

# Pilot Analysis: Exploratory Visualization

This document is intended to give an overview of the response distributions from our pilot.

## Data

### Load Worker Responses from Pilot

The data is already anonymous and in a tidy format at this stage in the analysis pipeline. We just need to read it in and do some preprocessing

```
# read in data
responses_df <- read_csv("pilot-anonymous.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   workerId = col_character(),
##   batch = col_integer(),
##   condition = col_character(),
##   start_gain_frame = col_character(),
##   numeracy = col_integer(),
##   gender = col_character(),
##   age = col_character(),
##   education = col_character(),
##   chart_use = col_character(),
##   intervene = col_integer(),
##   outcome = col_character(),
##   pSup = col_integer(),
##   trial = col_character(),
##   trialIdx = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```

# preprocessing
responses_df <- responses_df %>%
  rename( # rename to convert away from camel case
    worker_id = workerId,
    company_value = companyValue,
    ground_truth = groundTruth,
    p_contract_new = pContractNew,
    p_contract_old = pContractOld,
    p_superiority = pSup,
    start_time = startTime,
    resp_time = respTime,
    trial_dur = trialDur,
    trial_idx = trialIdx
  ) %>%
  mutate( # mutate to jitter probability of superiority away from boundaries
    p_superiority = ifelse(p_superiority == 0, 0.25, p_superiority),      # avoid
    # responses equal to zero
    p_superiority = ifelse(p_superiority == 100, 99.75, p_superiority)    # avoid
    # responses equal to one-hundred
  )

head(responses_df)

```

```

## # A tibble: 6 x 27
##   worker_id batch condition baseline contract_value exchange
##   <chr>      <int> <chr>      <dbl>          <dbl>      <dbl>
## 1 be209114     0 interval...    0.5            2.25    0.0480
## 2 be209114     0 interval...    0.5            2.25    0.0480
## 3 be209114     0 interval...    0.5            2.25    0.0480
## 4 be209114     0 interval...    0.5            2.25    0.0480
## 5 be209114     0 interval...    0.5            2.25    0.0480
## 6 be209114     0 interval...    0.5            2.25    0.0480
## # ... with 21 more variables: start_gain_frame <chr>, total_bonus <dbl>,
## #   duration <dbl>, numeracy <int>, gender <chr>, age <chr>,
## #   education <chr>, chart_use <chr>, company_value <dbl>,
## #   ground_truth <dbl>, intervene <int>, outcome <chr>,
## #   p_contract_new <dbl>, p_contract_old <dbl>, p_superiority <int>,
## #   payoff <dbl>, resp_time <dbl>, start_time <dbl>, trial <chr>,
## #   trial_dur <dbl>, trial_idx <chr>

```

## Response Distributions

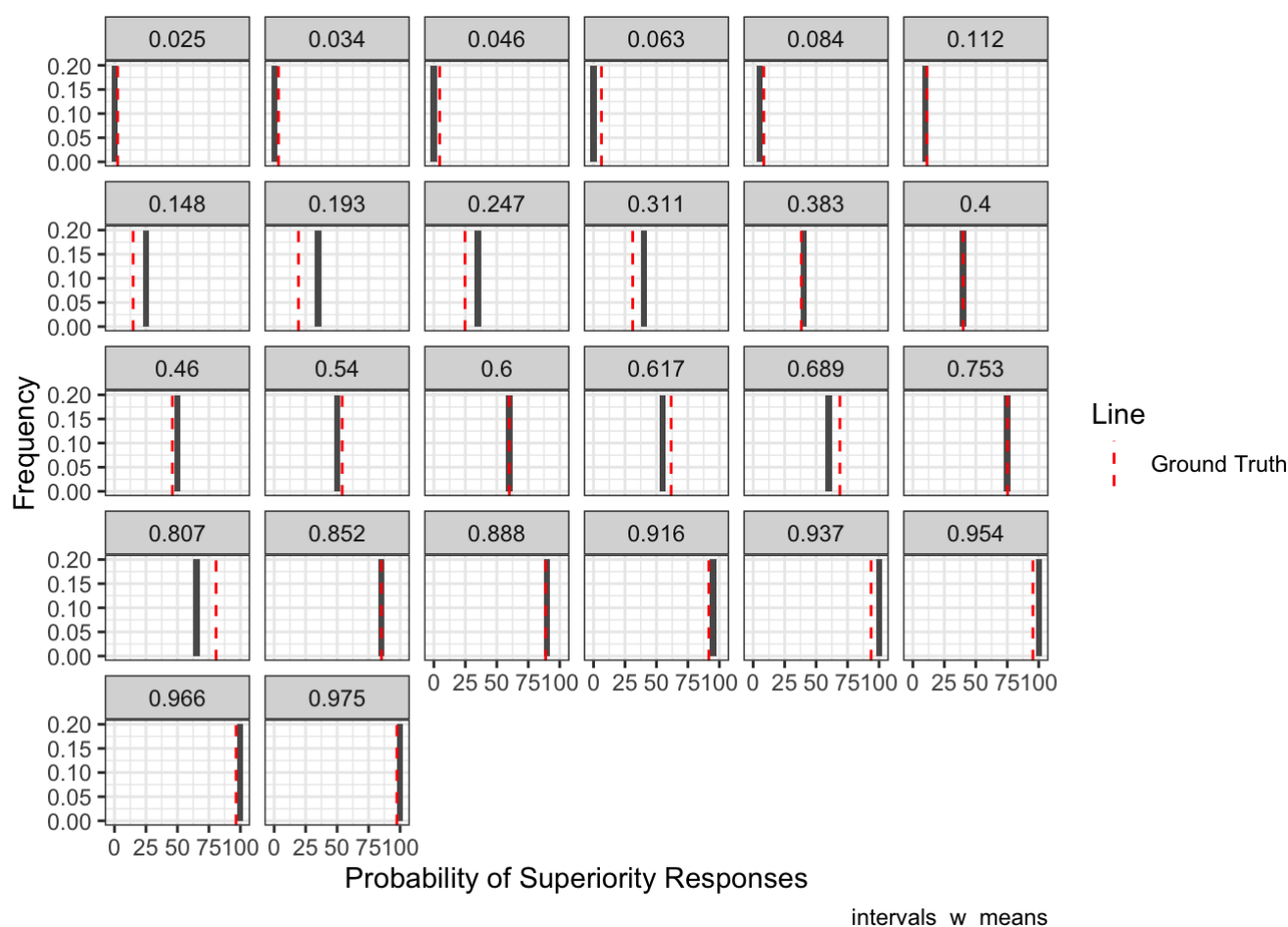
### Probability of Superiority Judgments

