

Homework 3: Do Trivia Nerds Cheat?

Neeraj Sharma

05/19/2020

Question 1: Clean data and report summary statistics of percent correct answers by year and round of the championship, as well as when these players are on the honor system. Provide hypotheses as to why these summary percentages might vary.

In order to clean this data set, the first thing I do is convert it from a wide data structure to a tidy structure. Several relational variables like name, year, round, and honorsystemcorrect remain fixed, but I pivot questions vertically to make question number a unique variable. The second cleaning action I perform is I convert the data type of the question number column I just created by removing the “Q” at the beginning of each entry and coercing the datatype. The third modification I perform to clean the data is I coerce all “-99” observations in the `honorsystemcorrect` column to be NA. Often times, -99 is code for missing values.

With the cleaned data, I was able to get insight into the aggregate performance of competitors in each year.

Table 1: Summary Statistics of Correct Answers by Year and Round of the Championship

Round	2018						2019					
	Questions Overall		Individual Performance				Questions Overall		Individual Performance			
	N	% Correct	Min	Max	Mean	St Dev	N	% Correct	Min	Max	Mean	St Dev
1	1668	43.0%	0	12	5.16	2.81	1968	64.5%	3	12	7.74	2.12
2	1668	51.0%	0	11	6.12	2.75	1968	47.9%	1	12	5.74	2.43
3	840	60.4%	3	11	7.24	1.99	984	60.7%	2	12	7.28	2.20
4	840	45.4%	0	11	5.44	2.07	984	50.2%	2	12	6.02	2.18

In both years, the round to round percentage of questions answered correctly fluctuated

Table 2: Summary Statistics of Honor System Performance by Year and Round

Round	2018				2019			
	Mean	Median	Min	Max	Mean	Median	Min	Max
1	79.7%	79.8%	65.3%	92.7%	79.5%	79.6%	64.8%	92.7%
2	79.7%	79.8%	65.3%	92.7%	79.5%	79.6%	64.8%	92.7%
3	80.7%	80.9%	65.3%	92.7%	80.7%	80.1%	64.8%	92.7%
4	80.7%	80.9%	65.3%	92.7%	80.7%	80.1%	64.8%	92.7%

Question 2: In the first two championship rounds, estimate how difficult the championship questions are relative to the regular season questions for people who play honestly during the regular season. Suggest at least two strategies for coming up with such an estimate. Be extremely explicit about the assumptions for each of your strategies to yield truthful estimates. Given the likely violation of your assumptions, say whether your estimates overestimate or underestimate the true amounts of cheating.

In office hours with Eric on 5/18, there was some confusion about the interpretation of this question. I interpret this question to ask us to find two distinct methods of identifying cheaters and then to come up with way to identify the increase in question difficulty based on the subset of fair players.

I think this this approach to this question makes a lot of sense. In order to quantify the impact that a change has on a sample group, one needs to only change that variable and keep everything else as stable as possible. In this case, cheaters have two dimensions of change: they go from cheating to not cheating, and “easy” questions to “hard” questions. For fair players, they only have one dimension of change: they go from easy questions to hard questions. Because fair players only experience one type of shift, the difference between their honor system performance and championship performance has only one motivating factor, while the cheaters have two factors driving differences in their performance.

Thus, identifying fair players is a vital and technical step to understanding how difficult these two phases of competition are. Once I have controlled for the participants’ cheating tendencies, I can evaluate the magnitude and impact of the different types of questions in isolation.

I make two overarching assumptions that are resonable, but I would like to formally state them just for completeness.

1. I assume that people try as hard as possible in the championship rounds and that their performance there is closely representative of their true potential.
2. I assume that people only cheat in ways that will improve their performance. This rules out Google giving people wrong answers more frequently than they would give wrong answers themselves.

Step 1: Identify Cheaters (Three Strategies)

Approach 1: Assume superior regular season performance is evidence of cheating

One possible strategy is to distinguish between cheaters and non-cheaters by calculating each person’s percentage of correct answers given in the first two rounds and comparing that number to their honor system answers. If we assume that their scores in live competition represent a firm upper limit on their ability, than any individual who performed better at home must have cheated.

This assumes highly optimized behavior because it requires players to achieve and exhibit their full potential in a competitive environment.

This assumes that every player is playing at their full potential in the first two rounds of the championship. Secondly, it assumes that

Approach 2: High honor system scores but low scores on easy championship questions as evidence of cheating

Honestly, if you are messing up on lots of easy questions, you are bad at trivia. People that are cheat will mess up on easy questions much less than they would otherwise, so identifying questions A question is difficult if very few people get it right.

Approach 3: Segment population into cheaters and fair players via T-Testing

An optimal strategy is to identify people whose play differs (statistically) significantly in the tournament from their play on the honor system. Specifically, I segment people by performing a one-tailed one-sample t-test for every individual. This t-test is one tailed because I am only interested in people who perform significantly worse in the championship than at home. Superior performance in the championship compared to performance at home is legitimate due to the live broadcast. I perform a one-sample t-test because I assume that the number of questions they answer at home is sufficiently large to approximate a population mean for that individual person. This is realistic, as Professor Levitt indicated that every participant had answered hundreds to thousands of questions at home during the regular season. In summary, these t-tests will tell us if an individual's performance in competition is in line with their performance at home, or if their performance in competition is statistically significantly worse than their performance at home.

Step 2: Difficulty between championship questions relative to the regular season questions (1 Strategy)

Approach 1: Difference between regular season and championship scores for fair players

As discussed above, fair players only experience one shift in going from the regular season to championship. Thus, any difference in the mean values of these two is only due to different types of questions, not the Thus, this difference in each form of partitioning players is the estimated average increase in question difficulty.

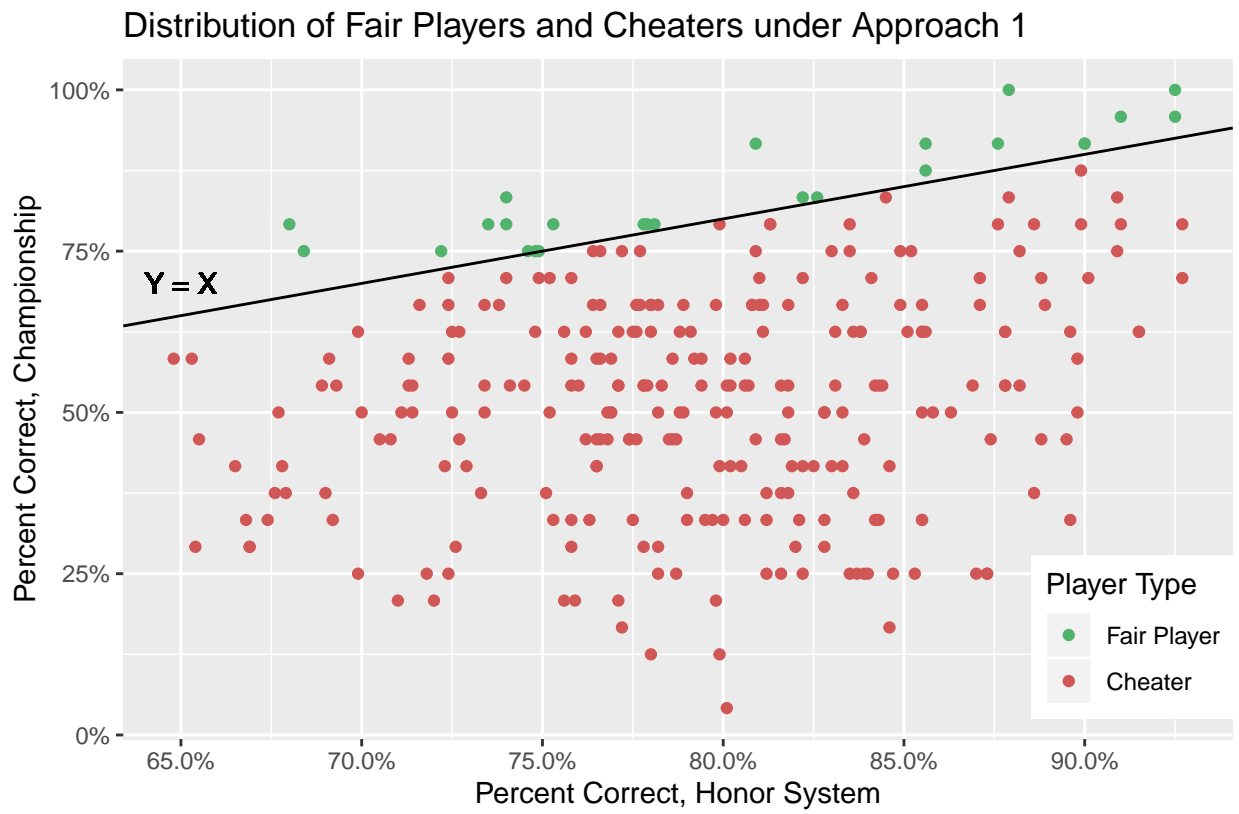
Question 3: Report your findings from the strategies in from question 2. (Hint: the most sensible way to report your findings would be a predicted value for the percent of questions you would predict each player have gotten correct over the first two rounds if they were not cheating in the regular season.). For each strategy you used, answer the following questions: what is your estimate of the average percent of questions that are cheated on for the entire group during the regular season? What percent of the players do you think cheat on at least 3 percent of the regular season questions? How many individual players can you say cheat with a high degree of confidence?

