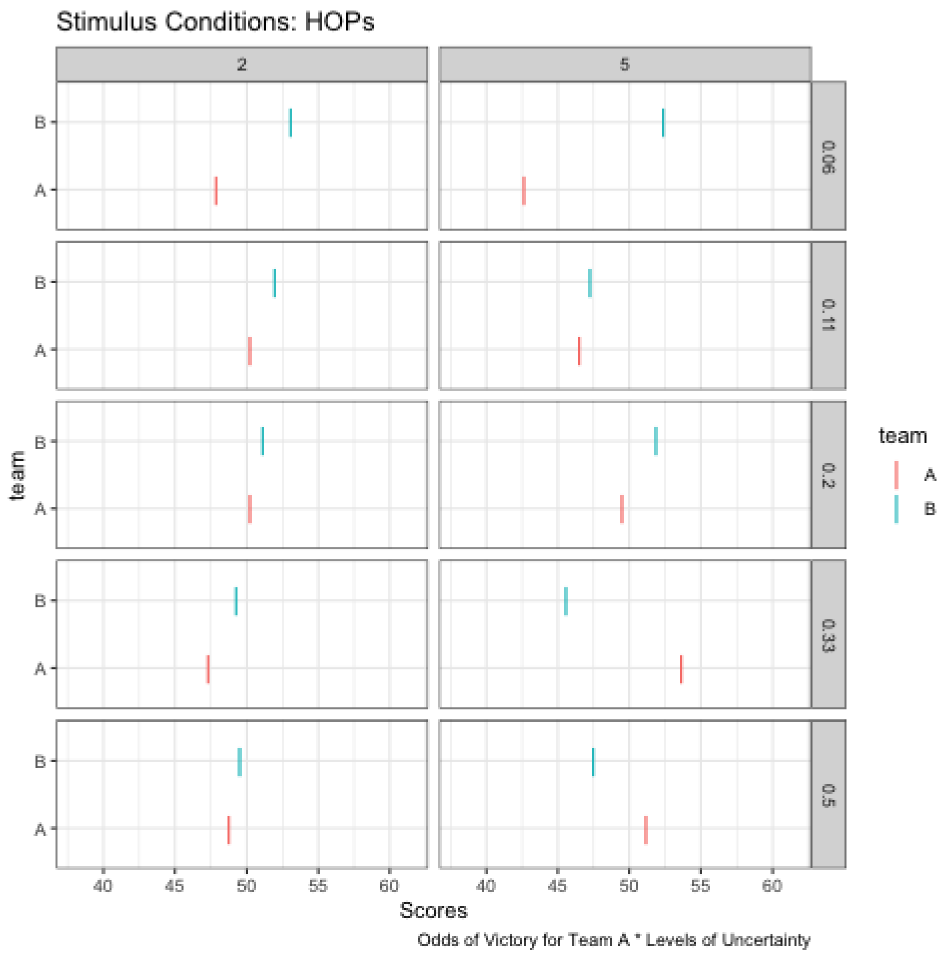
or

$$PerceivedPr(A > B) \propto \begin{cases} 100 - 50 * \frac{IntervalOverlap}{AxisRange} & A \geq B \\ 50 * \frac{IntervalOverlap}{AxisRange} & A < B \end{cases}$$

These heuristics predict that, when there is a small amount of overlap, users will overestimate $Pr(A > B)$ when the mean of B is larger than A and underestimate $Pr(A > B)$ when the mean of A is larger than B. These functions are piecewise because the interpretation of the interval overlap depends on which group mean is larger. Where the mean score for team A is larger than the mean score for team B, interval overlap is a cue to the degree to which $A > B$ is uncertain. However, where the mean score for team A is smaller than the mean score for team B, interval overlap is a cue to the degree to which $A > B$ is possible.

