ABtest_Project

Yi Chen Chung

Data Check

```
df[!complete.cases(df),]
```

```
## # A tibble: 0 × 22
## # ℹ 22 variables: Variant <chr>, loanofficer_id <chr>, day <dbl>,
## #   typeI_init <dbl>, typeI_fin <dbl>, typeII_init <dbl>, typeII_fin <dbl>,
## #   agree_init <dbl>, agree_fin <dbl>, conflict_init <dbl>, conflict_fin <dbl>,
## #   revised_per_ai <dbl>, revised_agst_ai <dbl>, fully_complt <dbl>,
## #   confidence_init_total <dbl>, confidence_fin_total <dbl>, complt_init <dbl>,
## #   complt_fin <dbl>, ai_typeI <dbl>, ai_typeII <dbl>, badloans_num <dbl>,
## #   goodloans_num <dbl>
```

There is no row containing an NA value in the dataset; however, further check is needed to see if there is any inappropriate data. Some of them have a final complete case equal to 0.

```
sum(duplicated(df))
```

```
## [1] 0
```

No duplicated rows in this dataset

```
df |>
group_by(loanofficer_id) %>%
  summarise(num_variants = n_distinct(Variant)) %>%
  filter(num_variants > 1)
```

```
## # A tibble: 0 × 2
## # ℹ 2 variables: loanofficer_id <chr>, num_variants <int>
```

No result showed up, which means that each officer is assigned to only one group.

```
df |>
  group_by(loanofficer_id,Variant) |>
  reframe(Variant,Case=n())
```

```
## # A tibble: 470 × 3
##    loanofficer_id Variant    Case
##    <chr>          <chr>     <int>
##  1 0899qxvc       Treatment    10
##  2 0899qxvc       Treatment    10
##  3 0899qxvc       Treatment    10
##  4 0899qxvc       Treatment    10
##  5 0899qxvc       Treatment    10
##  6 0899qxvc       Treatment    10
##  7 0899qxvc       Treatment    10
##  8 0899qxvc       Treatment    10
##  9 0899qxvc       Treatment    10
## 10 0899qxvc       Treatment    10
## # ℹ 460 more rows
```

```
df |>
  group_by(Variant) |>
  reframe(Case=n())
```

```
## # A tibble: 2 × 2
##   Variant    Case
##   <chr>     <int>
## 1 Control     190
## 2 Treatment   280
```

Every officer has 10 cases, 190 cases in the control group, and 280 cases in the other group. In other words, 19 officers were assigned to the control group and 28 were assigned to the treatment group

```
df |>
  group_by(Variant) |>
  filter(confidence_fin_total==0 & fully_complt==0 & complt_fin==0) |>
  reframe(Number=n())
```

```
## # A tibble: 1 × 2
##   Variant Number
##   <chr>    <int>
## 1 Control     90
```

After looking through the data, it seems like fully complete = min(complt_init,complt_fin). There are 90 data points where complt_fin = 0, and this situation is only in the control group. Complt_fin will be zero, and fully_complt will also equal to 0.

```
df <- df |>
  filter(complt_fin != 0)
df |>
  group_by(Variant) |>
  reframe(Case=n())
```