Approach 1

```
app1 <- edited %>%
  filter(round %in% c(1, 2)) %>%
  group_by(year, name, honorsystemcorrect) %>%
  summarize(pct_rt = mean(ans, na.rm = TRUE)) %>%
  drop_na() %>%
  mutate(cheater = if_else(honorsystemcorrect > pct_rt, 1, 0))

app1 %>%
  ggplot(aes(x = honorsystemcorrect, y = pct_rt, color = factor(cheater))) +
  geom_point() +
  geom_abline(slope = 1) +
  geom_text(aes(0.65, 0.70, label = "Y = X"), color = "black") +
  scale_x_continuous(labels = scales::label_percent()) +
```

```
scale_y_continuous(labels = scales::label_percent()) +
scale_color_manual(name = "Player Type",
  labels = c("Fair Player", "Cheater"),
  values = c("#52b36c", "#d15656")) +
labs(title = "Distribution of Fair Players and Cheaters under Approach 1",
  x = "Percent Correct, Honor System",
  y = "Percent Correct, Championship",
  caption = "Source: LearnedLeague") +
theme(legend.position = c(0.91, 0.15))
```



```
mean(app1 %>% filter(cheater == 0) %>% magrittr::use_series(honorsystemcorrect))
```

```
## [1] 0.80476
```

```
mean(app1 %>% filter(cheater == 0) %>% magrittr::use_series(pct_rt))
```

```
## [1] 0.8466667
```

Approach 2

```
champ_difficulty <- edited %>%
  filter(round %in% c(1, 2)) %>%
  group_by(year, round, qno) %>%
  summarize(corr = sum(ans), asked = n()) %>%
  mutate(pct_corr = corr/asked,
  # I let difficulty be 1/pct_corr because I want its value to be higher for questions that peop
  difficulty = 1/pct_corr) %>%
```

```
select(year, round, qno, pct_corr, difficulty)
champ_difficulty
```

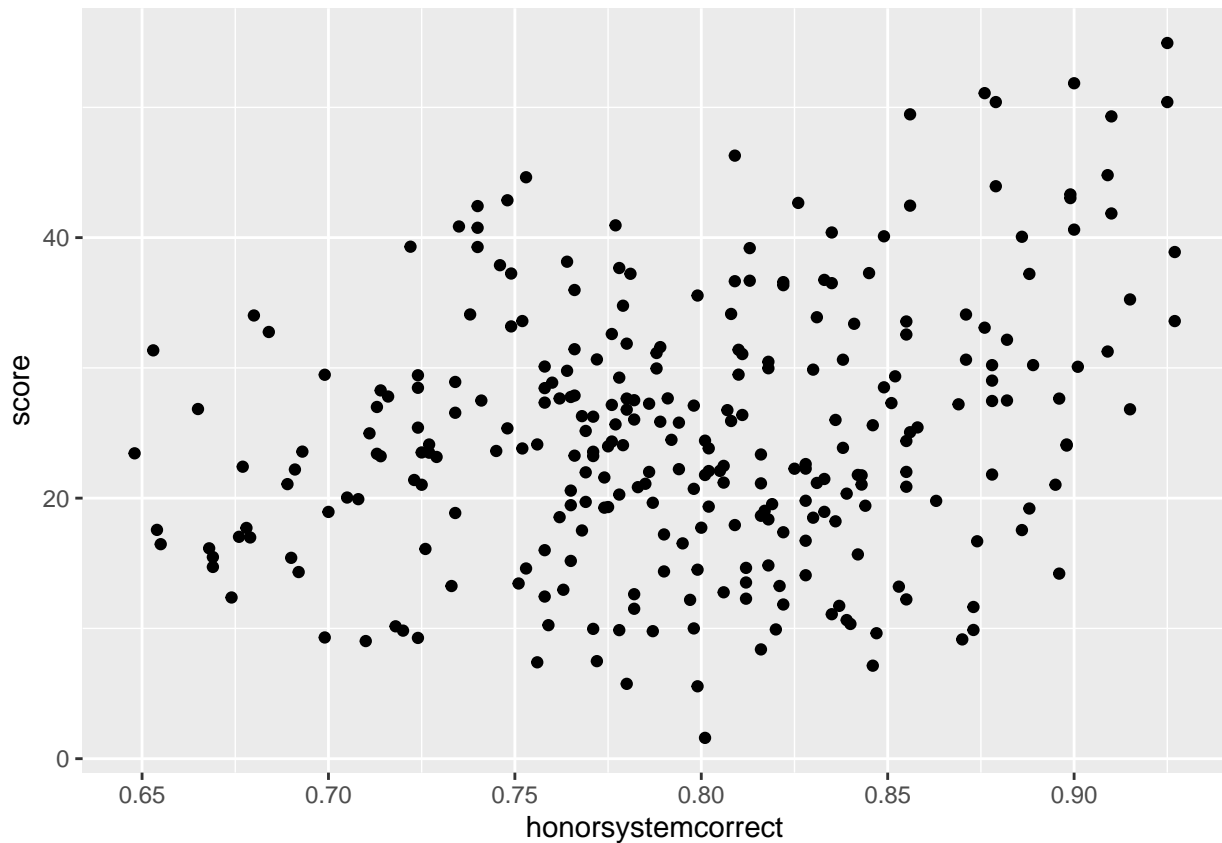
```
## # A tibble: 48 x 5
## # Groups:   year, round [4]
##   year round  qno pct_corr difficulty
##   <dbl> <dbl> <dbl>   <dbl>      <dbl>
## 1 2018     1     1   0.453      2.21
## 2 2018     1     2   0.165      6.04
## 3 2018     1     3   0.432      2.32
## 4 2018     1     4   0.453      2.21
## 5 2018     1     5   0.324      3.09
## 6 2018     1     6   0.446      2.24
## 7 2018     1     7   0.604      1.65
## 8 2018     1     8   0.281      3.56
## 9 2018     1     9   0.381      2.62
## 10 2018     1    10   0.640      1.56
## # ... with 38 more rows
```

Difficult questions are answered correctly a standard deviation less frequently than the mean of each

```
metrics <- champ_difficulty %>%
  group_by(year, round) %>%
  summarize(mean_corr = mean(pct_corr), sd = sd(pct_corr))
```

```
edited %>%
  drop_na(honorsystemcorrect) %>%
  filter(ans == 1) %>%
  inner_join(champ_difficulty) %>%
  group_by(name, year, honorsystemcorrect) %>%
  summarize(score = sum(difficulty)) %>%
  arrange(desc(score)) %>%
  ggplot(mapping = aes(x = honorsystemcorrect, y = score)) +
  geom_point()
```

```
## Joining, by = c("year", "round", "qno")
```



```
plot <- champ_difficulty %>%
  left_join(metrics) %>%
  mutate(difficulty = if_else(pct_corr > mean_corr + sd, 0, 1),
         difficulty = if_else(pct_corr < mean_corr - sd, 2, difficulty))
```

```
## Joining, by = c("year", "round")
```

```
plot
```