Let's plot histograms of probability of superiority judgments at each level of the ground truth probability of superiority. We show the ground truth in red. This will give us an overview of bias and precision in judgments. We do this separately for each visualization condition to limit the number of faceted subplots in a single view.

```

for (cond in unique(responses_df$condition)) {
  plt <- responses_df %>% filter(condition == cond) %>%
    ggplot(aes(x = p_superiority)) +
    geom_histogram(aes(y = ..density..), binwidth = 5) +
    geom_vline(aes(xintercept = ground_truth * 100, linetype = "Ground Truth"), color
= "red") +
    scale_linetype_manual(name = "Line", values = c(2,1), guide=guide_legend(override.aes = list(color = c("red")))) +
    theme_bw() +
    labs(
      caption=cond,
      x = "Probability of Superiority Responses",
      y = "Frequency"
    ) +
    facet_wrap( ~ ground_truth)
  print(plt)
}

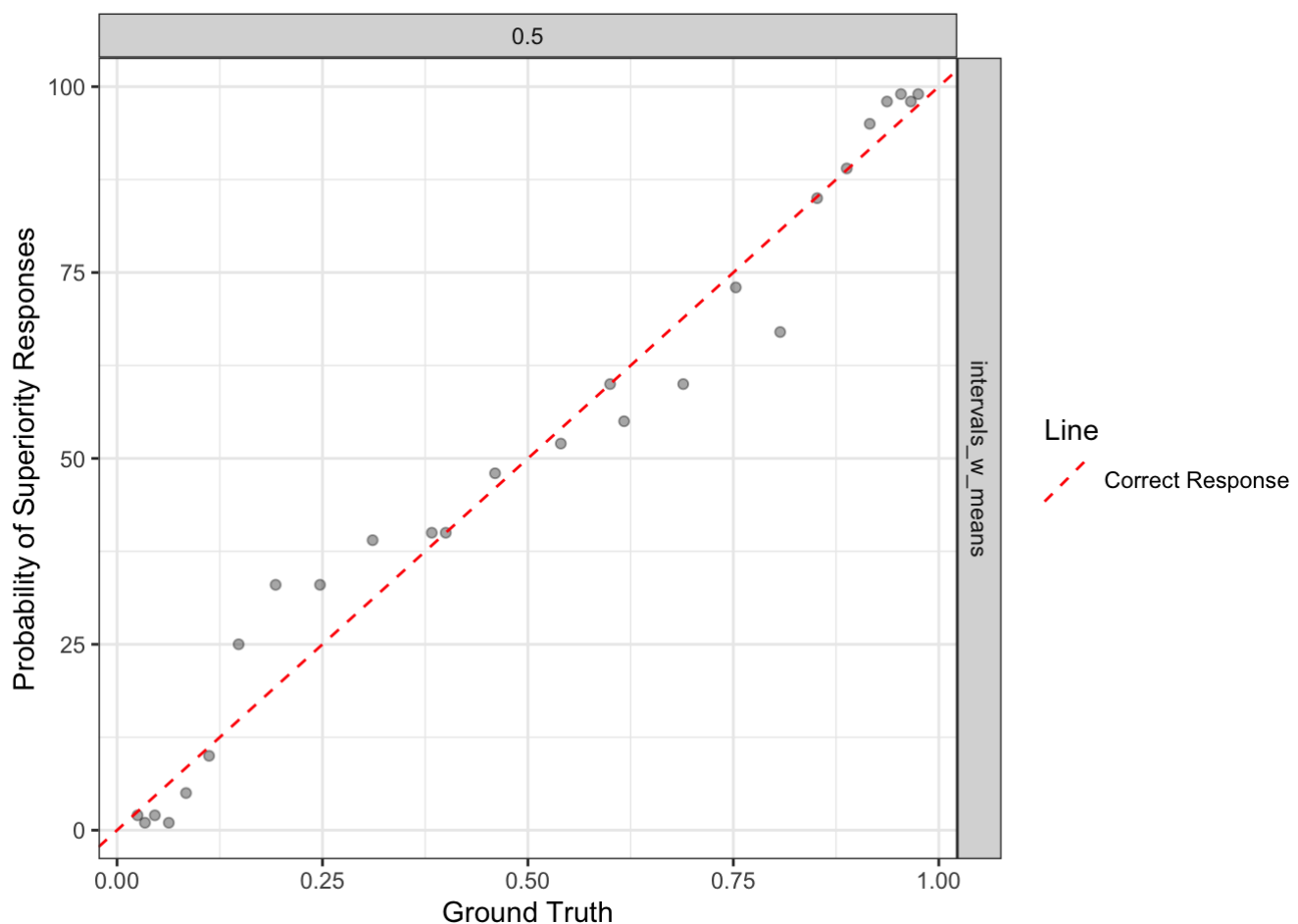
```