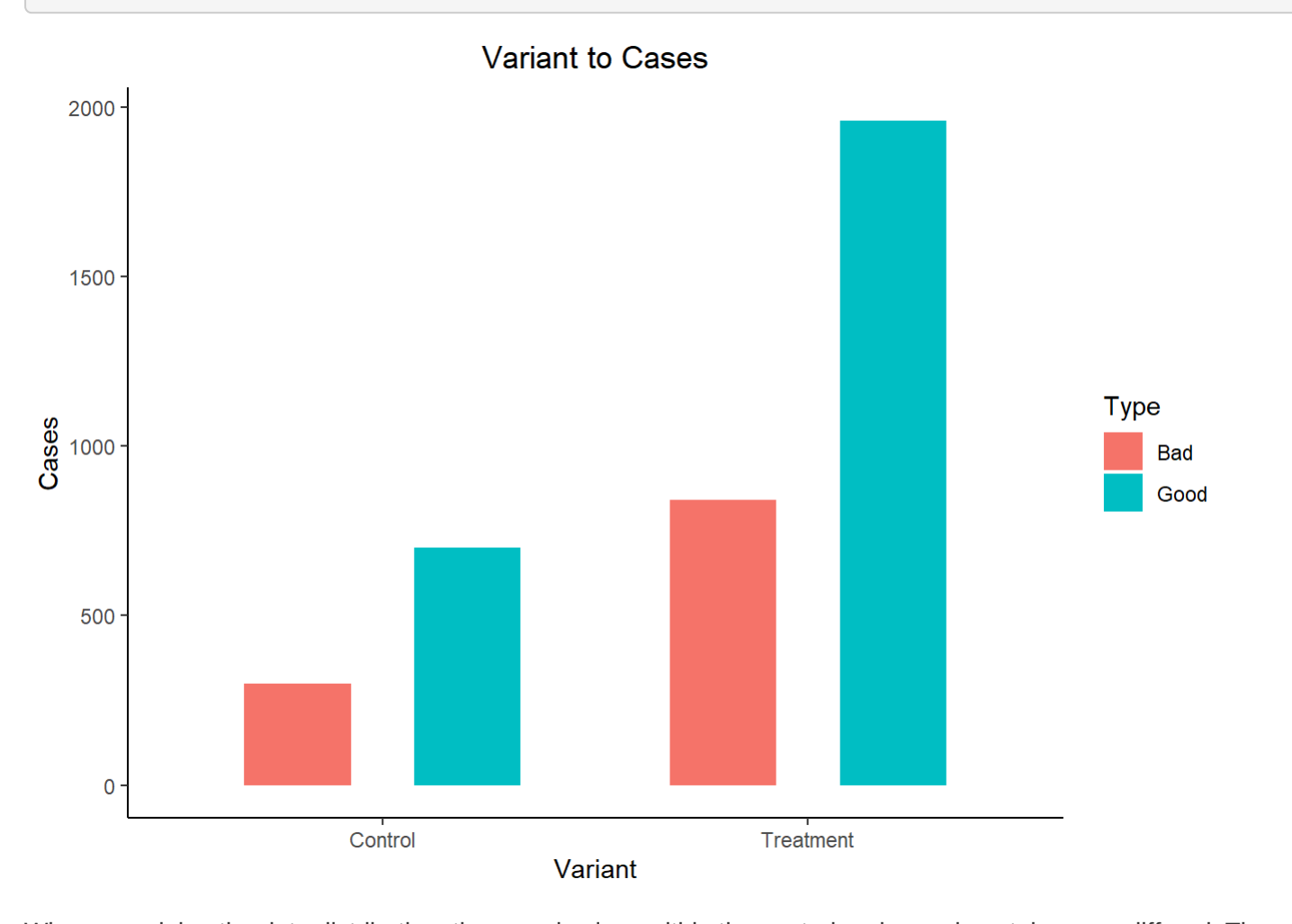
```
## # A tibble: 2 × 2
##   Variant    Case
##   <chr>     <int>
## 1 Control     100
## 2 Treatment   280
```

There seems to be a systematic loss in the control group, so the data points which complt_fin=0 are dropped. 380 data points remained, with 100 in the control group and 280 in the other after cleaning the data. There is an imbalance between the two groups, the low number of control groups will affect the power of the later statistical tests.

