Homework 4

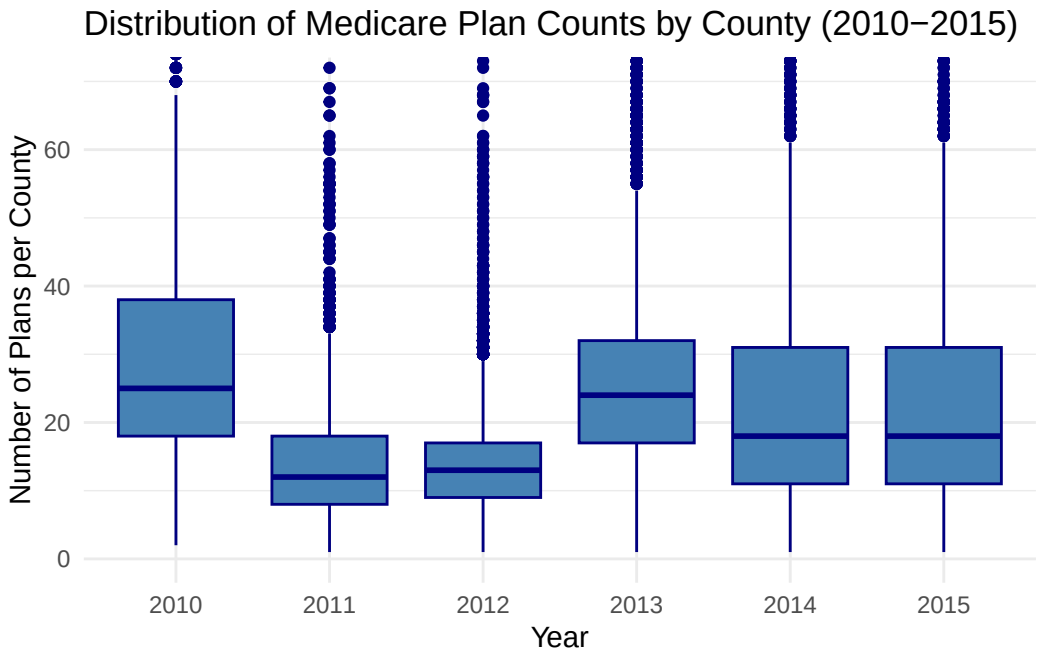
ECON 470, Spring 2025

Sushmita Rajan

Here is a link to my repository: {https://github.com/sarajan03/econ470spring2025/tree/main/Homework4} emove all SNPs, 800-series plans, and prescription drug only plans (i.e., plans that do not offer Part C benefits). Provide a box and whisker plot showing the distribution of plan counts by county over time. Do you think that the number of plans is sufficient, too few, or too many?

```
# Step 1: Filter data for relevant years and valid partc_score
final_data_clean <- final_data %>%
  filter(year %in% 2010:2015, !is.na(partc_score))

# Step 2: Exclude unwanted plans
# - Remove SNPs
# - Remove 800-series plans (typically employer/union-only)
# - Remove standalone drug plans (PDPs)
final_data_clean <- final_data_clean %>%
  filter(
    snp == "No",
    !(planid >= 800 & planid < 900),
    !(partd == "Y" & plan_type == "PDP")
  )

# Step 3: Count number of plans per county per year
county_plan_counts <- final_data_clean %>%
  group_by(year, state, county) %>%
  summarise(plan_count = n(), .groups = "drop")

# Step 4: Create a boxplot to show distribution of plan counts over time
question1 <- ggplot(county_plan_counts, aes(x = factor(year), y = plan_count)) +
  geom_boxplot(fill = "steelblue", color = "navy") +
  coord_cartesian(ylim = c(0, 70)) +
  labs(
```