As we would expect based on a linear log odds representation of probability, probability of superiority judgments tend to be biased toward 50% relative to the ground truth. We can also see that responses are highly variable, with some responses falling completely on the wrong side of 50%, suggesting that some participants may have been confused by this question.

Another more compact way of looking at the relationship between estimated probability of superiority and the ground truth is to just plot them against one another. Let's look at this even though its sort of a mess.

```
# plot estimated probability of superiority vs the ground truth
responses_df %>%
  ggplot(aes(x = ground_truth, y = p_superiority)) +
  geom_point(alpha = 0.35) +
  geom_abline(aes(intercept = 0, slope = 100, linetype = "Correct Response"), color =
"red") +
  scale_linetype_manual(name = "Line", values = c(2,1), guide=guide_legend(override.a
es = list(color = c("red")))) +
  theme_bw() +
  labs(
    x = "Ground Truth",
    y = "Probability of Superiority Responses"
  ) +
  facet_grid(condition ~ baseline)
```



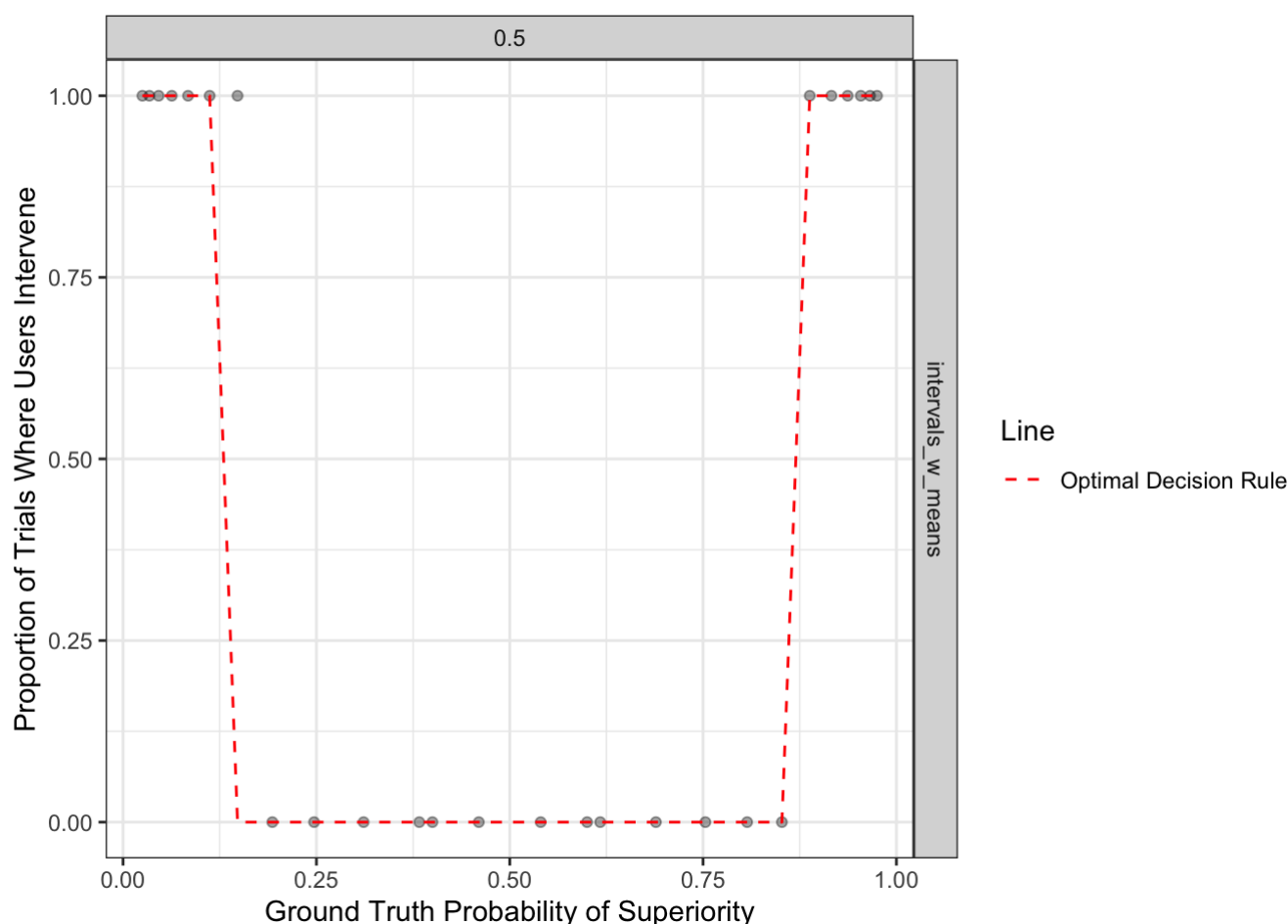
## Decisions to Intervene

In order to see how people are doing on the decision task, we want to benchmark their performance against a utility optimal decision rule. The rule is different depending on whether the task is framed as a gain or a loss (i.e., whether the ground truth probability of superiority is greater than or less than 50%).

```
# determine whether or not intervention is utility optimal on each trial
responses_df <- responses_df %>%
  mutate(should_intervene = if_else(ground_truth > 0.5,
    (p_contract_old + 1 / contract_value) < p_contract_new, # gain framing decision rule
    ((1 - p_contract_old) - 1 / contract_value) > (1 - p_contract_new))) # loss framing decision rule
```

Let's plot the proportion of users who intervene at each level of ground truth probability of superiority in each visualization \* baseline condition. People should intervene more often at extreme probabilities. We show the utility optimal decision threshold in red. This should give us an overview of decision quality.

```
# summarise the data as the overall proportion of trials where users intervene vs what they should do at each level of ground_truth * condition * baseline
responses_df %>%
  group_by(condition, baseline, ground_truth) %>%
  summarise(
    proportion_intervene = sum(intervene) / n(),
    optimal_decision = mean(should_intervene)
  ) %>%
  ggplot(aes(x = ground_truth, y = proportion_intervene)) +
  geom_point(alpha = 0.35) +
  geom_line(aes(y = optimal_decision, linetype="Optimal Decision Rule"), color="red")
+
  scale_linetype_manual(name="Line", values = c(2,1), guide=guide_legend(override.aes=
=list(color=c("red")))) +
  theme_bw() +
  labs(
    x = "Ground Truth Probability of Superiority",
    y = "Proportion of Trials Where Users Intervene"
  ) +
  facet_grid(condition ~ baseline)
```