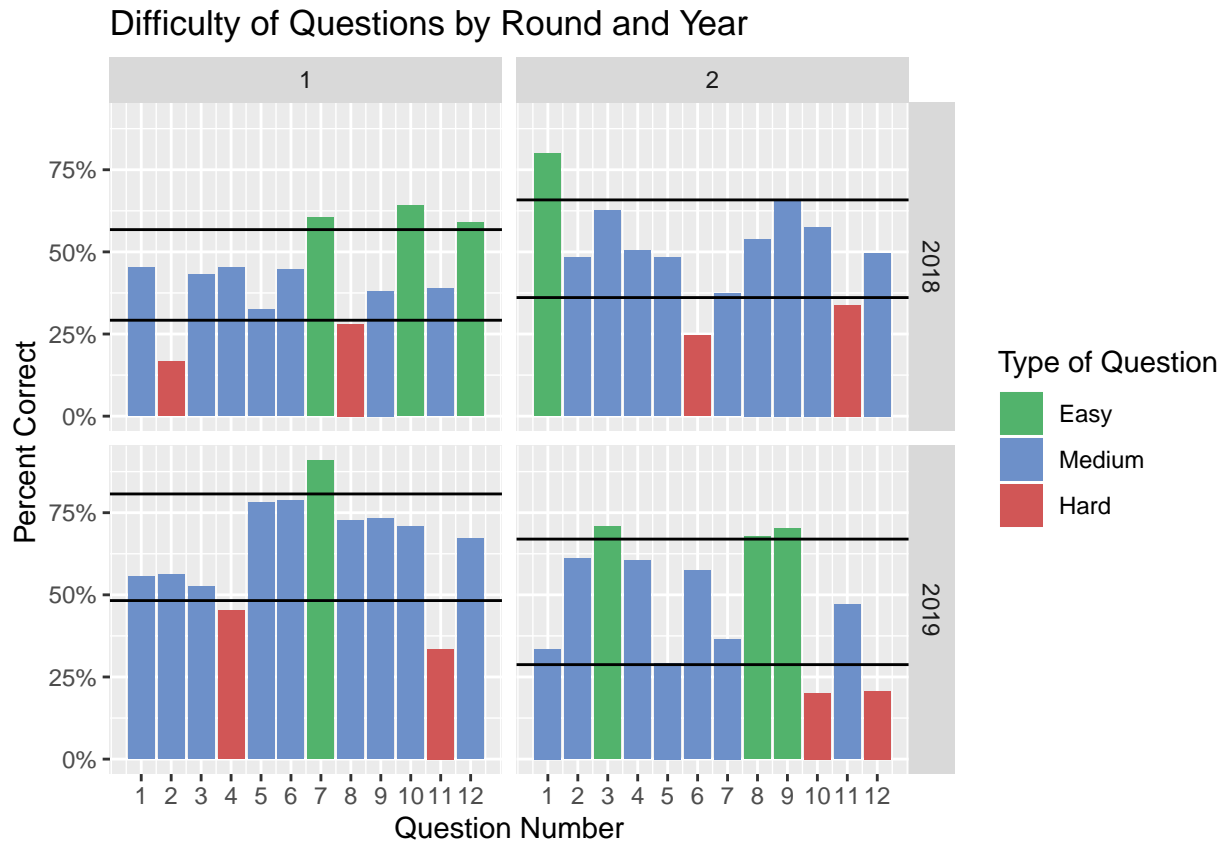
```
## # A tibble: 48 x 7
## # Groups:   year, round [4]
##   year round  qno pct_corr difficulty mean_corr    sd
##   <dbl> <dbl> <dbl>   <dbl>      <dbl>   <dbl> <dbl>
## 1  2018     1     1  0.453         1    0.430 0.138
## 2  2018     1     2  0.165         2    0.430 0.138
## 3  2018     1     3  0.432         1    0.430 0.138
## 4  2018     1     4  0.453         1    0.430 0.138
## 5  2018     1     5  0.324         1    0.430 0.138
## 6  2018     1     6  0.446         1    0.430 0.138
## 7  2018     1     7  0.604         0    0.430 0.138
## 8  2018     1     8  0.281         2    0.430 0.138
## 9  2018     1     9  0.381         1    0.430 0.138
## 10 2018     1    10  0.640         0    0.430 0.138
## # ... with 38 more rows
```

```
ggplot(plot, mapping = aes(x = qno, y = pct_corr, fill = factor(difficulty))) +
  geom_col() +
  geom_hline(plot, mapping = aes(yintercept = mean_corr + sd)) +
```

```

geom_hline(plot, mapping = aes(yintercept = mean_corr - sd)) +
scale_y_continuous(labels = scales::label_percent()) +
scale_x_continuous(breaks = seq(1, 12)) +
scale_fill_manual(name = "Type of Question",
                  labels = c("Easy", "Medium", "Hard"),
                  values = c("#52b36c", "#6d90c9", "#d15656")) +
labs(title = "Difficulty of Questions by Round and Year",
     x = "Question Number",
     y = "Percent Correct ") +
facet_grid(year ~ round)

```



```

easyqs <- plot %>%
  filter(difficulty == 0) %>%
  select(year, round, qno)

inner_join(easyqs, edited) %>%
  drop_na(honorsystemcorrect) %>%
  group_by(name, honorsystemcorrect) %>%
  summarize(count = sum(ans)) %>%
  mutate(num_easyqs = nrow(easyqs)) %>%
  arrange(-desc(count))

## Joining, by = c("year", "round", "qno")
## # A tibble: 232 x 4
## # Groups:   name [232]
##   name          honorsystemcorrect count num_easyqs

```

```
##      <chr>                <dbl> <dbl>      <int>
##  1 VolkA                  0.801    0         8
##  2 AldenR                  0.763    1         8
##  3 AvilaD                  0.790    1         8
##  4 CalcagnoR              0.759    1         8
##  5 ChildersM              0.733    1         8
##  6 di GiovanniaA          0.780    1         8
##  7 DwoskinS               0.818    1         8
##  8 HerderS                0.828    1         8
##  9 LachD                  0.855    1         8
## 10 LagardeW               0.87     1         8
## # ... with 222 more rows
```

```
## being good at trivia means you can answer lots of hard questions.
```

Approach 3

A better strategy is to run a one-sample t-test for every person's performance at home.

Using this approach, I identify a subset of people who I believe cheated in at home.

NOTE THAT THIS DATA ONLY INCLUDES PEOPLE WHO HAD PLAYED AT HOME AND PLAYED IN COMPETITION. THOSE WITHOUT VALUES FOR HONORSYSTEMCORRECT AND ANS WERE DISCOUNTED.

A one-sample t-test is used to compare the mean value of a sample with a constant value denoted μ_0 . The test has the null hypothesis that the population mean is equal to μ_0 and the alternative hypothesis that it is not equal to μ_0 . <http://www.instantr.com/2012/12/29/performing-a-one-sample-t-test-in-r/>

```
corr_r_1and2 <- edited %>%
  filter(round %in% c(1, 2)) %>%
  drop_na(honorsystemcorrect, ans) %>%
  group_by(year, name, honorsystemcorrect) %>%
  summarize(pct_rt = mean(ans))

tttester <- function(current_selection, year_sel) {
  obs <- edited %>%
    filter(round %in% c(1, 2),
           year == year_sel,
           name == current_selection)
  # Case where people get a perfect score in competition returns no std dev so
  # do not do a ttest in those situations
  if (sum(obs$ans) == 24) {
    return(1)
  }
  honsyscorr_num <- obs %>%
    magrittr::extract2(1,6)
  pval <- t.test(obs$ans, mu = honsyscorr_num, alternative = "less") %>%
    broom::tidy() %>%
    magrittr::extract2(1, 3)
}

pvals <- edited %>%
  drop_na(honorsystemcorrect, ans) %>%
  distinct(name, year) %>%
  mutate(pvals = map2(name, year, tttester),
```



```

    statsig = if_else(pvals < 0.05, 1, 0))

allplayers <- left_join(corr_r_1and2, pvals)

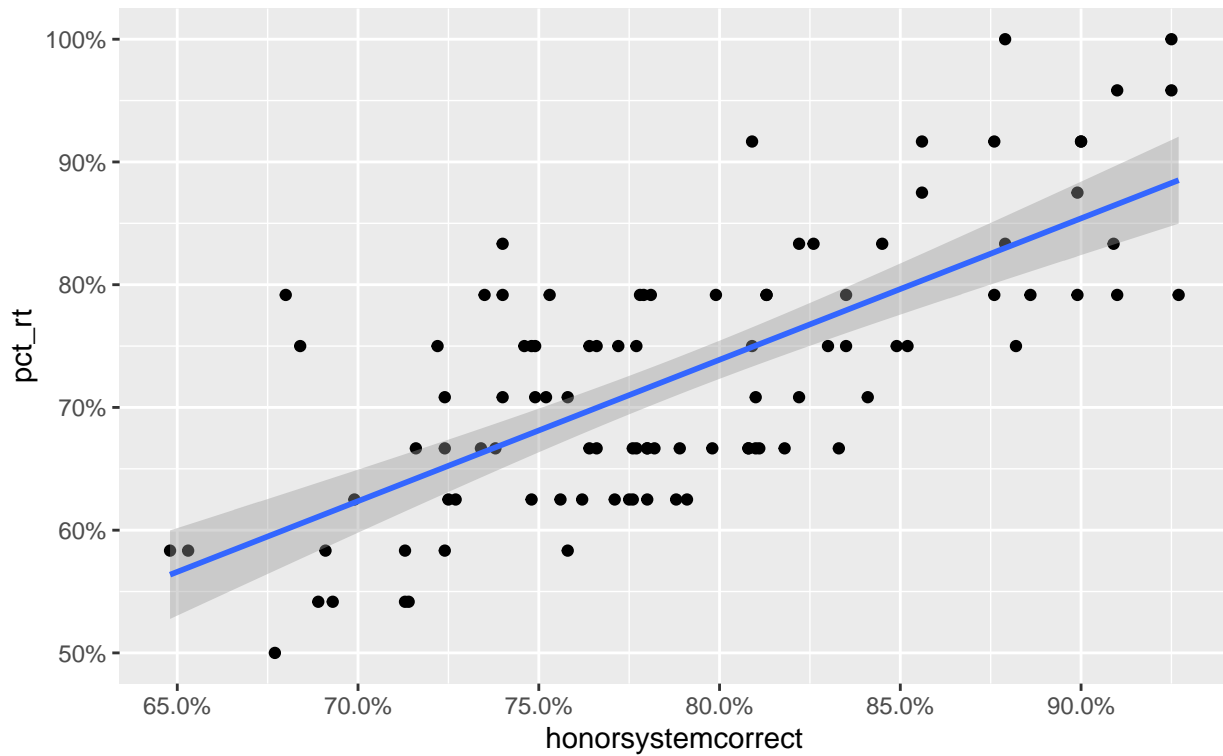
## Joining, by = c("year", "name")
# List of people that cheated
cheaters <- allplayers %>%
  filter(statsig == 1) %>%
  select(-statsig) %>%
  mutate(type = "cheater")

# List of people that didn't cheat
fairplayers <- allplayers %>%
  filter(statsig == 0) %>%
  select(-statsig) %>%
  mutate(type = "fair")
# %>%
# mutate(honorsystemcorrect = percent(honorsystemcorrect, accuracy = 0.1),
#        pct_rt = percent(pct_rt, accuracy = 0.1))

fairplayers %>%
  ggplot(aes(honorsystemcorrect, pct_rt)) +
  geom_point() +
  labs(title = "FAIR PLAYERS ONLY -- Percent right at home vs percent right in champ",
       subtitle = "Rounds 1 and 2 of Championship") +
  scale_x_continuous(labels = scales::label_percent()) +
  scale_y_continuous(labels = scales::label_percent()) +
  geom_smooth(method = "lm")

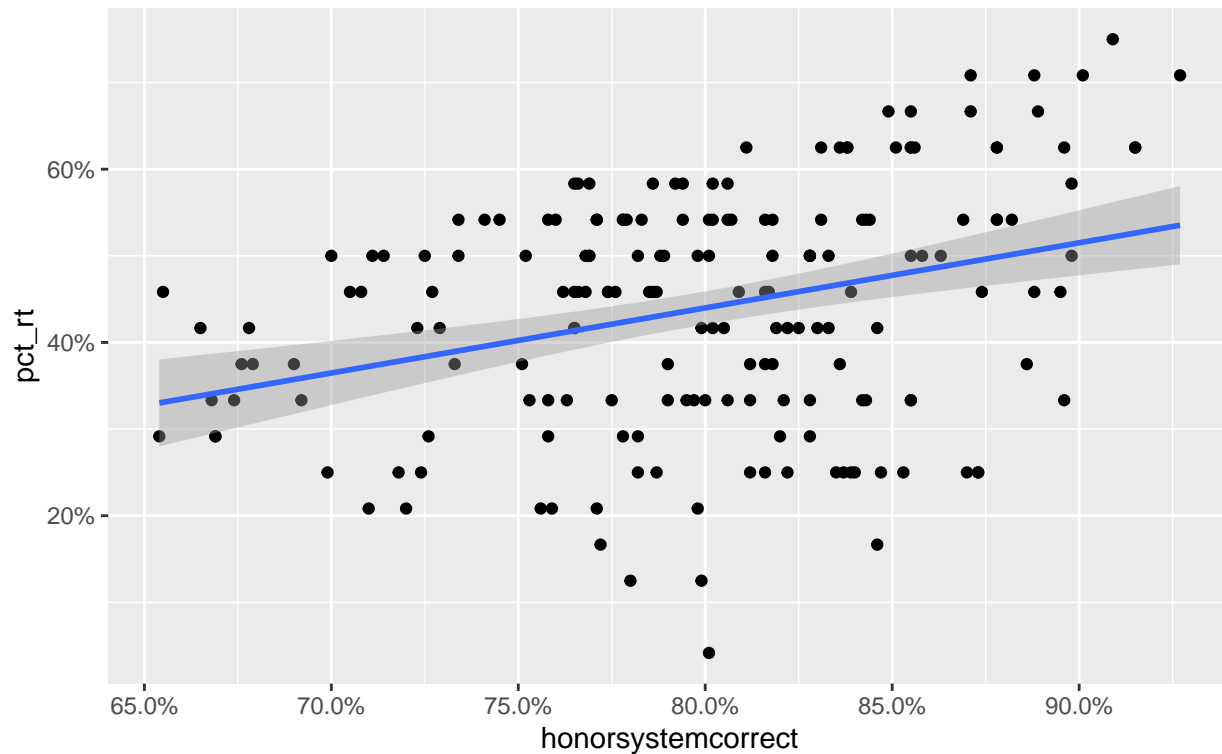
```