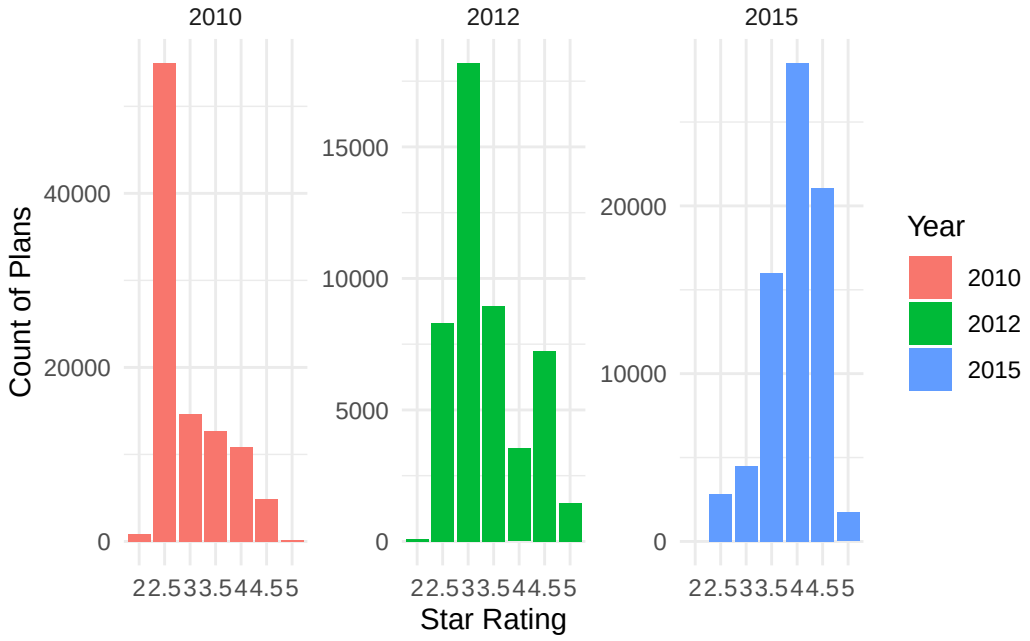
```
    title = "Distribution of Medicare Plan Counts by County (2010-2015)",
    x = "Year",
    y = "Number of Plans per County"
  ) +
  theme_minimal()

# Step 5: Display the plot
  print(question1)
```



Provide bar graphs showing the distribution of star ratings in 2010, 2012, and 2015. How has this distribution changed over time?

```
# Drop NA values before processing
star_dist <- final_data_clean %>%
  filter(year %in% c(2010, 2012, 2015)) %>%
  drop_na(year, Star_Rating) %>%  # Drop rows where year or Star_Rating is NA
  group_by(year, Star_Rating) %>%
  summarise(count = n(), .groups = "drop")

# Create the plot and separate by year
question2 <- ggplot(star_dist, aes(x = as.factor(Star_Rating), y = count, fill = as.factor(ye
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Star Rating", y = "Count of Plans", fill = "Year") +
```

```
  theme_minimal() +
  facet_wrap(~ year, scales = "free_y")  # Separate by year with independent y-axis scales

# Print the plot
print(question2)
```