EDA

Variant to Cases

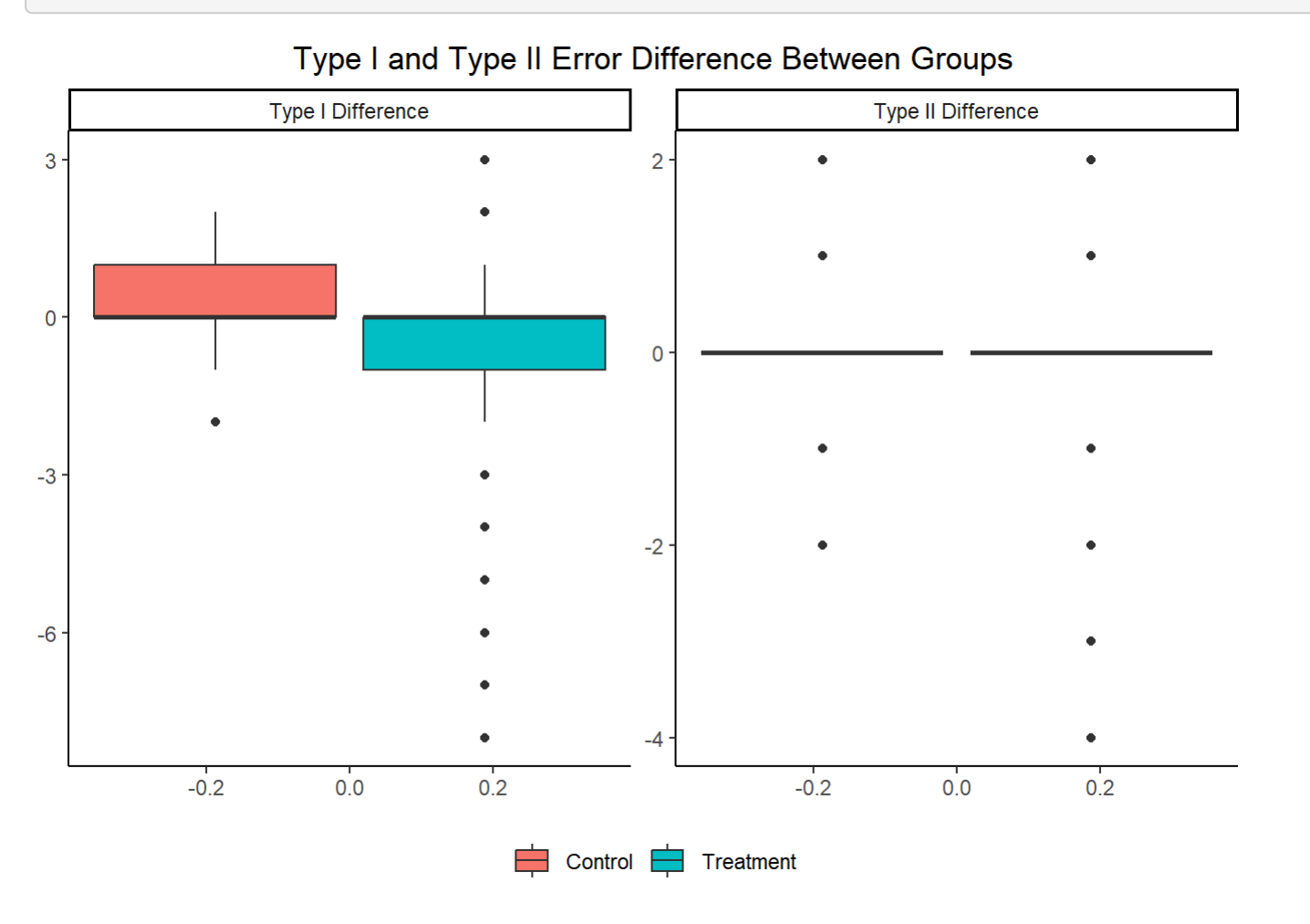
```
df |>
  group_by(Variant) |>
  reframe(Good=sum(goodloans_num),Bad=sum(badloans_num)) |>
  pivot_longer(cols=c(Good,Bad),names_to = "type",values_to = "cases") |>
  ggplot(aes(x=Variant,y=cases,fill=type))+
  geom_col(position=position_dodge(width=0.8),width=0.5)+
  labs(title="Variant to Cases",x="Variant",fill="Type",y="Cases")+
  theme_classic()+
  theme(plot.title= element_text(hjust=0.5))
```



When examining the data distribution, the sample sizes within the control and experimental groups differed. The number of good loans in both groups was higher than that of bad loans, indicating an uneven data distribution. This may indicate an uneven experimental design or random sample allocation, so caution should be exercised when interpreting the model results.