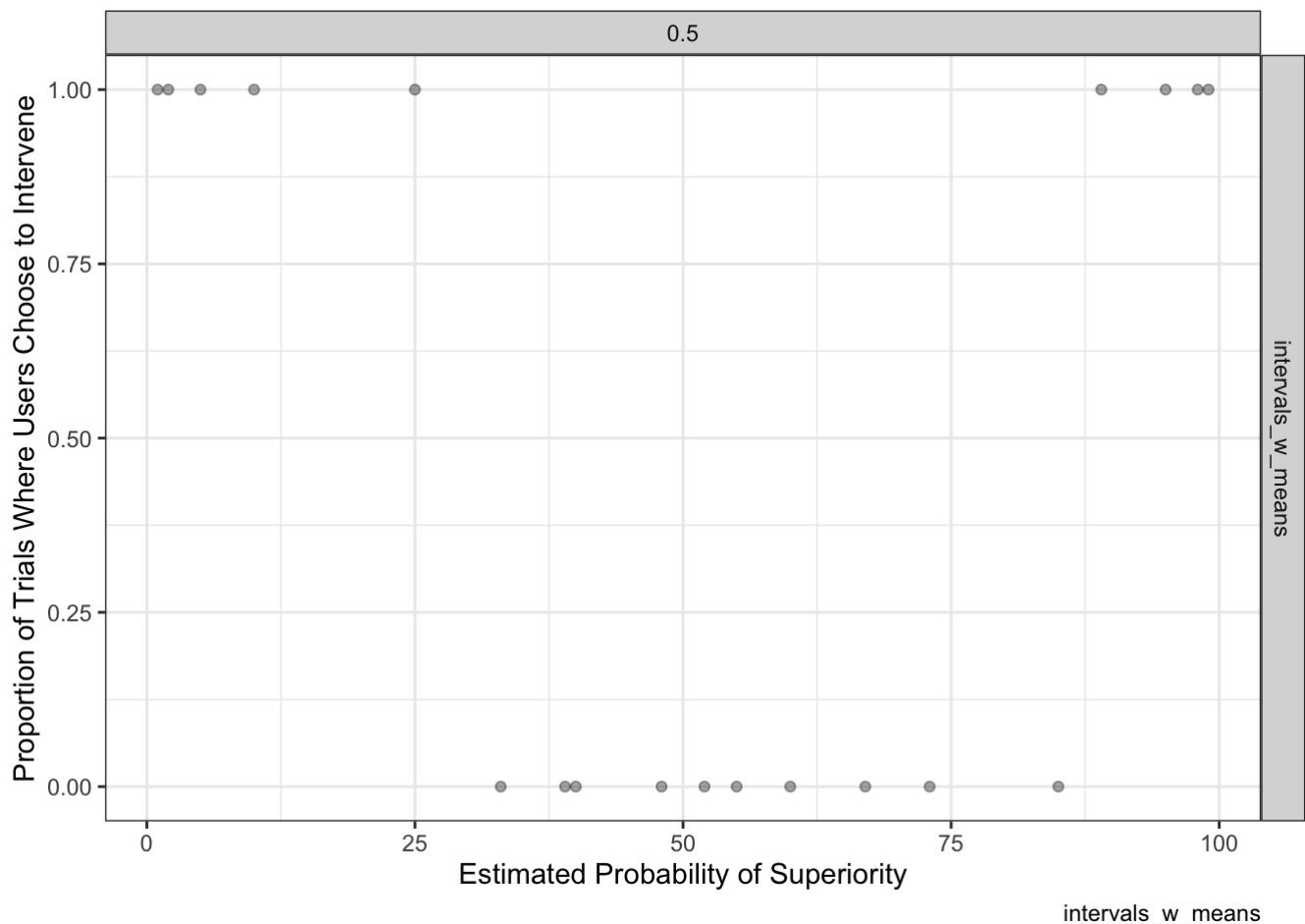


Unsurprisingly, users seem pretty bad at decision-making when they use only means to inform their reasoning. Interestingly, the baseline condition seems to have a large impact on decision quality (provisionally, people seem to make better decisions when entropy is lower or the baseline outcome is more certain).

# Probability of Superiority Judgments vs Decisions to Intervene

It might also be interesting to see how decisions correspond to probability of superiority judgments. We omit the ground truth and optimal decision rule from this chart.

```
# summarise the data as the overall proportion of trials where users choose to intervene at each level of condition * baseline * p_superiority
responses_df %>%
  group_by(condition, baseline, p_superiority) %>%
  summarise(proportion_intervene = sum(intervene) / n()) %>%
  ggplot(aes(x = p_superiority, y = proportion_intervene)) +
  geom_point(alpha = 0.35) +
  theme_bw() +
  labs(
    caption=cond,
    x = "Estimated Probability of Superiority",
    y = "Proportion of Trials Where Users Choose to Intervene"
  ) +
  facet_grid(condition ~ baseline)
```