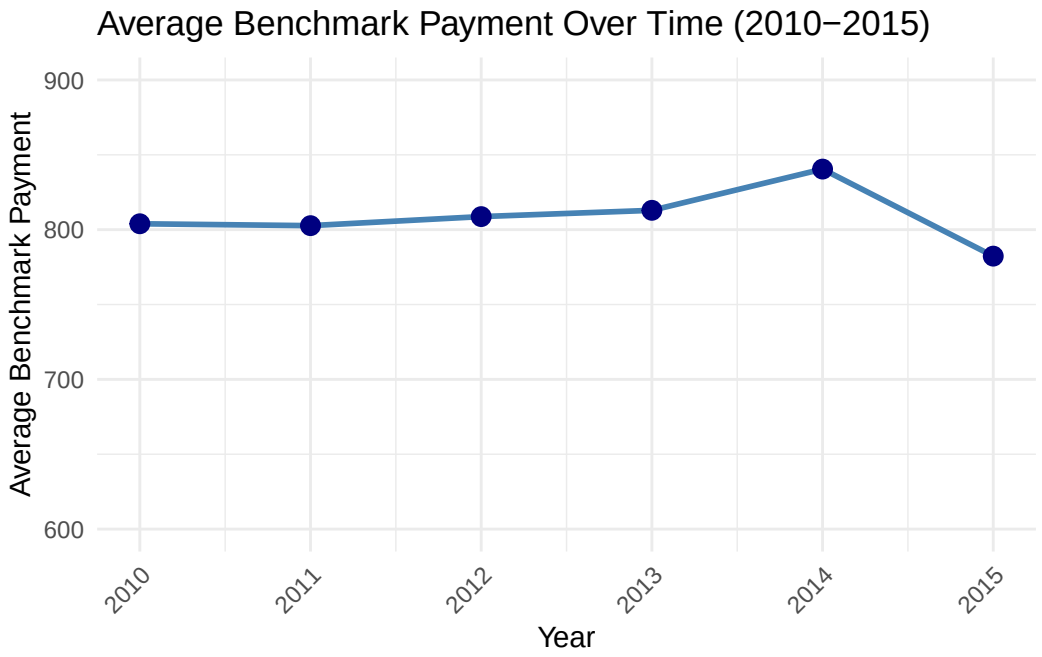
Plot the average benchmark payment over time from 2010 through 2015. How much has the average benchmark payment risen over the years?

```
# Step 1: Calculate the average benchmark payment
avg_benchmark <- final_data %>%
  filter(year >= 2010 & year <= 2015) %>%
  group_by(year) %>%
  summarise(average_benchmark = mean(ma_rate, na.rm = TRUE))

# Step 2: Plot the average benchmark payment over time (2010-2015)
question3 <-ggplot(avg_benchmark, aes(x = year, y = average_benchmark)) +
  geom_line(color = "steelblue", size = 1) +
  geom_point(color = "navy", size = 3) +
  labs(title = "Average Benchmark Payment Over Time (2010-2015)",
       x = "Year",
       y = "Average Benchmark Payment") +
coord_cartesian(ylim = c(600, 900)) +                  # Expand y-axis range without clipping
```

```
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotate x-axis labels
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```

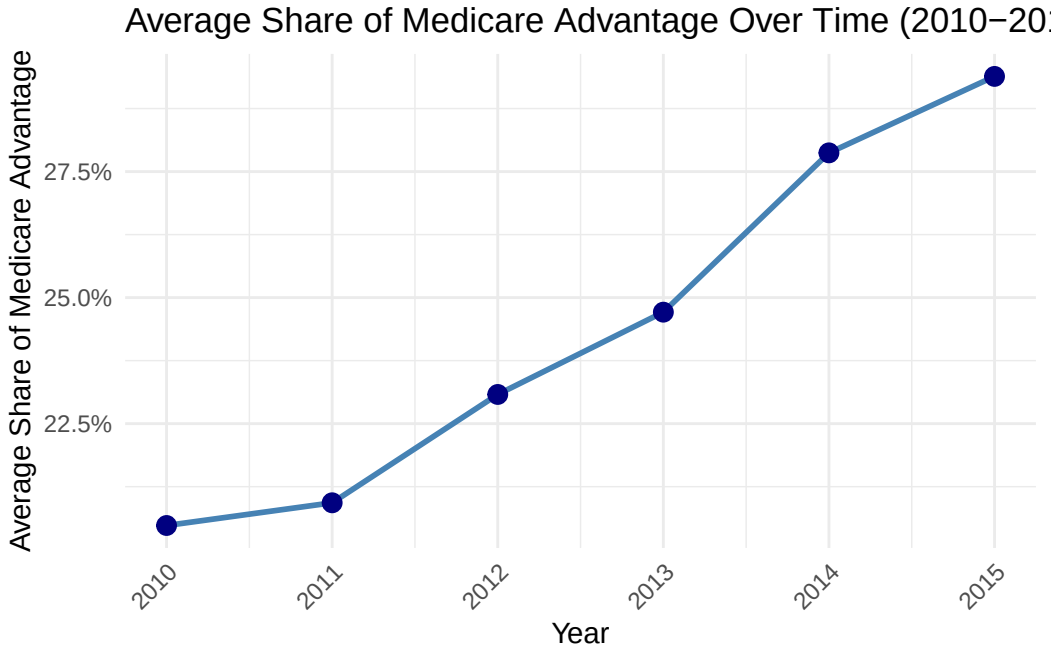
```
print(question3)
```



Plot the average share of Medicare Advantage (relative to all Medicare eligibles) over time from 2010 through 2015. Has Medicare Advantage increased or decreased in popularity? How does this share correlate with benchmark payments?