Type I and Type II Error Difference Between Groups

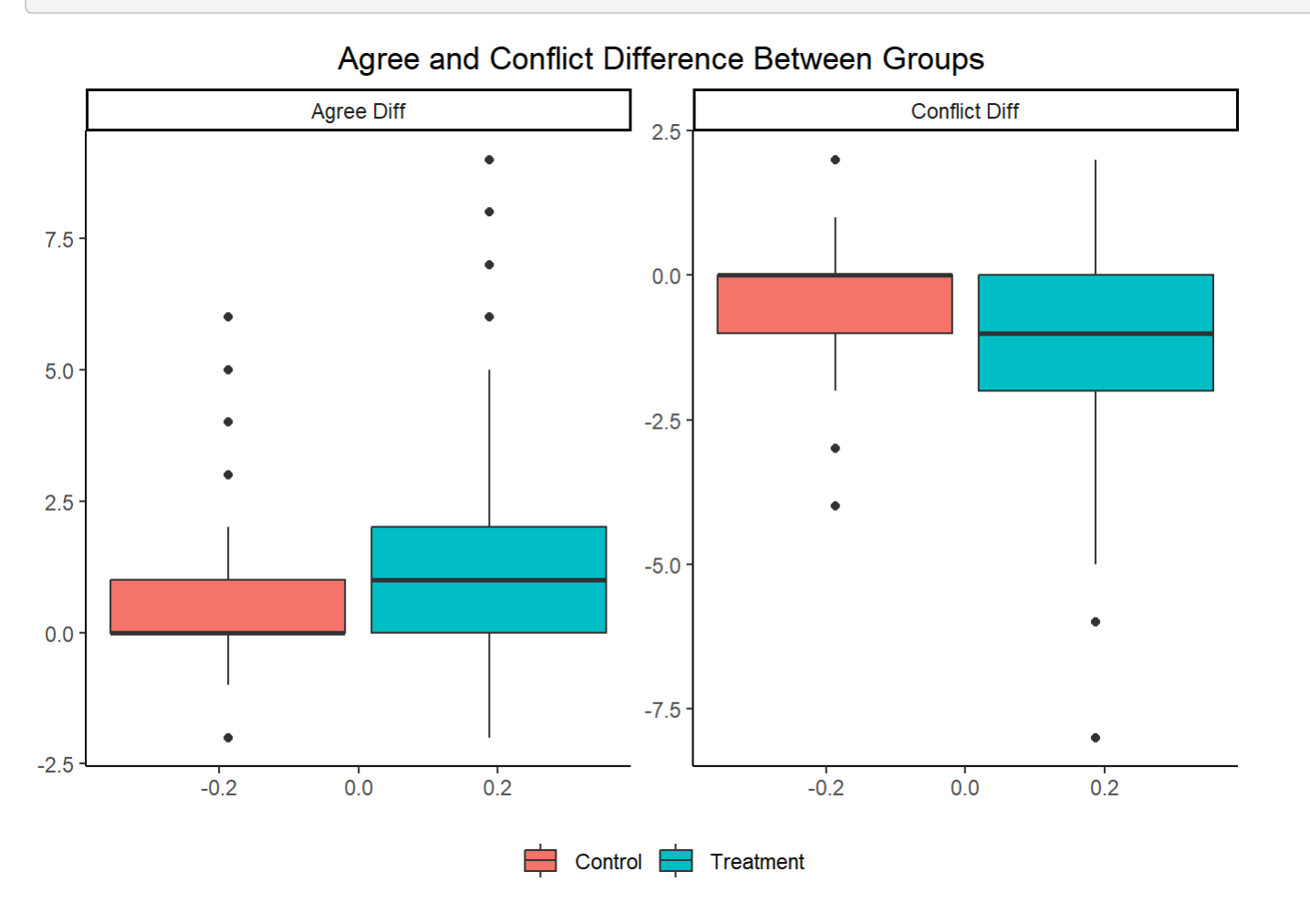
```
df_new <- df |>
  mutate(diff_typeI = typeI_fin-typeI_init,
         diff_typeII = typeII_fin - typeII_init) |>
  select(Variant,day,loanofficer_id,diff_typeI,diff_typeII)
df_long <- df_new |>
  pivot_longer(cols = c('diff_typeI','diff_typeII'),names_to = 'type',values_to = 'values')
df_long$type <- factor(df_long$type, levels = c("diff_typeI", "diff_typeII"),labels=c('Type I Difference','Type I
I Difference'))
df_long |>
  ggplot(aes(y=values,fill=Variant))+geom_boxplot()+
  facet_wrap(.~type,scales='free_y')+theme_classic()+labs(y=NULL,title = 'Type I and Type II Error Difference Bet
ween Groups',fill=NULL)+
  theme(legend.position = 'bottom',
        plot.title = element_text(hjust=0.5))
```



The Treatment group showed a reduction in Type I errors compared to the Control group, suggesting improved ability to avoid approving bad loans. However, there was little to no difference in Type II errors, indicating that the mew model did not materially affect the rejection of good loans.