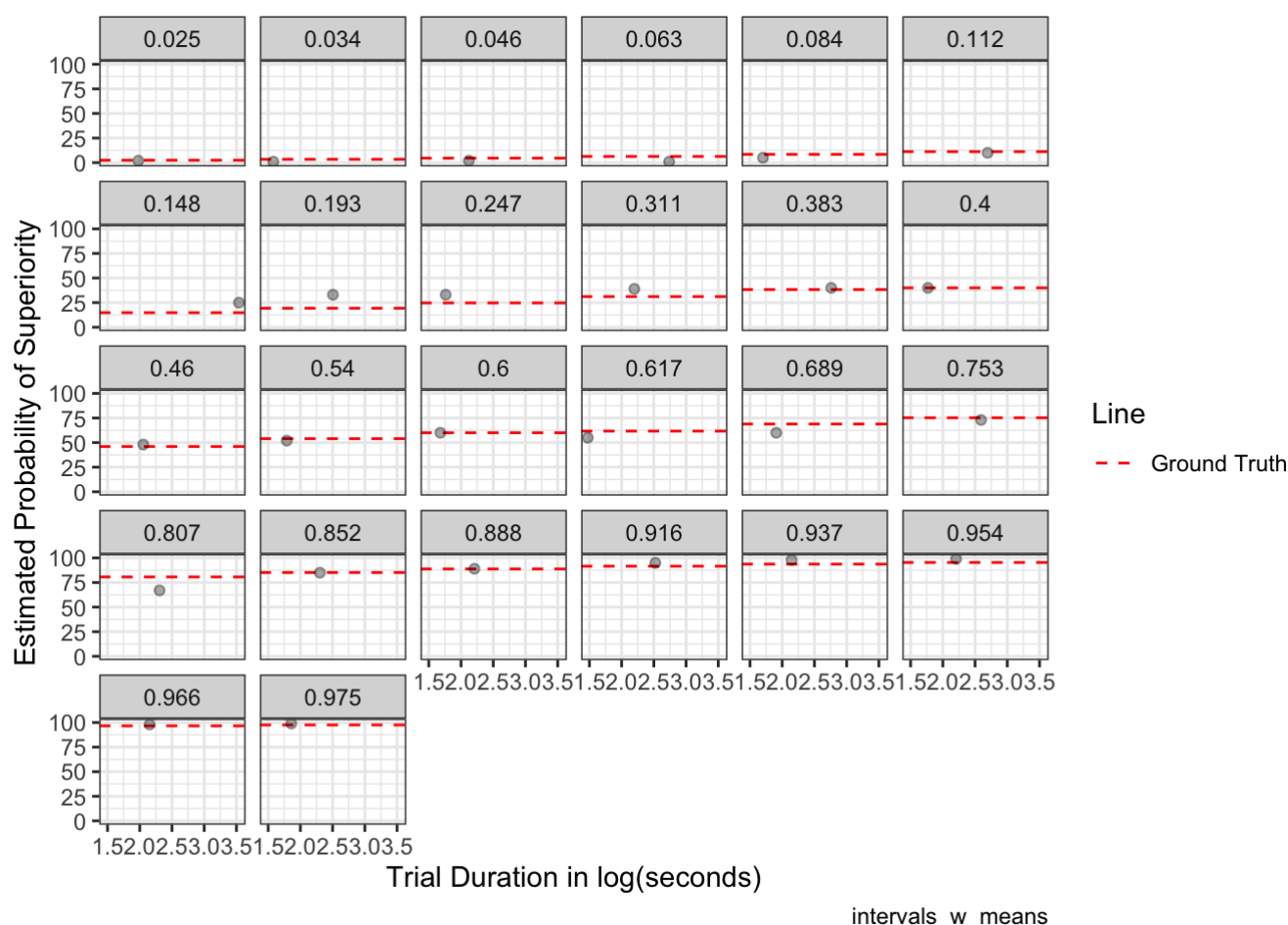
## Relationships with Trial Duration

We want to know when, if at all, spending more time on a response results in improved performance.

### Trial Duration vs Probability of Superiority Judgments

Let's look at probability of superiority estimates as a function of trial duration. As before, we show the ground truth in red and separate visualization conditions into different views to limit the number of faceted subplots in a single view.

```
for (cond in unique(responses_df$condition)) {
  plt <- responses_df %>% filter(condition == cond) %>%
    ggplot(aes(x = log(trial_dur), y = p_superiority)) +
    geom_hline(aes(yintercept = ground_truth * 100, linetype = "Ground Truth"), color
= "red") +
    scale_linetype_manual(name = "Line", values = c(2,1), guide=guide_legend(override.aes = list(color = c("red")))) +
    geom_point(alpha = 0.35) +
    theme_bw() +
    labs(
      caption=cond,
      x = "Trial Duration in log(seconds)",
      y = "Estimated Probability of Superiority"
    ) +
    facet_wrap( ~ ground_truth)
  print(plt)
}
```



Trial duration seems mostly unrelated to probability of superiority judgments except for in the case of HOPs, where responses seem to cluster closer to the ground truth on longer trial durations (with some exceptions). We should expect that people will have more accurate perceptions of probability the longer they watch HOPs.