FAIR PLAYERS ONLY -- Percent right at home vs percent right in champ Rounds 1 and 2 of Championship



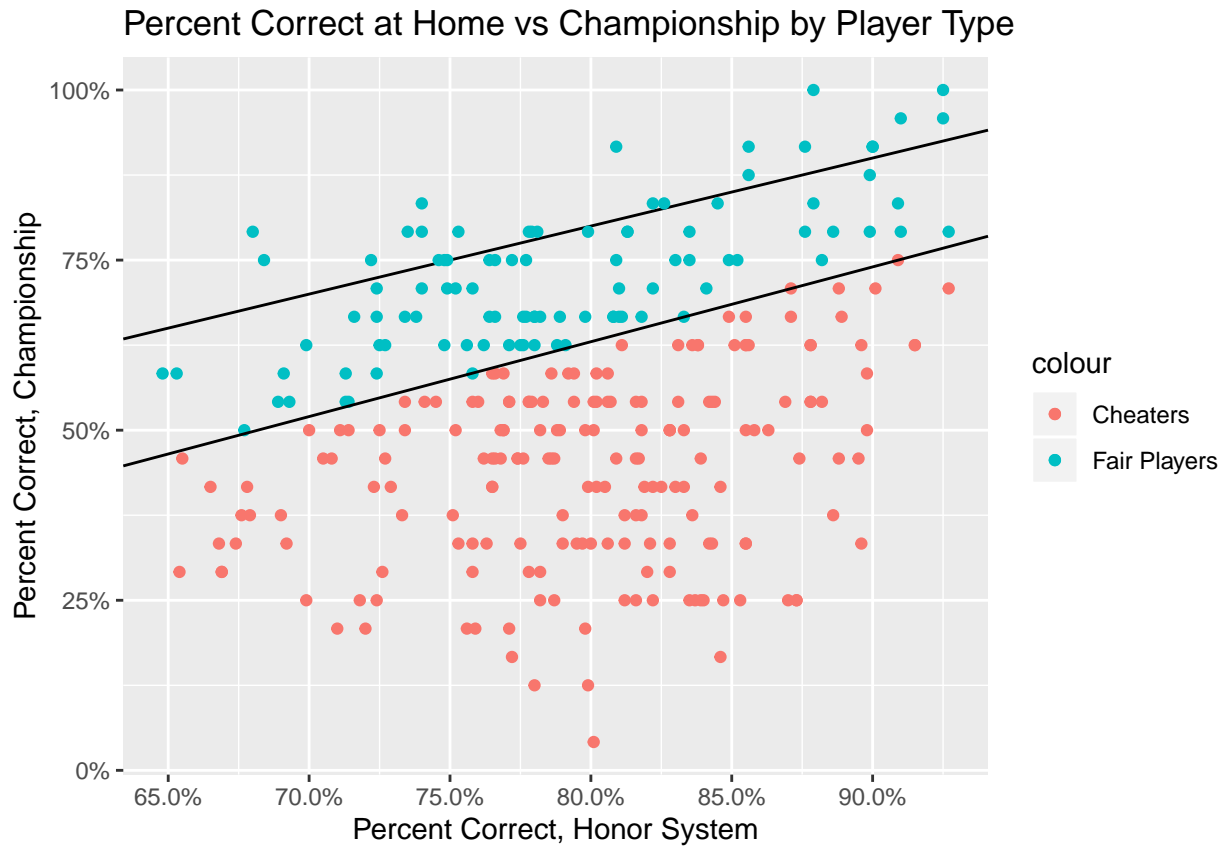
```
cheaters %>%
  ggplot(aes(honorsystemcorrect, pct_rt)) +
  geom_point() +
  labs(title = "Cheaters ONLY -- Percent right at home vs percent right in champ",
        subtitle = "Rounds 1 and 2 of Championship") +
  scale_x_continuous(labels = scales::label_percent()) +
  scale_y_continuous(labels = scales::label_percent()) +
  geom_smooth(method = "lm")
```