HOPs

```
# for HOPs we need to add draws to our dataframe
n <- 100 # number of samples
conds_df2$sample_n <- n
conds_df2_draws <- conds_df2 %>% as_tibble() %>%
  mutate(draw=pmap(list(sample_n, mean, sd), rnorm), # get a list of draws from the dist
ribution for each condition
         draw_n=list(seq(1, n))) %>% #number each sample in order to animate multiple vi
ews simultaneously
  unnest() # get back to a tidy format
```



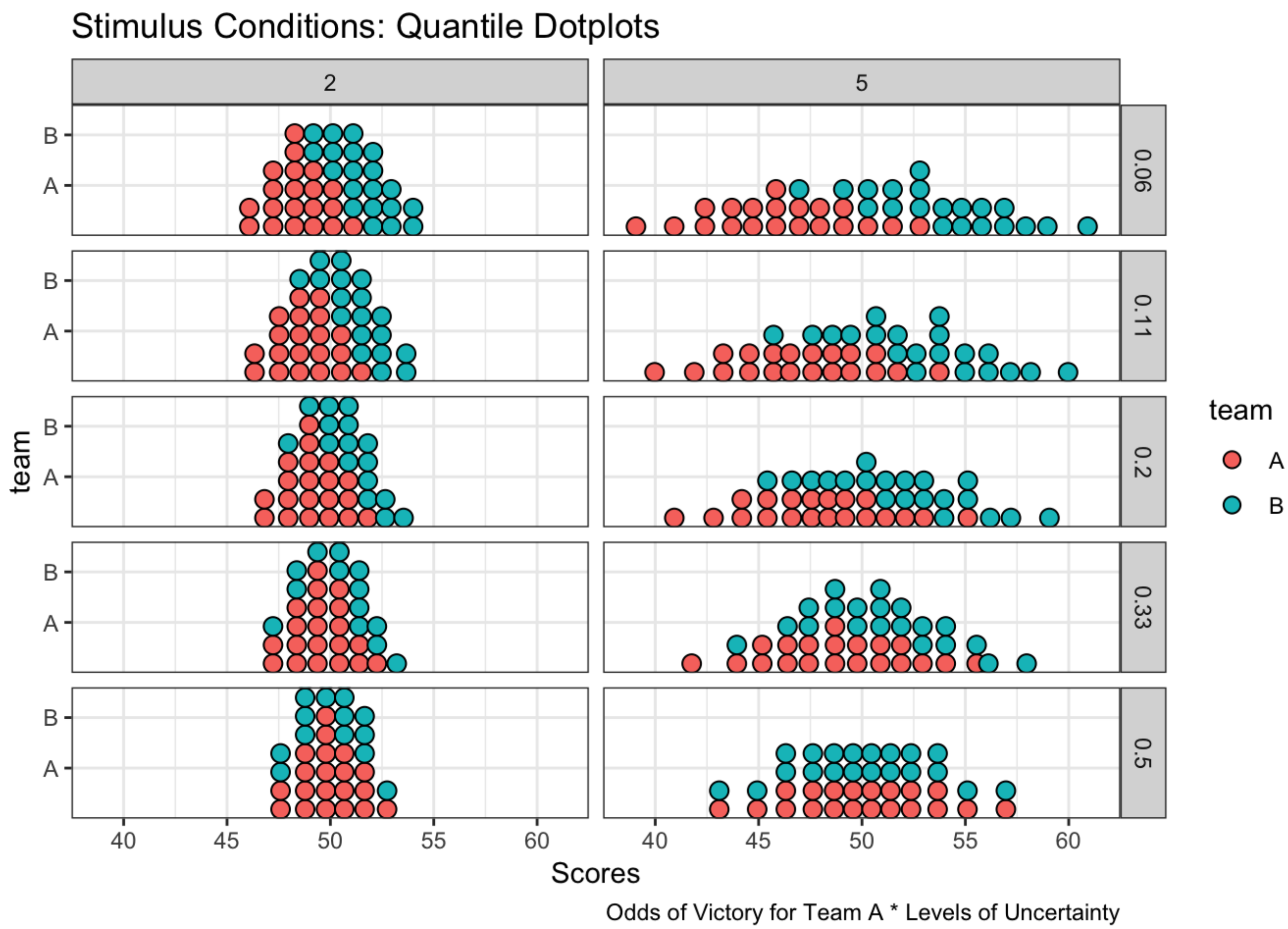
With HOPs, I would expect users to judge CLES based on the proportion of draws crossing each other.