```
# Step 1: Calculate the Share of Medicare Advantage Enrollment relative to Total Medicare El
medicare_share <- final_data %>%
  filter(year >= 2010 & year <= 2015) %>%
  filter(avg_eligibles > 0, avg_enrolled >= 0) %>%
  group_by(year) %>%
  mutate(ma_share = avg_enrolled / avg_eligibles)  # Share of Medicare Advantage

# Step 2: Calculate the Average Share per Year
avg_share_per_year <- medicare_share %>%
  group_by(year) %>%
```

```
  summarise(average_ma_share = mean(ma_share, na.rm = TRUE))  # Average share for each year

# Step 3: Plot the Trend of Medicare Advantage Share Over Time (2010-2015)
question4<-ggplot(avg_share_per_year, aes(x = year, y = average_ma_share)) +
  geom_line(color = "steelblue", size = 1) +    # Line plot to show the trend
  geom_point(color = "navy", size = 3) +
  labs(title = "Average Share of Medicare Advantage Over Time (2010-2015)",
       x = "Year",
       y = "Average Share of Medicare Advantage") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) + # Rotate x-axis
  scale_y_continuous(labels = label_percent(scale = 100))

print(question4)
```




```
avg_benchmark <- final_data %>%
  filter(year >= 2010 & year <= 2015, !is.na(mean_risk)) %>%
  group_by(year) %>%
  summarise(average_benchmark = mean(mean_risk, na.rm = TRUE))

# Join avg_share_per_year and avg_benchmark by year
combined_data <- avg_share_per_year %>%
```

```
  inner_join(avg_benchmark, by = "year")

# Check structure (optional, to confirm numeric types)
str(combined_data)
```

```
tibble [6 x 3] (S3: tbl_df/tbl/data.frame)
 $ year              : num [1:6] 2010 2011 2012 2013 2014 ...
 $ average_ma_share  : num [1:6] 0.205 0.209 0.231 0.247 0.279 ...
 $ average_benchmark: num [1:6] 0.968 0.97 0.972 0.972 0.973 ...
```

```
# Now calculate correlation
cor_value <- cor(combined_data$average_ma_share, combined_data$average_benchmark)

# Print nicely
print(paste("Correlation between MA Share and Benchmark Payments:", round(cor_value, 3)))
```

```
[1] "Correlation between MA Share and Benchmark Payments: 0.94"
```

Estimate ATEs For the rest of the assignment, we'll use a regression discontinuity design to estimate the average treatment effect from receiving a marginally higher rating. We'll focus only on 2010.

Calculate the running variable underlying the star rating. Provide a table showing the number of plans that are rounded up into a 3-star, 3.5-star, 4-star, 4.5-star, and 5-star rating.

```
data_2010 <- final_data %>%
  filter(year == 2010)  # Filter for year 2010

# Step 1: Calculate the raw rating and assign it to a new column
raw_2010 <- data_2010 %>%
  mutate(raw_rating = rowMeans(
    cbind(
      breastcancer_screen, rectalcancer_screen, cv_diab_cholscreen, glaucoma_test,
      monitoring, flu_vaccine, pn_vaccine, physical_health, mental_health, osteo_test,
      physical_monitor, primaryaccess, osteo_manage, diab_healthy, bloodpressure, ra_manage,
      copd_test, bladder, falling, nodelays, doctor_communicate, carequickly, customer_servi
      overallrating_care, overallrating_plan, complaints_plan, appeals_timely, appeals_revie
      leave_plan, audit_problems, hold_times, info_accuracy, ttyt_available),
    na.rm = TRUE)) %>%
  select(contractid, planid, fips, avg_enrollment, state, county, raw_rating, partc_score,
         avg_eligibles, avg_enrolled, premium_partc, risk_ab, Star_Rating,
```

```
          bid, avg_ffscost, ma_rate, plan_type) %>%
    mutate(mkt_share = avg_enrollment/avg_eligibles,
           HMO=str_detect(plan_type, "HMO"))

# Step 2: Round star ratings based on raw scores and calculate rounded counts
ratings_2010_summary <- raw_2010 %>%
  filter(Star_Rating %in% c(3, 3.5, 4, 4.5, 5)) %>%  # Focus on specific star ratings
  mutate(
    round_to_3 = as.integer(raw_rating >= 2.75 & raw_rating < 3.00 & Star_Rating == 3),
    round_to_35 = as.integer(raw_rating >= 3.25 & raw_rating < 3.50 & Star_Rating == 3.5),
    round_to_4 = as.integer(raw_rating >= 3.75 & raw_rating < 4.00 & Star_Rating == 4),
    round_to_45 = as.integer(raw_rating >= 4.25 & raw_rating < 4.50 & Star_Rating == 4.5),
    round_to_5 = as.integer(raw_rating >= 4.75 & raw_rating < 5.00 & Star_Rating == 5)
  ) %>%
  group_by(Star_Rating) %>%
  summarise(
    count_round_3 = sum(round_to_3),
    count_round_35 = sum(round_to_35),
    count_round_4 = sum(round_to_4),
    count_round_45 = sum(round_to_45),
    count_round_5 = sum(round_to_5),
    .groups = "drop"
  ) %>%
  mutate(
    total_rounded_up = count_round_3 + count_round_35 + count_round_4 + count_round_45 + cou
  ) %>%
  select(Star_Rating, total_rounded_up)