Cheaters ONLY — Percent right at home vs percent right in champ Rounds 1 and 2 of Championship



```
pp <- edited %>%
  filter(round %in% c(1, 2)) %>%
  group_by(year, name, honorsystemcorrect) %>%
  summarize(pct_rt = mean(ans)) %>%
  drop_na() %>%
  ggplot(aes(honorsystemcorrect, pct_rt)) +
  labs(x = "Percent Correct, Honor System",
       y = "Percent Correct, Championship") +
  geom_abline(slope=1) +
  scale_x_continuous(labels = scales::label_percent()) +
  scale_y_continuous(labels = scales::label_percent())

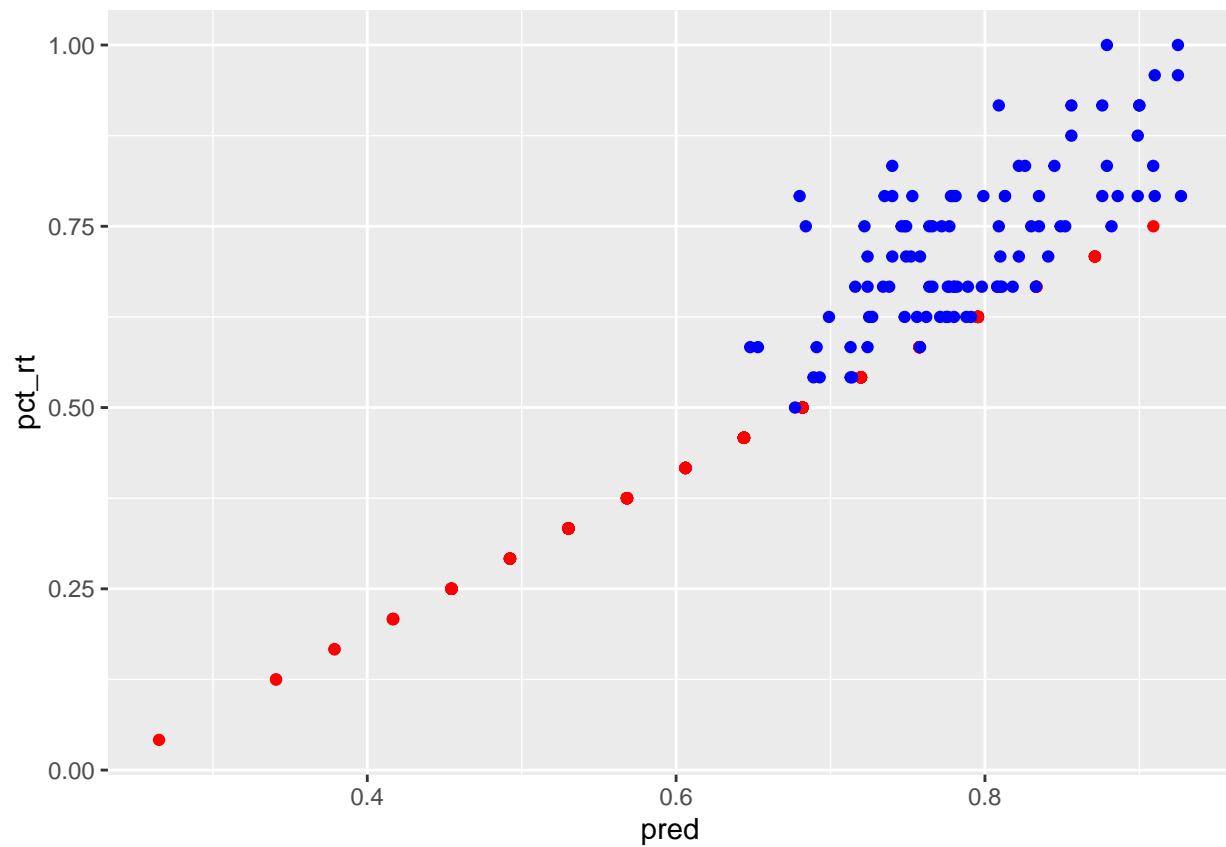
pp +
  geom_point(cheaters, mapping = aes(honorsystemcorrect, pct_rt, color = "Cheaters")) +
  geom_point(fairplayers, mapping = aes(honorsystemcorrect, pct_rt, color = "Fair Players")) +
  labs(title = "Percent Correct at Home vs Championship by Player Type") +
  geom_abline(slope = 1.1, intercept = -0.25)
```



```
# theme(legend.position = c(0.93, 0.1))
# Point on the far right side scored 95% but scored in competition like someone who scored 80% on the h

adj <- cheaters %>%
  mutate(pred = (pct_rt + 0.25)/1.1)

ggplot() +
  geom_point(adj, mapping = aes(pred, pct_rt), color = "red") +
  geom_point(fairplayers, mapping = aes(honorsystemcorrect, pct_rt), color = "blue")
```



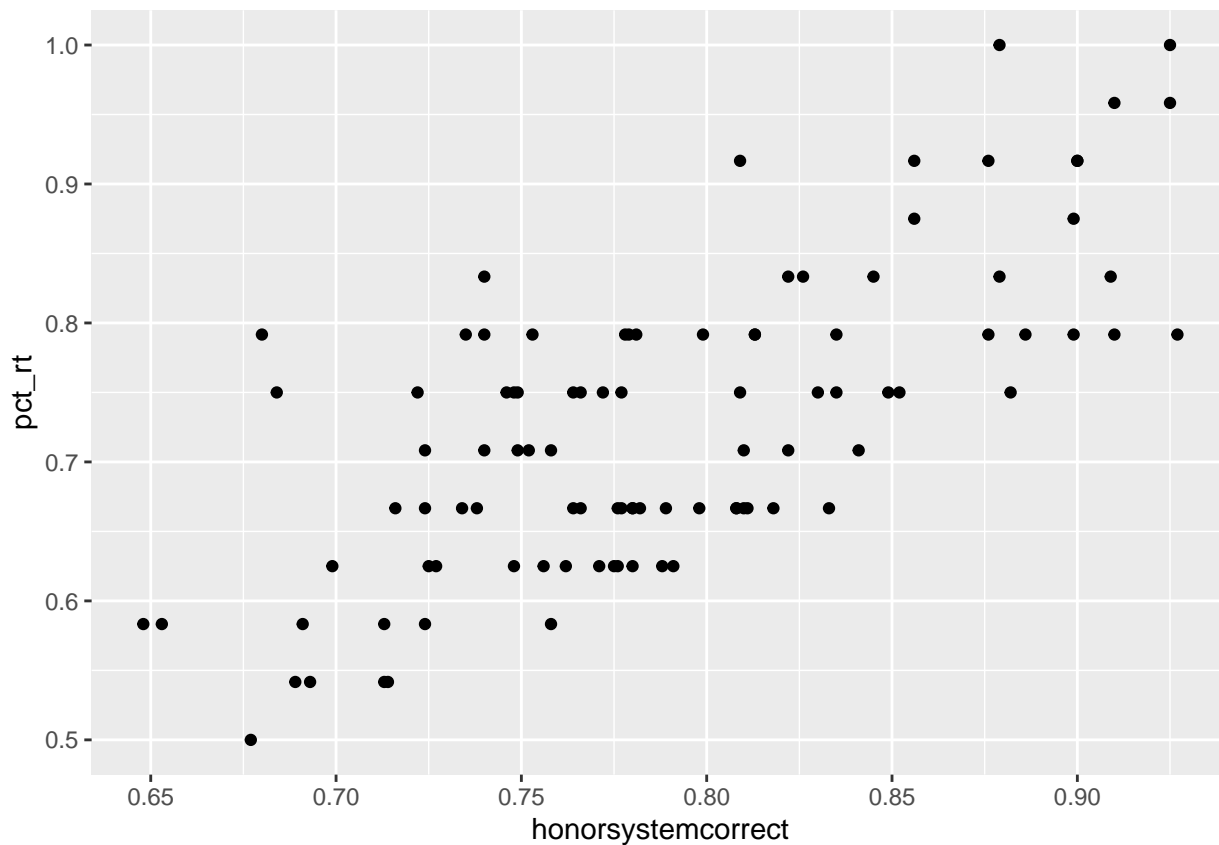
```
mean(fairplayers$honorsystemcorrect)
```

```
## [1] 0.7890204
```

```
mean(fairplayers$pct_rt)
```

```
## [1] 0.7261905
```

```
fairplayers %>%  
  ggplot(aes(honorsystemcorrect, pct_rt)) +  
  geom_point()
```