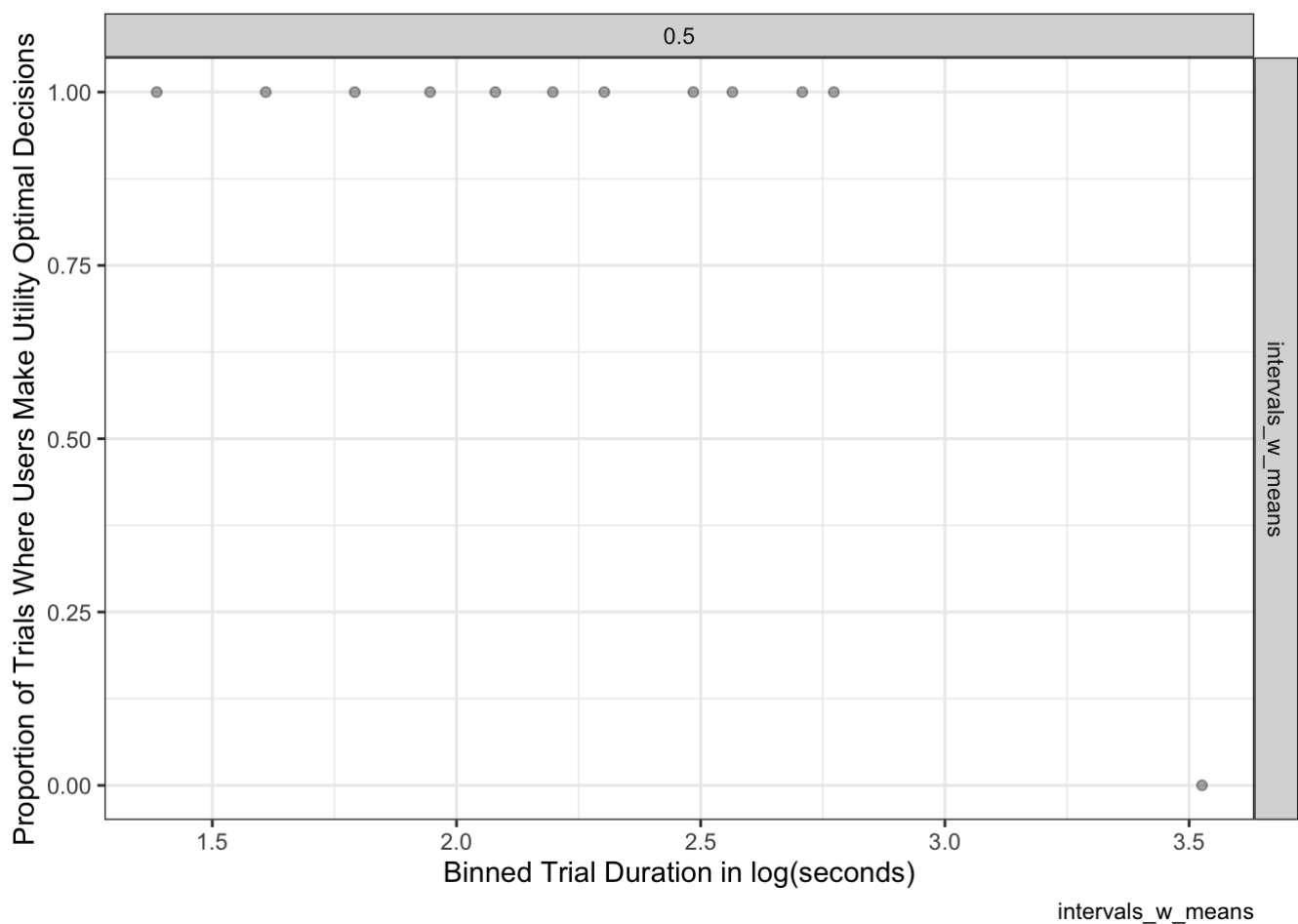
## Trial Duration vs Decision Quality

A nice metric for decision quality is whether users responded “correctly” or in line with the normative utility optimal decision rule. We calculate whether the user was “correct” or not on each trial.

```
# determine whether response on each trial is utility optimal
responses_df <- responses_df %>%
  mutate(correct = intervene == should_intervene)
```

Let's look at the proportion correct as a function of trial duration, faceting baseline and visualization conditions as above.

```
# summarise the data as the overall proportion of trials where users make utility opt
imal decisions at each level of condition * baseline * trial_dur
responses_df %>%
  mutate(trial_dur_binned = round(trial_dur)) %>%
  group_by(condition, baseline, trial_dur_binned) %>%
  summarise(proportion_correct = sum(correct) / n()) %>%
  ggplot(aes(x = log(trial_dur_binned), y = proportion_correct)) +
  geom_point(alpha = 0.35) +
  theme_bw() +
  labs(
    caption=cond,
    x = "Binned Trial Duration in log(seconds)",
    y = "Proportion of Trials Where Users Make Utility Optimal Decisions"
  ) +
  facet_grid(condition ~ baseline)
```



Trial duration seems to have little to do with decision quality.