Agree and Conflict Difference Between Groups

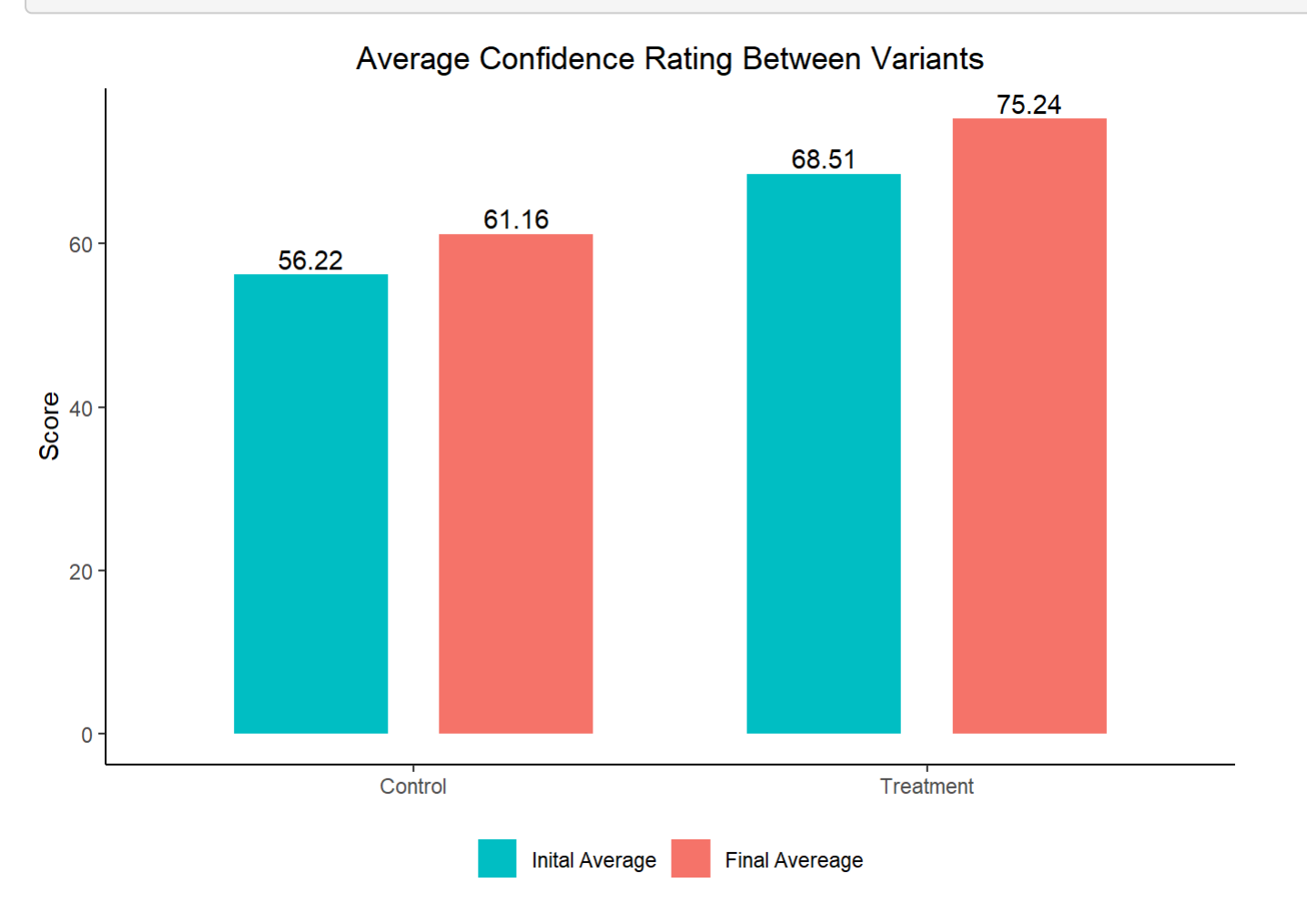
```
df_new <- df |>
  mutate(diff_agree = agree_fin-agree_init,
         diff_conflict = conflict_fin - conflict_init) |>
  select(Variant,day,loanofficer_id,diff_agree,diff_conflict)
df_long <- df_new |>
  pivot_longer(cols = c('diff_agree','diff_conflict'),names_to = 'type',values_to = 'values')
df_long$type <- factor(df_long$type, levels = c("diff_agree", "diff_conflict"),labels=c('Agree Diff','Conflict Di
ff'))
df_long |>
  ggplot(aes(y=values,fill=Variant))+geom_boxplot()+
  facet_wrap(.~type,scales='free_y')+theme_classic()+labs(y=NULL,title = 'Agree and Conflict Difference Between G
roups',fill=NULL)+
  theme(legend.position = 'bottom',
        plot.title = element_text(hjust=0.5))
```



The Treatment group not only increased operators' agreement with the model but also reduced their disagreement. In contrast, the Control group showed little to no change, with most differences centered around zero.

Average Confidence Rating Between Variants

```
df_new <- df |>
  select(Variant,day,loanofficer_id,confidence_init_total,confidence_fin_total,complt_init,complt_fin)
df_new <- df_new |>
  group_by(Variant) |>
  summarise(average_init = sum(confidence_init_total)/sum(complt_init),
            average_fin = sum(confidence_fin_total)/sum(complt_fin))
df_new <- df_new |>
  pivot_longer(cols=c('average_init','average_fin'),names_to = 'type',values_to = 'values')
df_new$type <- factor(df_new$type, levels = c("average_init", "average_fin"),labels=c('Inital Average','Final Ave
reage'))
df_new |>
  ggplot(aes(x=Variant, y=values, fill=type, label=round(values,2))) +
  geom_col(position = position_dodge(width = 0.8), width = 0.6) +
  geom_text(position = position_dodge(width = 0.8), vjust = -0.3) +
  theme_classic() +
  labs(fill = NULL, y = 'Score', x = NULL, title = 'Average Confidence Rating Between Variants') +
  scale_fill_manual(values = c("Inital Average" = "#00BFC4", "Final Avereage" = "#F8766D")) +
  theme(
    plot.title = element_text(hjust = .5),
    legend.position = 'bottom'
  )
```



Both groups showed higher confidence in the model over time, but the Treatment group not only started with a higher baseline score, it also experienced a larger increase (6.7 vs. 4.9). These results provide preliminary support for the notion that the Treatment model might had a positive effect on operators' attitudes.