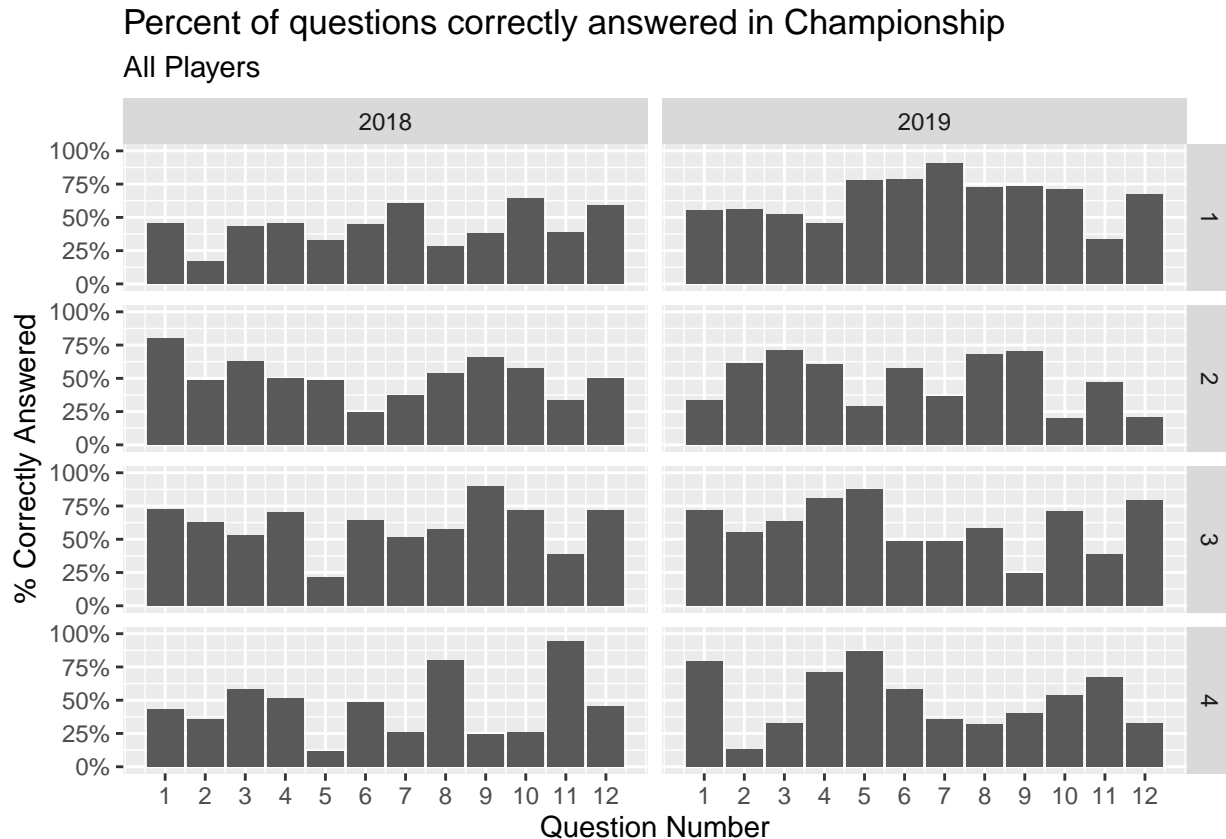


```
edited %>%
  filter(round %in% c(3, 4)) %>%
  drop_na(honorsystemcorrect, ans) %>%
  group_by(year, name, honorsystemcorrect) %>%
  summarize(pct_rt = mean(ans))
```

```
## # A tibble: 151 x 4
## # Groups:   year, name [151]
##   year name      honorsystemcorrect pct_rt
##   <dbl> <chr>          <dbl> <dbl>
## 1 2018 BahnamanS      0.899 0.833
## 2 2018 BerrettJ      0.776 0.458
## 3 2018 BlishS        0.835 0.792
## 4 2018 CannonS       0.802 0.333
## 5 2018 CareyC        0.769 0.417
## 6 2018 ChiltonC2     0.758 0.542
## 7 2018 CohenD2       0.831 0.625
## 8 2018 CraneN        0.713 0.333
## 9 2018 CurtinR       0.788 0.542
## 10 2018 DhuwaliaR     0.879 0.75
## # ... with 141 more rows
```

```
edited %>%
  filter(merge == "merged") %>%
  group_by(year, round, qno) %>%
  summarize(correct_ans = sum(ans), times_asked = n()) %>%
  ggplot(mapping = aes(x = qno, y = correct_ans/times_asked)) +
  geom_col() +
```

```
facet_grid(round ~ year) +
scale_x_continuous(breaks = seq(1, 12)) +
scale_y_continuous(limits = c(0, 1), labels = label_percent()) +
labs(title = "Percent of questions correctly answered in Championship",
      subtitle = "All Players",
      x = "Question Number",
      y = "% Correctly Answered")
```

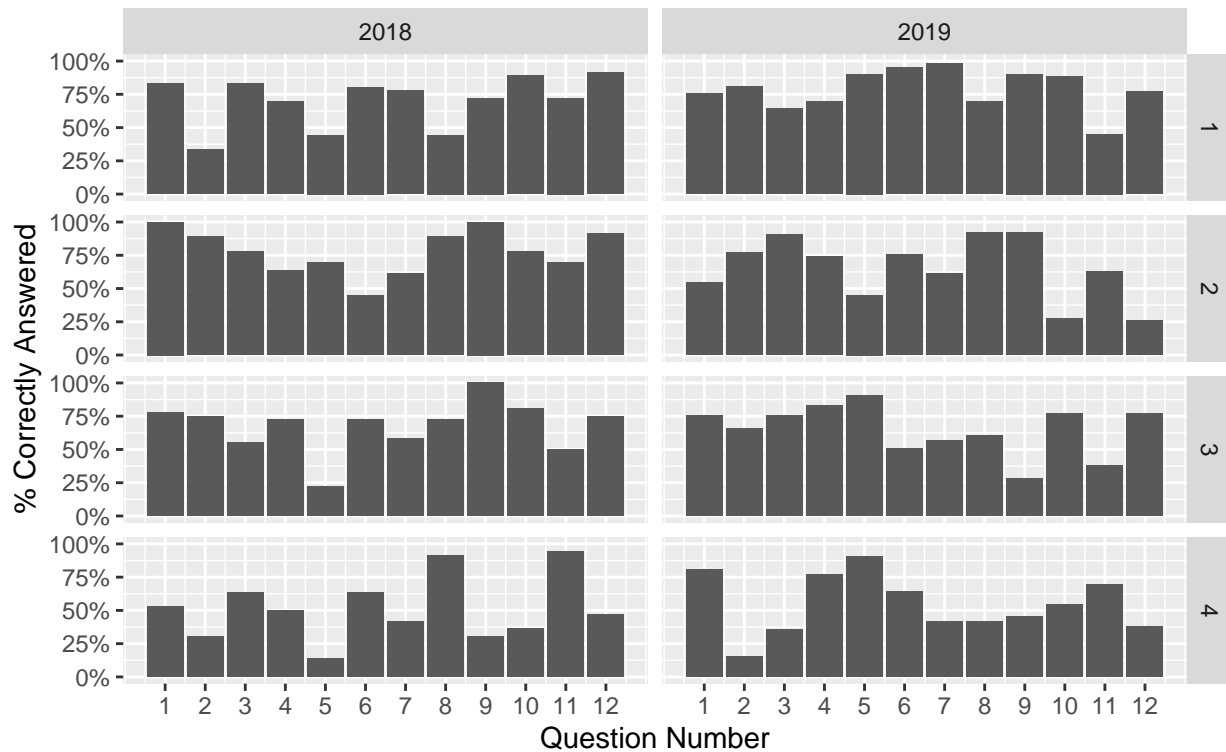


```
inner_join(fairplayers, edited) %>%
  group_by(year, round, qno) %>%
  summarize(correct_ans = sum(ans), times_asked = n()) %>%
  ggplot(mapping = aes(x = qno, y = correct_ans/times_asked)) +
  geom_col() +
  facet_grid(round ~ year) +
  scale_x_continuous(breaks = seq(1, 12)) +
  scale_y_continuous(limits = c(0, 1), labels = label_percent()) +
  labs(title = "Percent of questions correctly answered in Championship",
        subtitle = "Fair Players only",
        x = "Question Number",
        y = "% Correctly Answered")
```

```
## Joining, by = c("year", "name", "honorsystemcorrect")
```

Percent of questions correctly answered in Championship

Fair Players only

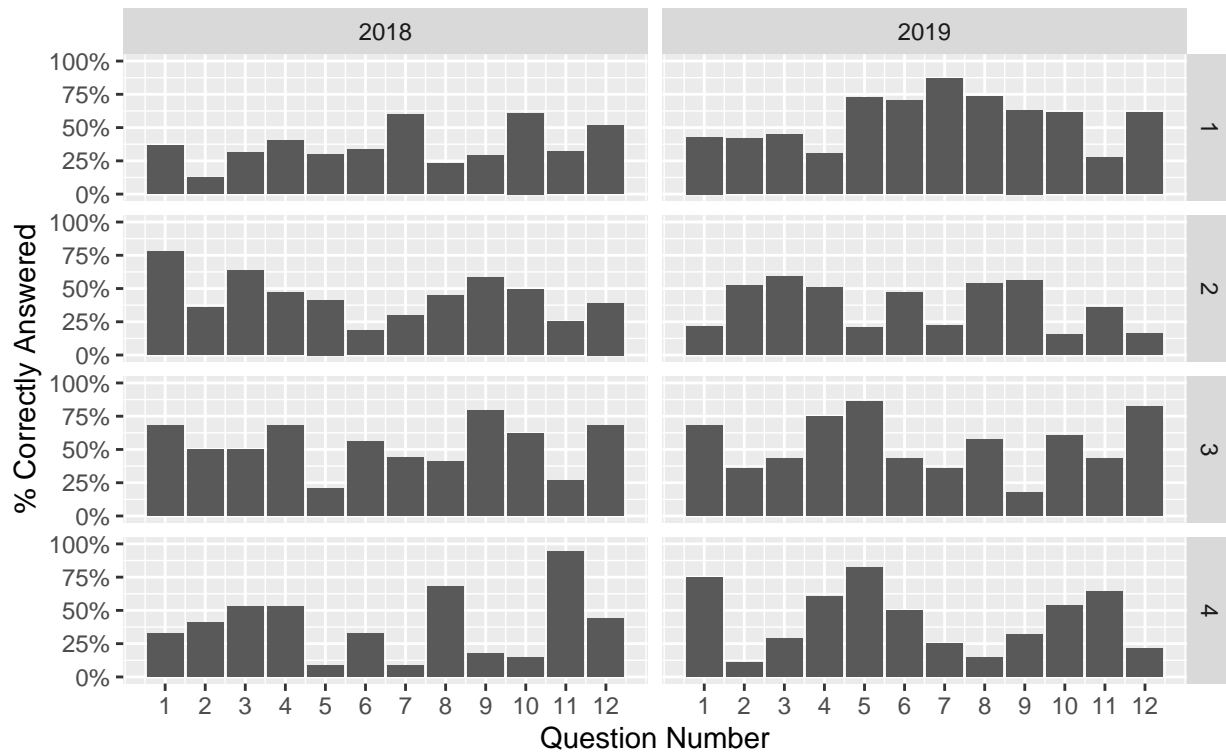


```
inner_join(cheaters, edited) %>%
  group_by(year, round, qno) %>%
  summarize(correct_ans = sum(ans), times_asked = n()) %>%
  ggplot(mapping = aes(x = qno, y = correct_ans/times_asked)) +
  geom_col() +
  facet_grid(round ~ year) +
  scale_x_continuous(breaks = seq(1, 12)) +
  scale_y_continuous(limits = c(0, 1), labels = label_percent()) +
  labs(title = "Percent of questions correctly answered in Championship",
       subtitle = "Cheaters only",
       x = "Question Number",
       y = "% Correctly Answered")
```

```
## Joining, by = c("year", "name", "honorsystemcorrect")
```