# Display the results in a table format
kable(ratings_2010_summary, caption = "Summary of Rounded Ratings by Star Rating")
```

Table 1: Summary of Rounded Ratings by Star Rating

Star_Rating	total_rounded_up
3.0	6132
3.5	6938
4.0	7216
4.5	1506
5.0	60

Using the RD estimator with a bandwidth of 0.125, provide an estimate of the effect of receiving

a 3-star versus a 2.5 star rating on enrollments. Repeat the exercise to estimate the effects at 3.5 stars, and summarize your results in a table.

```
# Estimate effect of receiving 3 stars vs 2.5
rd_effect_3_vs_2_5 <- lm(mkt_share ~ treat_3 + score_diff,
                         data = raw_2010 %>%
                           filter(raw_rating >= (2.75 - 0.125),
                                  raw_rating <= (2.75 + 0.125),
                                  Star_Rating %in% c(2.5, 3.0)) %>%
                           mutate(
                             treat_3 = Star_Rating == 3.0,
                             score_diff = raw_rating - 2.75
                           ))

# Estimate effect of receiving 3.5 stars vs 3
rd_effect_3_5_vs_3 <- lm(mkt_share ~ treat_3_5 + score_diff,
                         data = raw_2010 %>%
                           filter(raw_rating >= (3.25 - 0.125),
                                  raw_rating <= (3.25 + 0.125),
                                  Star_Rating %in% c(3.0, 3.5)) %>%
                           mutate(
                             treat_3_5 = Star_Rating == 3.5,
                             score_diff = raw_rating - 3.25
                           ))

# Extract and rename tidy outputs
tidy_3 <- tidy(rd_effect_3_vs_2_5) %>%
  select(term, estimate, std.error) %>%
  rename(Estimate_3 = estimate, SE_3 = std.error)

tidy_3_5 <- tidy(rd_effect_3_5_vs_3) %>%
  select(term, estimate, std.error) %>%
  rename(Estimate_3_5 = estimate, SE_3_5 = std.error)

# Combine and format table
results_table <- full_join(tidy_3, tidy_3_5, by = "term") %>%
  mutate(
    Estimate_3 = sprintf("%.4f", Estimate_3),
    SE_3 = sprintf("(%.4f)", SE_3),
    Estimate_3_5 = sprintf("%.4f", Estimate_3_5),
    SE_3_5 = sprintf("(%.4f)", SE_3_5)
  ) %>%
  select(term, Estimate_3, SE_3, Estimate_3_5, SE_3_5)
```