$$PerceivedPr(A > B) \propto 100 * \frac{\Sigma(draws_{A>B})}{\Sigma(draws)}$$

Given representative sampling this should be pretty close to the ground truth. Although maybe we can test order effects—and thus differentiate the outcome proportion heuristic from the ground truth—by manipulating the series of draws and modeling primacy and recency as different weighting functions over the timeseries of draws.

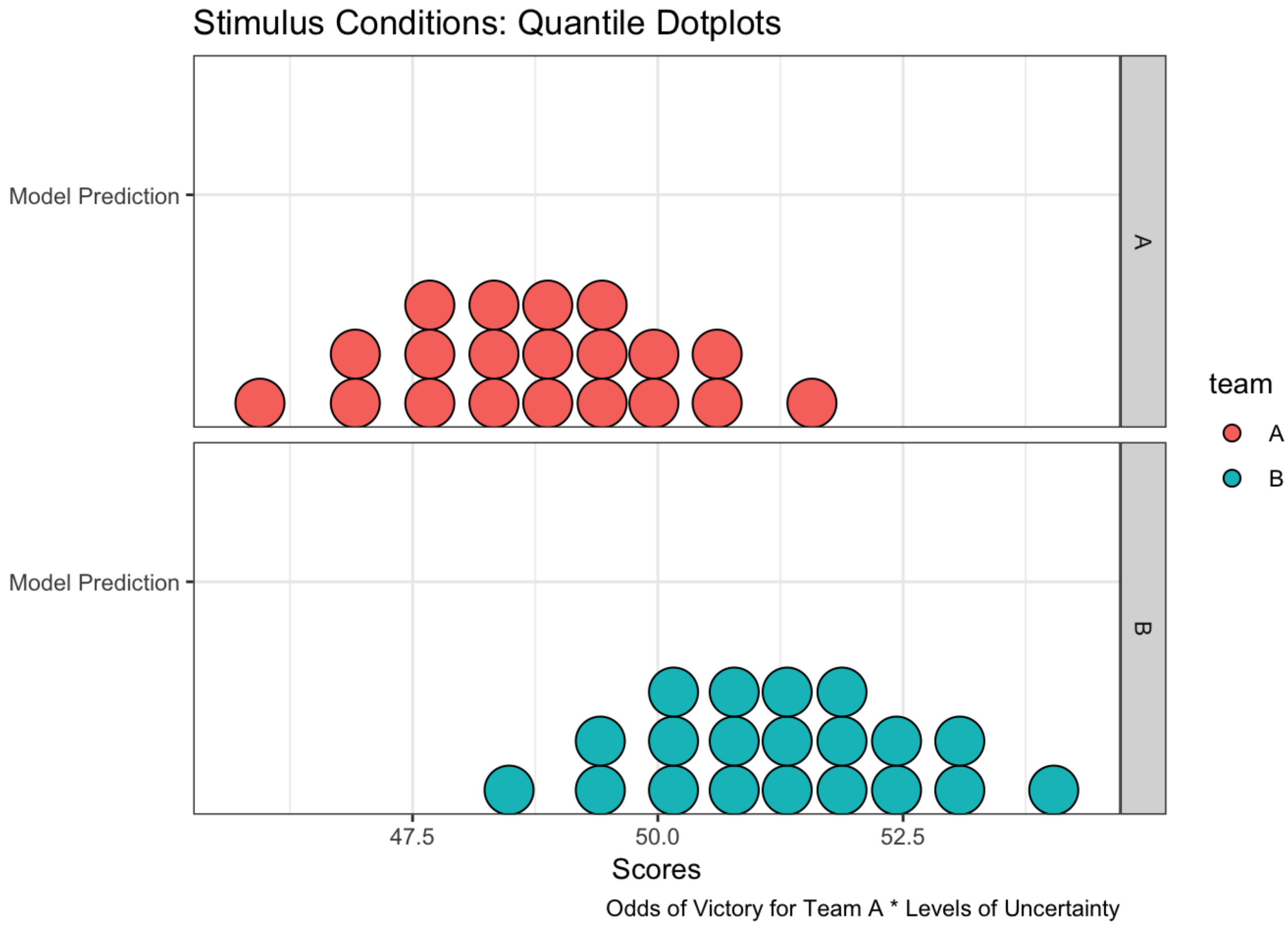
Quantile Dotplots

```
# as with hops, we need to add to the dataframe to build quantile dotplots
n_dots <- 20 # number of dots
n <- 10000 # number of samples
conds_df2$sample_n <- n
conds_df2_quantiles <- conds_df2 %>% as_tibble() %>%
  mutate(draws=pmap(list(sample_n, mean, sd), rnorm), # sample each distribution (as bef
ore)
    quantiles=map(draws, ~ quantile(unlist(.x), ppoints(n_dots))), # use these draws to
 get quantiles
    draws=NULL) %>% # drop draws from the dataframe since these were an intermediate ste
p anyway
  unnest()
```