Percent of questions correctly answered in Championship

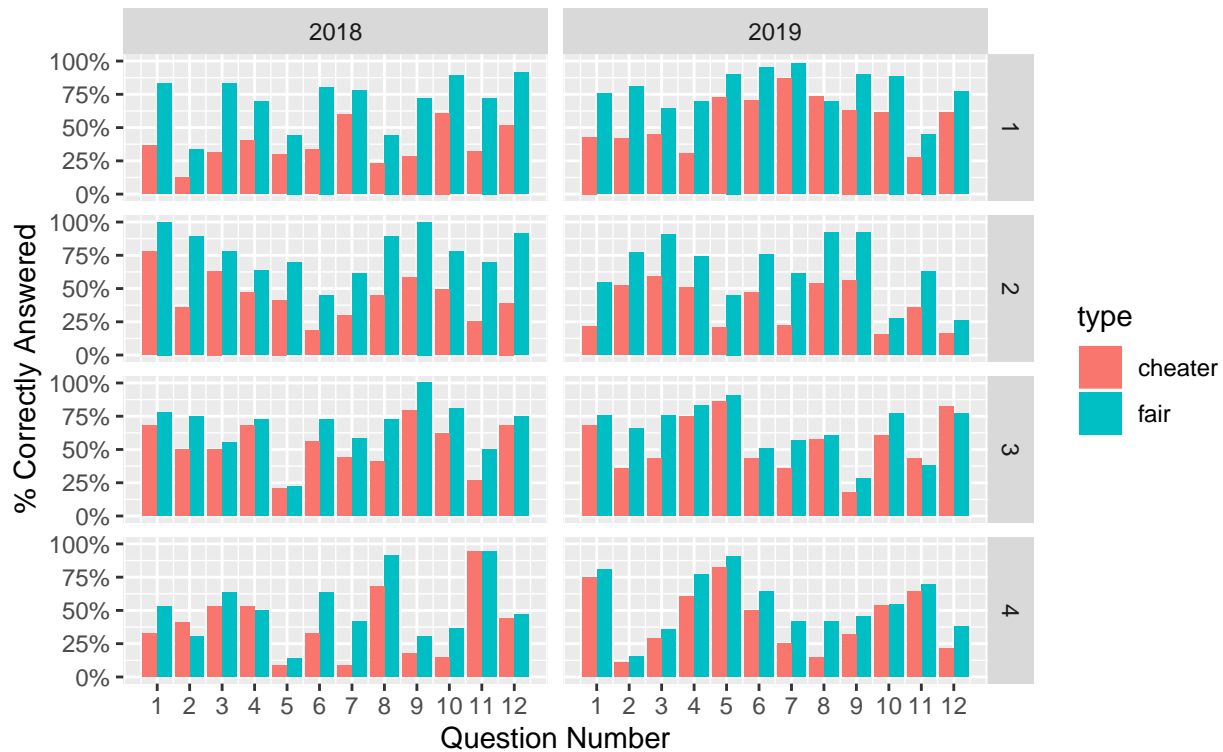
Cheaters only



```
bind_rows(cheaters, fairplayers) %>%
  inner_join(edited) %>%
  group_by(year, round, qno, type) %>%
  summarize(correct_ans = sum(ans), times_asked = n()) %>%
  ggplot(aes(x = qno, y = correct_ans/times_asked, group = type, fill = type)) +
  geom_col(position = "dodge") +
  facet_grid(round ~ year) +
  scale_x_continuous(breaks = seq(1, 12)) +
  scale_y_continuous(limits = c(0, 1), labels = label_percent()) +
  labs(title = "Percent of questions correctly answered in Championship",
       subtitle = "Both",
       x = "Question Number",
       y = "% Correctly Answered")
```

```
## Joining, by = c("year", "name", "honorsystemcorrect")
```

Percent of questions correctly answered in Championship Both



```
cheaters %>%
  mutate(honorsystemcorrect = percent(honorsystemcorrect, accuracy = 0.1),
         pct_rt = percent(pct_rt, accuracy = 0.1)) %>%
  slice(1:10) %>%
```

```
## # A tibble: 185 x 6
## # Groups:   year, name [185]
##   year name             honorsystemcorrect pct_rt pvals      type
##   <dbl> <chr>             <chr>          <chr> <list>    <chr>
## 1 2018 Abou-SayedT      85.3%          25.0% <dbl [1]> cheater
## 2 2018 BawdonG          82.8%          29.2% <dbl [1]> cheater
## 3 2018 BerrettJ         77.6%          45.8% <dbl [1]> cheater
## 4 2018 ButschekAHeyHey 81.6%          37.5% <dbl [1]> cheater
## 5 2018 BuxtonK          78.6%          45.8% <dbl [1]> cheater
## 6 2018 CalcagnoR        75.9%          20.8% <dbl [1]> cheater
## 7 2018 CannonS          80.2%          54.2% <dbl [1]> cheater
## 8 2018 CareyC           76.9%          50.0% <dbl [1]> cheater
## 9 2018 ChiltonC2        75.8%          54.2% <dbl [1]> cheater
## 10 2018 ChinG            80.0%          33.3% <dbl [1]> cheater
## # ... with 175 more rows
```

```
# kable(col.names = c("Year", "Name", "Percent Correct, Honor System", "Percent Correct, Competition",
```

Step 2: how difficult the championship questions are relative to the regular season questions for the above group of people

One fairly awful strategy is to calculate mean correctness percentage of all eligible participants at home, and then compare that to the mean correctness percentage of all eligible participants in competition. By eligible,

I mean participants that pass the tests I define in step 1.

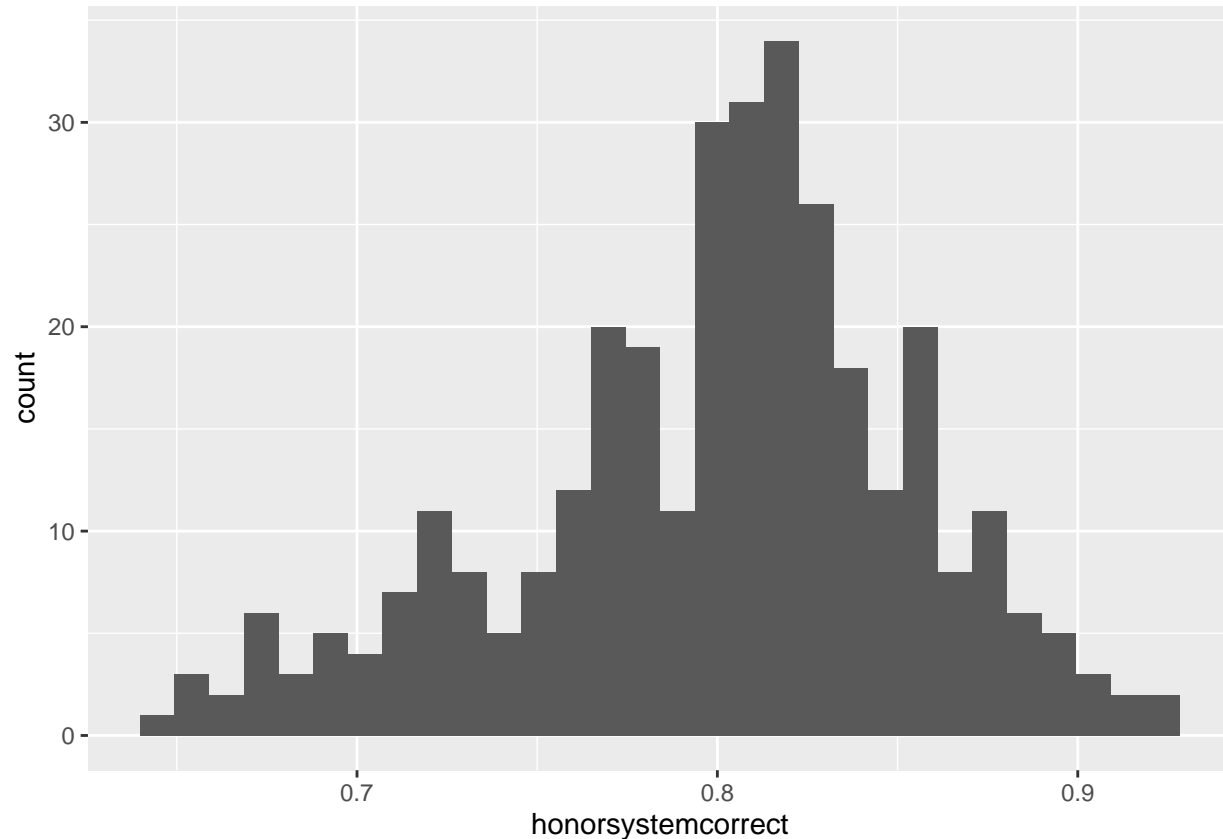
One better strategy is to

```
# Lets find people who I think played honestly during the regular season.  
edited %>%
```

```
  distinct(name, honorsystemcorrect) %>%  
  ggplot(aes(honorsystemcorrect)) +  
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 19 rows containing non-finite values (stat_bin).
```



```
edited %>%  
  group_by(name, year, round) %>%  
  mutate(stdev = sd(ans), competcorrect = mean(ans)) %>%  
  mutate(tstatistic = (competcorrect - honorsystemcorrect)/(stdev/sqrt(12)))
```

```
## # A tibble: 12,132 x 11
```

```
## # Groups:   name, year, round [1,011]
```

```
##   name   year round merge numbercorrect honorsystemcorr~  qno  ans stdev  
##   <chr> <dbl> <dbl> <chr>          <dbl>          <dbl> <dbl> <dbl> <dbl>  
## 1 Abou~ 2018     2 merg~              2            0.853     1     0 0.389  
## 2 Abou~ 2018     2 merg~              2            0.853     2     0 0.389  
## 3 Abou~ 2018     2 merg~              2            0.853     3     0 0.389  
## 4 Abou~ 2018     2 merg~              2            0.853     4     1 0.389  
## 5 Abou~ 2018     2 merg~              2            0.853     5     0 0.389  
## 6 Abou~ 2018     2 merg~              2            0.853     6     0 0.389
```

```
## 7 Abou~ 2018 2 merg~ 2 0.853 7 1 0.389
## 8 Abou~ 2018 2 merg~ 2 0.853 8 0 0.389
## 9 Abou~ 2018 2 merg~ 2 0.853 9 0 0.389
## 10 Abou~ 2018 2 merg~ 2 0.853 10 0 0.389
## # ... with 12,122 more rows, and 2 more variables: competcorrect <dbl>,
## # tst statistic <dbl>
```