AI Type I and Type II Error Between Variants

```
df_long <- df |>select(Variant,day,loanofficer_id,ai_typeI,ai_typeII) |>
  pivot_longer(cols=c(ai_typeI,ai_typeII),names_to = 'type',values_to = 'values')
df_long$type <- factor(df_long$type, levels = c("ai_typeI", "ai_typeII"),labels=c('AI Type I Error','AI Type II E
rror'))
df_long |>
  ggplot(aes(y=values,fill=Variant))+geom_boxplot()+ theme_classic()+
  facet_wrap(.~type)+labs(title='AI Type I and Type II Error Between Variants',y=NULL,fill=NULL)+
  scale_fill_manual(values = c("Control" = "#00BFC4", "Treatment" = "#F8766D"))+
  theme(legend.position='bottom',
        plot.title= element_text(hjust=0.5))
```

Compared to Control, the Treatment group achieved lower AI Type I and Type II errors, suggesting the new model improved overall accuracy.

The exploratory analysis suggests that the new model (Treatment) may have a positive impact on operators. However, to assess whether these effects are statistically significant, the next section applies A/B testing for formal evaluation.

A/B Testing

The final OEC (Overall Evaluation Criterion) selected for this test is the difference in recall and precision rates between the two groups before and after using the model. Since there is no additional information regarding the company's priorities—such as avoiding losses or increasing profit—it is more appropriate to evaluate both metrics together.

OEC Calculation

```
df_final <- df |>
  mutate(TP_init = badloans_num-typeII_init,
         TP_fin = badloans_num-typeII_fin,
         TN_init = badloans_num+goodloans_num-typeII_init-typeI_init,
         TN_fin = badloans_num+goodloans_num-typeII_fin-typeI_fin)
df_final <- df_final |>
  mutate(
    recall_init = if_else(badloans_num == 0, 0, round(TP_init / badloans_num),2),
    recall_fin  = if_else(badloans_num == 0, 0, round(TP_fin / badloans_num,2)),
    precision_init = if_else((TP_init + typeI_init) == 0, 0, round(TP_init / (TP_init + typeI_init)),2),
    precision_fin  = if_else((TP_fin + typeI_fin) == 0, 0, round(TP_fin / (TP_fin + typeI_fin)),2),
    accuracy_init  = round((TP_init+TN_init)/(goodloans_num+badloans_num),2),
    accuracy_fin = round((TP_fin+TN_fin)/(goodloans_num+badloans_num),2)
  )
```

```
df_recall <- df_final |>
  mutate(recall_diff = recall_fin-recall_init) |>
  select(loanofficer_id,Variant,recall_init,recall_fin,recall_diff)
df_precision <- df_final |>
  mutate(precision_diff = precision_fin-precision_init) |>
  select(loanofficer_id,Variant,precision_init,precision_fin,precision_diff)
df_accuracy <- df_final |>
  mutate(accuracy_diff = accuracy_fin-accuracy_init) |>
  select(loanofficer_id,Variant,accuracy_init,accuracy_fin,accuracy_diff)
```