I'm trying to figure out how to plot these correctly. Usually, Matt uses facets to separate multiple distributions.

For example:

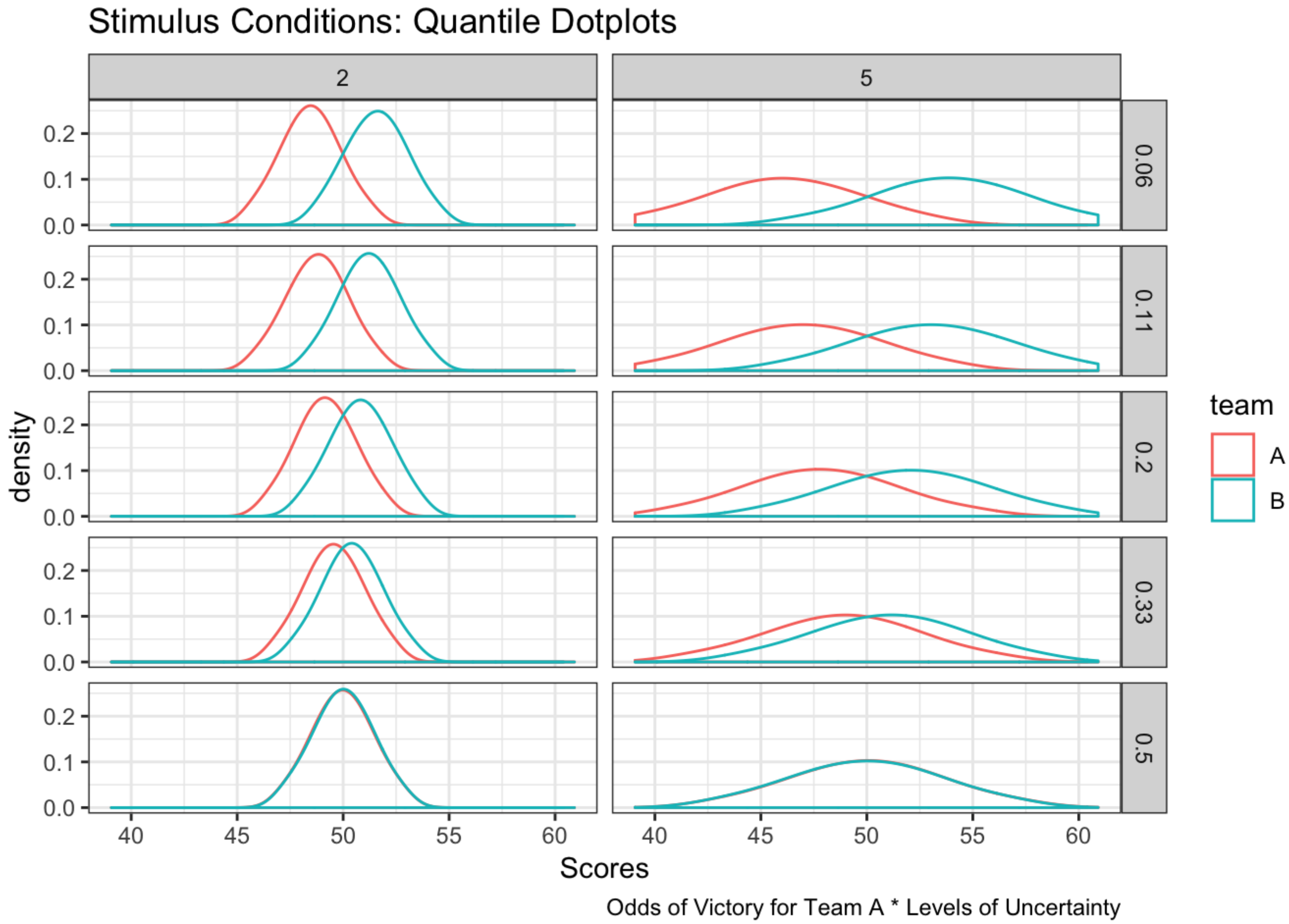


I would expect the heuristic in this case to be based on the number of dots that overlap, very similar to the interval overlap heuristic described above.

$$PerceivedPr(A > B) \propto \begin{cases} 100 - 50 * \frac{\Sigma(DotsOverlap)}{\Sigma(Dots)} & A \geq B \\ \\ 50 * \frac{\Sigma(DotsOverlap)}{\Sigma(Dots)} & A < B \end{cases}$$

I think the predictions in this case should be similar to the predictions for the interval overlap heuristic.