```r
# Display table
kable(results_table,
      col.names = c("Variable", "3 Star", "", "3.5 Star", ""),
      caption = "Table 6: RD Estimates at 3.0 and 3.5 Star Cutoffs",
      align = "lcccc")
```

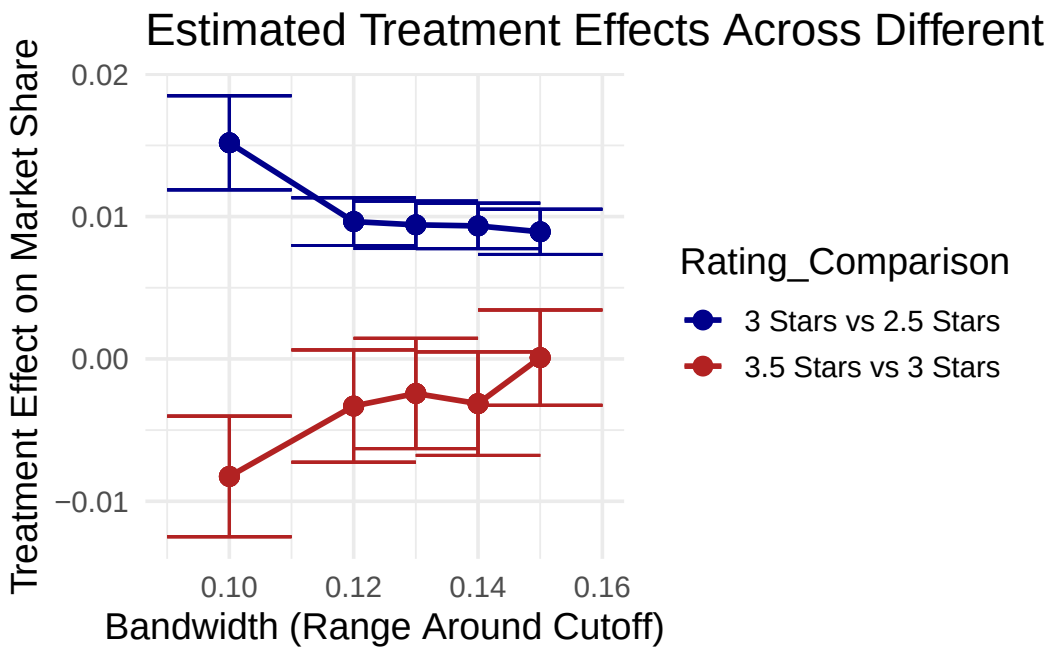
Table 2: Table 6: RD Estimates at 3.0 and 3.5 Star Cutoffs

Variable	3 Star		3.5 Star	
(Intercept)	0.0085	(0.0004)	0.0167	(0.0012)
treat_3TRUE	0.0095	(0.0009)	NA	(NA)
score_diff	-0.0236	(0.0048)	0.0224	(0.0116)
treat_3_5TRUE	NA	(NA)	-0.0027	(0.0019)

Repeat your results for bandwidhts of 0.1, 0.12, 0.13, 0.14, and 0.15 (again for 3 and 3.5 stars). Show all of the results in a graph. How sensitive are your findings to the choice of bandwidth?

```r
# Define new bandwidth values, rating cutoffs, and comparison labels
bw_values <- c(0.10, 0.12, 0.13, 0.14, 0.15)
cutoffs_list <- c(2.75, 3.25)
comparison_labels <- c("3 Stars vs 2.5 Stars", "3.5 Stars vs 3 Stars")

# Function to execute regression discontinuity analysis
run_rd_estimate <- function(dataset, cutoff, bandwidth, comparison_label) {
  lower_limit <- cutoff - bandwidth
  upper_limit <- cutoff + bandwidth

  treatment_group <- ifelse(cutoff == 2.75, 3.0, 3.5)
  control_group <- ifelse(cutoff == 2.75, 2.5, 3.0)

  filtered_data <- dataset %>%
    filter(raw_rating >= lower_limit, raw_rating <= upper_limit,
           Star_Rating %in% c(control_group, treatment_group)) %>%
    mutate(treatment_indicator = Star_Rating == treatment_group,
           score_diff = raw_rating - cutoff)

  # Fit the model and extract results
  model_fit <- lm(mkt_share ~ treatment_indicator + score_diff, data = filtered_data)
  model_summary <- tidy(model_fit)
```

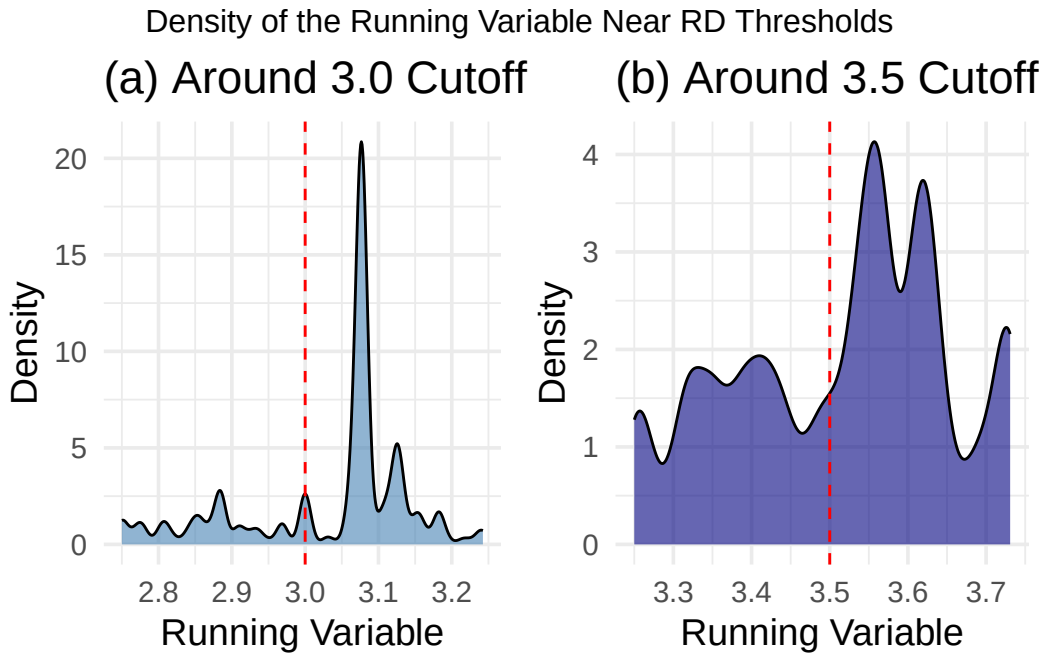
```
  model_glance <- glance(model_fit)