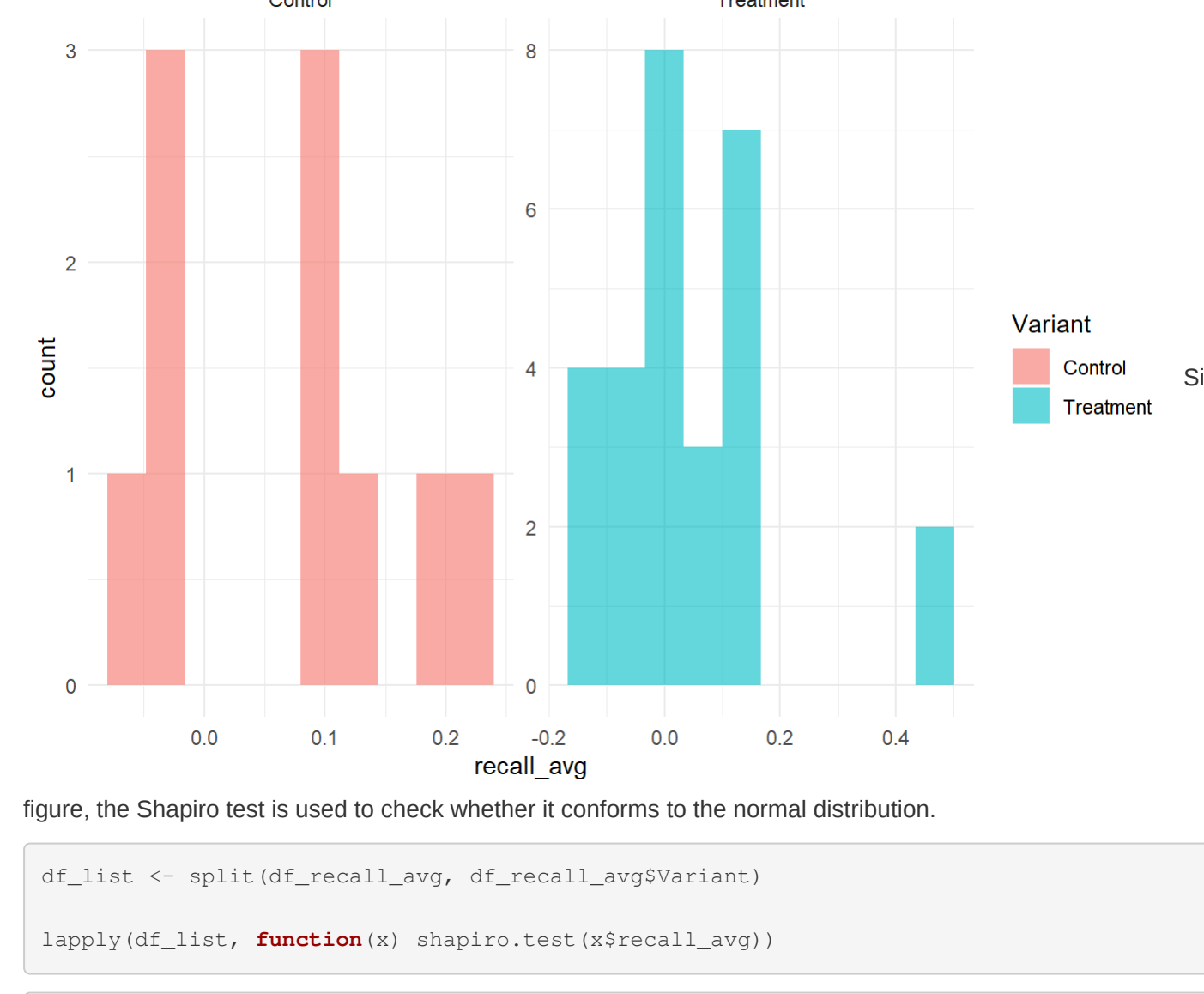
Since the AB test is based on individuals, the data needs to be averaged for each operator for ten days. However, this will result in only 10 data points for the control group and 28 data points for the experimental group.

```
df_recall_avg <- df_recall |>
  group_by(loanofficer_id, Variant) |>
  summarise(recall_avg = mean(recall_diff, na.rm = TRUE), .groups = "drop")
df_precision_avg <- df_precision |>
  group_by(loanofficer_id, Variant) |>
  summarise(precision_avg = mean(precision_diff, na.rm = TRUE), .groups = "drop")
df_accuracy_avg <- df_accuracy |>
  group_by(loanofficer_id, Variant) |>
  summarise(accuracy_avg = mean(accuracy_diff, na.rm = TRUE), .groups = "drop")
```

Recall

Draw the distribution graph of recall rate to check whether it conforms to the normal distribution

```
ggplot(df_recall_avg, aes(x = recall_avg, fill = Variant)) +
  geom_histogram(alpha = 0.6, bins = 10, position = "identity") +
  facet_wrap(~Variant, scales = "free") +
  theme_minimal()
```



Since it is difficult to see from the figure, the Shapiro test is used to check whether it conforms to the normal distribution.

```
df_list <- split(df_recall_avg, df_recall_avg$Variant)

lapply(df_list, function(x) shapiro.test(x$recall_avg))
```

```
## $Control
##
##  Shapiro-Wilk normality test
##
## data:  x$recall_avg
## W = 0.90135, p-value = 0.2267
##
##
## $Treatment
##
##  Shapiro-Wilk normality test
##
## data:  x$recall_avg
## W = 0.84347, p-value = 0.000692
```

In the Shapiro test, the null hypothesis is that the data follow a normal distribution. The results show that the control group did not reject the null hypothesis, indicating that it follows a normal distribution. In contrast, the experimental group rejected it, suggesting that it does not follow a normal distribution. Since neither groups follow a normal distribution, the Wilcox test is used to check whether the medians are different.

```
wilcox.test(recall_avg ~ Variant, data = df_recall_avg,
            exact = FALSE)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  recall_avg by Variant
## W = 161.5, p-value = 0.4862
## alternative hypothesis: true location shift is not equal to 0
```

```
df_recall_avg %>% wilcox_effsize(recall_avg ~ Variant)
```

```
## # A tibble: 1 × 7
##   .y.        group1  group2    effsize    n1    n2 magnitude
## * <chr>      <chr>   <chr>       <dbl> <int> <int> <ord>
## 1 recall_avg Control Treatment   0.116    10    28 small
```

```
df_recall_avg |>
  group_by(Variant) |>
  reframe(avg=mean(recall_avg))
```

```
## # A tibble: 2 × 2
##   Variant      avg
##   <chr>      <dbl>
## 1 Control   0.0677
## 2 Treatment 0.0489
```

```
r <- 0.1156434
d <- 2*r / sqrt(1 - r^2)
pwr.t2n.test(d = d, n1 = 10, n2 = 28, sig.level = 0.05,
             alternative = "two.sided")
```