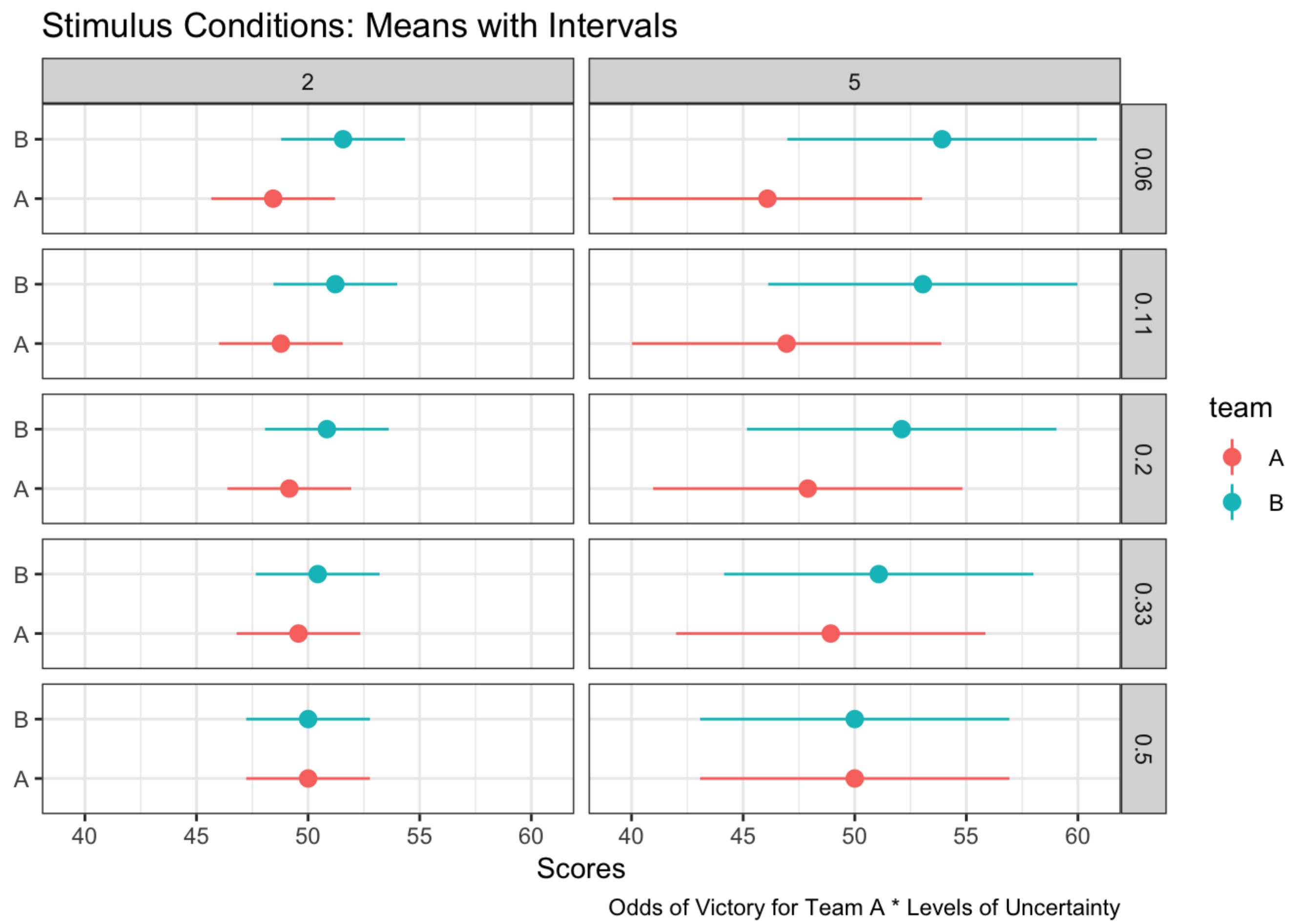
Densities



For densities, I expect the heuristic will be based on the proportion of overlap in area (on a scale from 0 to 1), similar to the interval overlap heuristic.

$$PerceivedPr(A > B) \propto \begin{cases} 100 - 50 * AreaOverlap & A \geq B \\ 50 * AreaOverlap & A < B \end{cases}$$

Means with Intervals



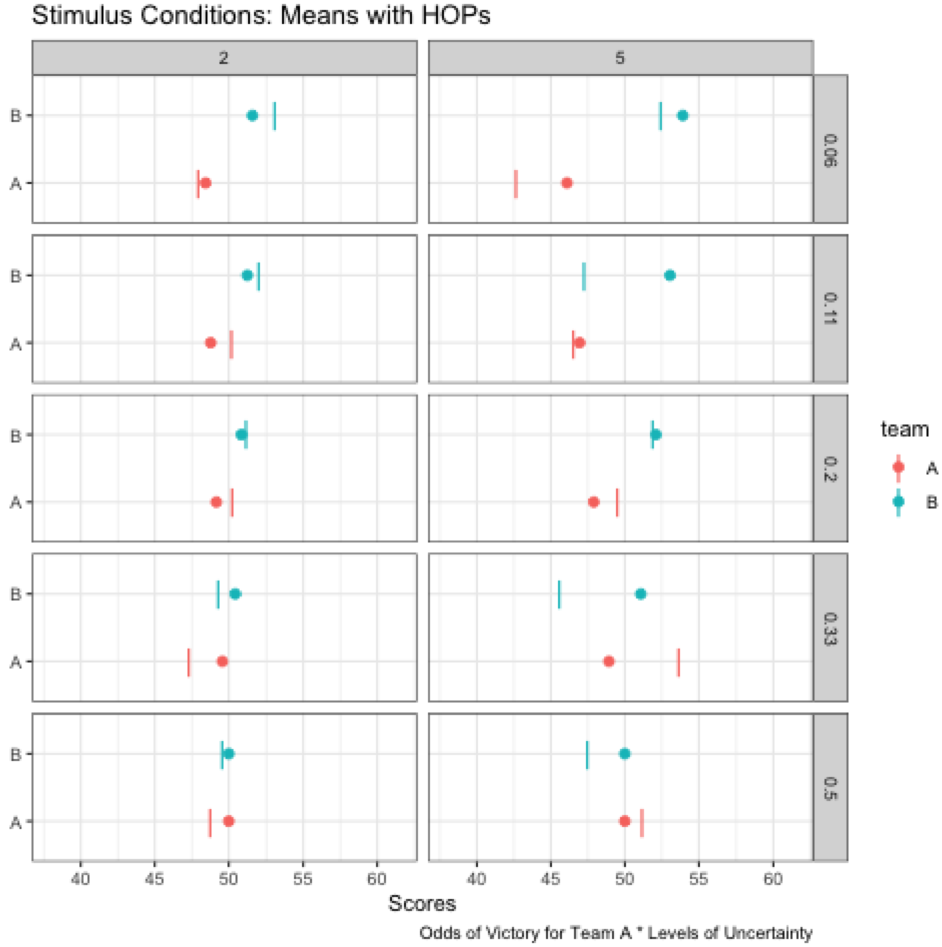
As we've discussed, visualizations which mix the representations above (e.g., means with intervals) might involve the use of different heuristics among different users. I predict that the CLES judgments of some portion of users in this condition will most closely align with the mean magnitude heuristic and others will most closely align with the interval overlap heuristic (see *Modeling Heuristics* below).