  tibble(
    Rating_Comparison = comparison_label,
    Bandwidth = bandwidth,
    Estimated_Effect = model_summary$estimate[model_summary$term == "treatment_indicatorTRUE"
    SE_Estimate = model_summary$std.error[model_summary$term == "treatment_indicatorTRUE"],
    Observations = model_glance$nobs,
    R_Squared = model_glance$r.squared
  )
}

# Apply the analysis across all combinations of bandwidths and cutoffs
result_data <- purrr::map2_dfr(
  rep(cutoffs_list, each = length(bw_values)),
  rep(comparison_labels, each = length(bw_values)),
  ~ {
    purrr::map_dfr(bw_values, function(bw) {
      run_rd_estimate(raw_2010, .x, bw, .y)
    })
  }
)

# Plot the treatment effects based on bandwidth
effect_plot <- ggplot(result_data, aes(x = Bandwidth, y = Estimated_Effect, color = Rating_C
  geom_line(linewidth = 1) +
  geom_point(size = 3) +
  geom_errorbar(aes(ymin = Estimated_Effect - 1.96 * SE_Estimate,
                    ymax = Estimated_Effect + 1.96 * SE_Estimate), width = 0.02) +
  labs(
    title = "Estimated Treatment Effects Across Different Bandwidths",
    x = "Bandwidth (Range Around Cutoff)",
    y = "Treatment Effect on Market Share"
  ) +
  theme_minimal(base_size = 14) +
  scale_color_manual(values = c("3 Stars vs 2.5 Stars" = "darkblue", "3.5 Stars vs 3 Stars" =

# Display the plot
print(effect_plot)
```



Examine (graphically) whether contracts appear to manipulate the running variable. In other words, look at the distribution of the running variable before and after the relevent threshold values. What do you find?

```
# Subset data around each cutoff
cutoff_3 <- raw_2010 %>% filter(raw_rating >= 2.75 & raw_rating < 3.25)
cutoff_35 <- raw_2010 %>% filter(raw_rating >= 3.25 & raw_rating < 3.75)

# Plot around 3.0 cutoff
plot_3 <- ggplot(cutoff_3, aes(x = raw_rating)) +
  geom_density(fill = "steelblue", alpha = 0.6) +
  geom_vline(xintercept = 3.0, linetype = "dashed", color = "red") +
  labs(title = "(a) Around 3.0 Cutoff", x = "Running Variable", y = "Density") +
  theme_minimal(base_size = 14)

# Plot around 3.5 cutoff
plot_35 <- ggplot(cutoff_35, aes(x = raw_rating)) +
  geom_density(fill = "navy", alpha = 0.6) +
  geom_vline(xintercept = 3.5, linetype = "dashed", color = "red") +
  labs(title = "(b) Around 3.5 Cutoff", x = "Running Variable", y = "Density") +
  theme_minimal(base_size = 14)

# Combine the plots side-by-side
```

```
grid.arrange(plot_3, plot_35, ncol = 2, top = "Density of the Running Variable Near RD Thresh
```



Similar to question 4, examine whether plans just above the threshold values have different characteristics than contracts just below the threshold values. Use HMO and Part D status as your plan characteristics. Summarize your findings from 5-9. What is the effect of increasing a star rating on enrollments? Briefly explain your results.