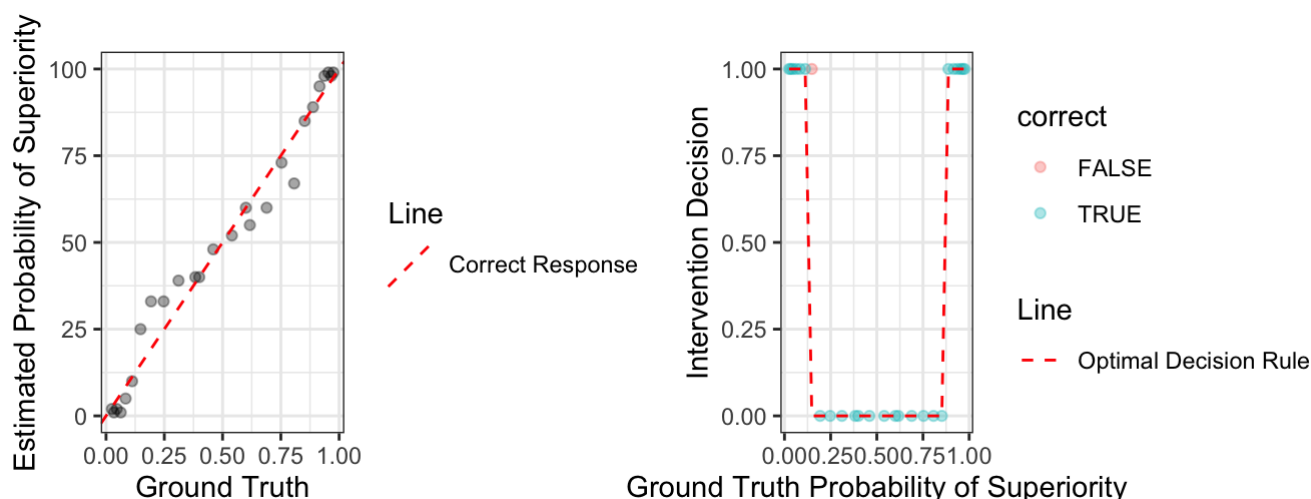
## Individual Patterns of Behavior



As is often the case with judgments from visualizations, the data seem highly heterogeneous. We try to get a sense of this by looking at individual patterns of responses in conjunction with individual characteristics such as gender, age, education, chart use, and numeracy. Below we create an overview of performance and individual characteristics for each participant separately.

```
for (worker in unique(responses_df$worker_id)) {
  # get a df for just this worker
  worker_df <- responses_df %>% filter(worker_id == worker)
  # plot probability of superiority judgments vs ground truth
  p_sup_plt <- worker_df %>%
    ggplot(aes(x = ground_truth, y = p_superiority)) +
    geom_point(alpha = 0.35) +
    geom_abline(aes(intercept = 0, slope = 100, linetype = "Correct Response"), color
= "red") +
    scale_linetype_manual(name = "Line", values = c(2,1), guide=guide_legend(override.aes = list(color = c("red")))) +
    theme_bw() +
    ylim(0, 100) +
    labs(
      x = "Ground Truth",
      y = "Estimated Probability of Superiority"
    )
  # plot intervention decisions vs ground truth, noting which are in line with the utility optimal decision rule
  decision_plt <- worker_df %>%
    ggplot(aes(x = ground_truth, y = intervene, color = correct)) +
    geom_point(alpha = 0.35) +
    geom_line(aes(y = as.numeric(should_intervene), linetype="Optimal Decision Rule"), color="red") +
    scale_linetype_manual(name="Line", values = c(2,1), guide=guide_legend(override.aes=list(color=c("red")))) +
    theme_bw() +
    labs(
      x = "Ground Truth Probability of Superiority",
      y = "Intervention Decision"
    )
  # create a table summarizing this worker
  summary_table <- worker_df %>%
    group_by(worker_id) %>%
    summarise(
      condition = unique(condition),
      baseline = unique(baseline),
      gender = unique(gender),
      age = unique(age),
      education = unique(education),
      chart_use = unique(chart_use),
      numeracy = unique(numeracy)
    ) %>%
    select(-worker_id) %>%
    ggtexttable(rows = NULL, theme = ttheme("blank"))
  # stitch together these three views
  charts <- ggarrange(p_sup_plt, decision_plt, ncol = 2, nrow = 1)
  figure <- ggarrange(summary_table, charts, ncol = 1, nrow = 2)
  print(figure)
}
```

condition	baseline	gender	age	education	chart_use	numeracy
intervals_w_means	0.5	M	25-34	Bachelor's degree	Daily	11



Looking at these plots for individual participants, we can see that not everyone was shown all 24 levels of ground truth as intended. This is due to a bug in the interface code that reshuffled the trial set throughout the experiment for some participants. This was fixed after HIT assignment batch 7. Let's see what the damage looks like. How many participants were shown various numbers of duplicate trials?

```
# create a grid of worker ids * trial indices, every trial that should exist
trials_should_exist_df <- data_grid(responses_df, worker_id = unique(worker_id), trial_idx = unique(trial_idx))

# check the number of times each worker was shown each trial, and plot the number of workers shown various numbers of duplicates
responses_df %>% select(worker_id, trial_idx) %>% filter(trial_idx != "practice") %>%
  right_join(trials_should_exist_df, by = "worker_id") %>%
  group_by(worker_id, trial_idx) %>%
  summarise(n_times_shown_trial = sum(trial_idx == trial_idx)) %>%
  group_by(worker_id) %>%
  summarise(n_duplicates = sum(n_times_shown_trial > 1)) %>%
  ggplot(aes(x = n_duplicates)) +
  geom_histogram(aes(y = ..count..), binwidth = 1, fill="black", col="grey") +
  theme_bw() +
  labs(
    x = "Number of Duplicate Trials",
    y = "Count of Participants"
  )
```