4. Explain why it is easier or harder to make claims about the aggregate amount of cheating in a sample versus identifying individual cheaters.
5. The players with -99 for honor code scores dropped out of the league after making one or both championships. Can you make any inferences about whether they cheated more or less than the players who have remained in the league, despite the fact you know nothing about their percent correct in the regular season?

I've identified the characteristics of what a cheater looks like, and I think all these people fit the bill based on one axis. Not a single person who stayed who had the same championship scores as the -99ers was categorised as a fair player. Thus, I think they are cheaters.

```
dropouts <- edited %>%
  filter(is.na(honorsystemcorrect)) %>%
  distinct(name, year) %>%
  pull(name)

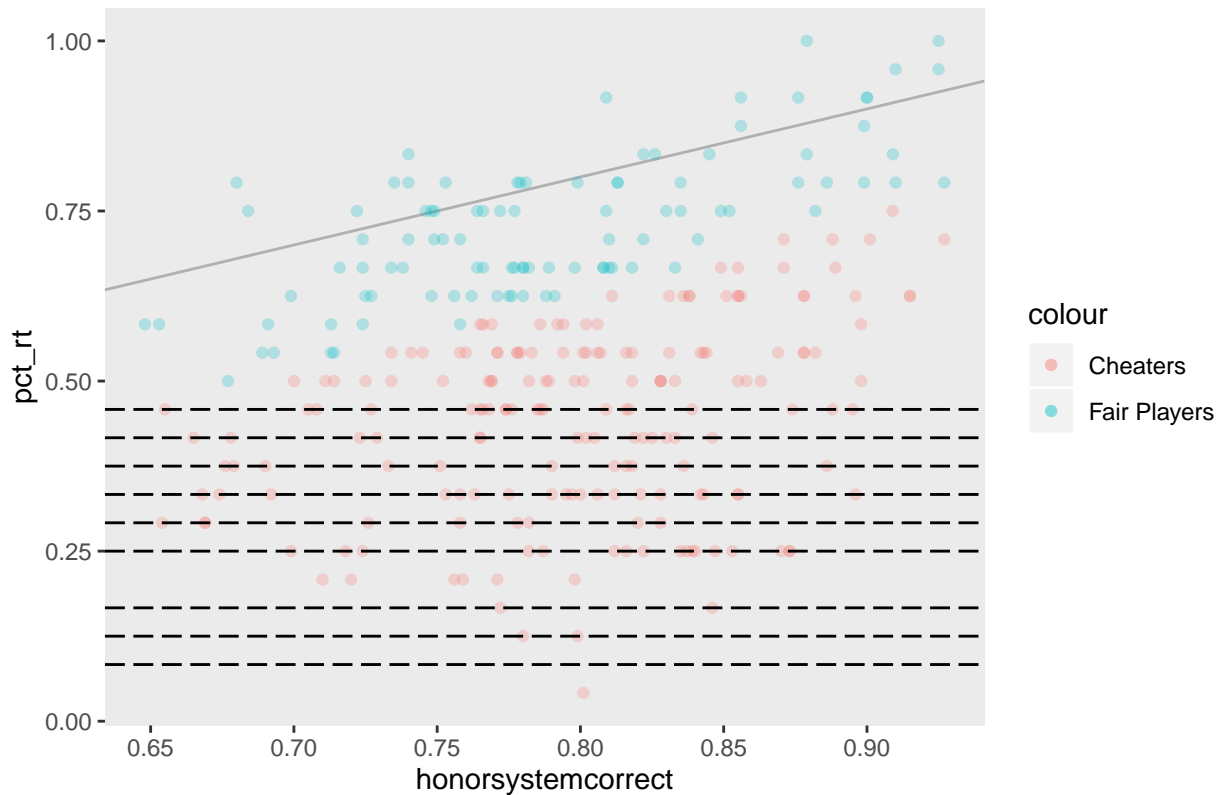
x <- edited %>%
  filter(name %in% dropouts) %>%
  group_by(name, year) %>%
  summarize(pct_corr = mean(ans)) %>%
  ungroup() %>%
  count(pct_corr)

ggplot() +
  geom_abline(slope = 1, alpha = 0.25) +
  geom_point(cheaters, mapping = aes(honorsystemcorrect, pct_rt, color = "Cheaters"), alpha = 0.25) +
  geom_point(fairplayers, mapping = aes(honorsystemcorrect, pct_rt, color = "Fair Players"), alpha = 0.25) +
  geom_hline(x, mapping = aes(yintercept = pct_corr), linetype = "longdash") +
  labs(title = "Percent Correct at Home vs Championship by Player Type") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
```

Table 3: 2018/2019 Championship Percentage Correct by Question

Round Number	2018				2019			
	1	2	3	4	1	2	3	4
Sample Size (n =)	1668	1668	840	840	1968	1968	984	984
Question 1	45.32%	79.86%	72.9%	42.9%	55.49%	33.54%	72.0%	79.3%
Question 2	16.55%	48.20%	62.9%	35.7%	56.10%	60.98%	54.9%	13.4%
Question 3	43.17%	62.59%	52.9%	58.6%	52.44%	70.73%	63.4%	32.9%
Question 4	45.32%	50.36%	70.0%	51.4%	45.12%	60.37%	80.5%	70.7%
Question 5	32.37%	48.20%	21.4%	11.4%	78.05%	29.27%	87.8%	86.6%
Question 6	44.60%	24.46%	64.3%	48.6%	78.66%	57.32%	48.8%	58.5%
Question 7	60.43%	37.41%	51.4%	25.7%	90.85%	36.59%	48.8%	35.4%
Question 8	28.06%	53.96%	57.1%	80.0%	72.56%	67.68%	58.5%	31.7%
Question 9	38.13%	65.47%	90.0%	24.3%	73.17%	70.12%	24.4%	40.2%
Question 10	64.03%	57.55%	71.4%	25.7%	70.73%	20.12%	70.7%	53.7%
Question 11	38.85%	33.81%	38.6%	94.3%	33.54%	46.95%	39.0%	67.1%
Question 12	58.99%	49.64%	71.4%	45.7%	67.07%	20.73%	79.3%	32.9%

Percent Correct at Home vs Championship by Player Type



6. the person who runs this league is interested in learning your findings. Create one visual that you think best would summarize your insights showing the amount/non-existence of cheating in his league.

Table 4: Percentages of Honor System Success by Round and Year

Year	1	2	3	4
2018	79.7% (n = 1668)	79.7% (n = 1668)	80.7% (n = 840)	80.7% (n = 840)
2019	79.5% (n = 1968)	79.5% (n = 1968)	80.7% (n = 984)	80.7% (n = 984)

Appendices

Percent of questions correctly answered in Championship