Alternatively, as Jessica pointed out, there may be a heuristic whereby users combine means and intervals such as:

$$PerceivedPr(A > B) \propto 50 - 50 * \frac{\mu_B - \mu_A}{\mu(IntervalLength)/2}$$

To the extent that users seem to rely on the mean magnitude heuristic in the presence of uncertainty information, we can make the case that people ignore uncertainty when given a mean to latch onto. We might see this same heuristic reflected to a lesser degree with quantile dotplots or means with HOPs.

Means with HOPs



My notes above about means with intervals sum up my thoughts about this condition as well. I predict that we will see some mixture of heuristics in this condition across users.

Modeling Heuristics

Each heursitic will be associated with its own submodel which makes predictions about performance in the perception of CLES. In addition to heuristics, we will also model an optimal strategy which is the ground truth response that the user should give if they have full access to information and perfect judgment. In a hierarchical Bayesian model, we will estimate the probability that each user employs each of a set of candidate strategies (i.e., heuristics, ground truth) depending on predictors like numeracy, and we will estimate hyperparameters for probability of each strategy in the population of users as a function of visualization condition.

The amount that a user bets in the decision task is a measure of utility. The decision task is designed so that each level of odds of victory has a distinct bet amount that will maximize the expected payout. Thus, the difference between a user's bet and the optimal bet represents the error in their judgment of utility, which is a measure of their ability to integrate uncertainty information from the visualization (i.e., perceived CLES) with the incentive structure of the task.

We will model the bias and noise in the difference between a user's bet and the optimal bet. The bias parameter will capture patterns in responses such as risk aversion which reflect the user's sense of the value of the payoff, rather than the incentive structure of the task. The noise parameter measures the distortion of decoded uncertainty information between the perceptual judgment of CLES and the decision about bet amount. We should expect to see some baseline noise due to difficulty balancing the incentive structure of the task, but I expect that some portion of this noise will be predicted by visualization condition. We can think of visualization-attributed noise as a proxy for the amount of working memory employed in reading the visualization such that if more effort is spent reading the vis, less attentional capacity remains to carefully consider the incentive structure of the task. This will allow us to draw conclusions about the degree of Type 2 processing involved in decoding each visualization, hopefully leading to an interesting discussion about when heuristics can be helpful (e.g., alleviating cognitive load for decisions) as well as when they can be harmful by leading to biased perceptions.