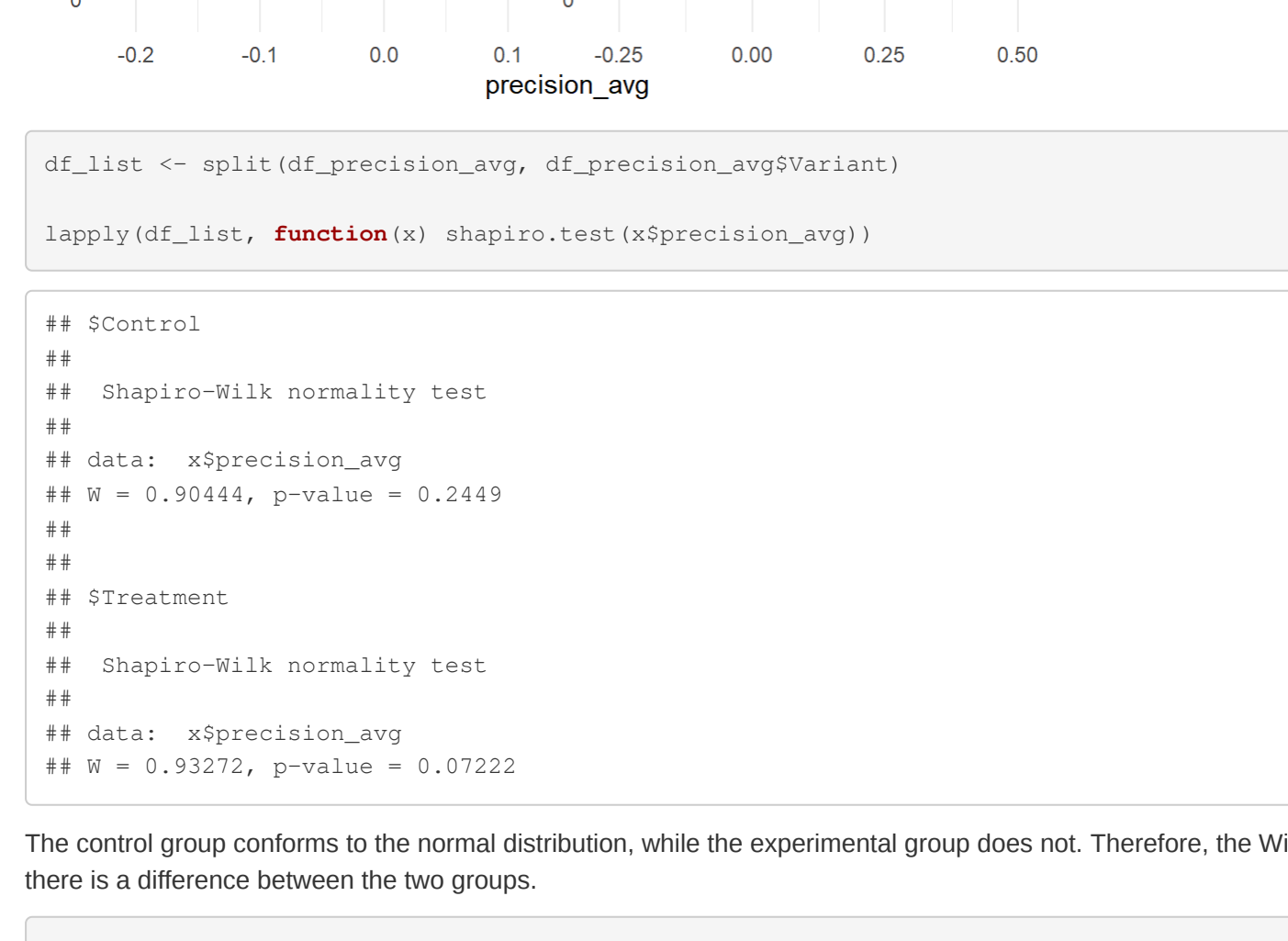
```
##
##      t test power calculation
##
##              n1 = 10
##              n2 = 28
##               d = 0.232849
##       sig.level = 0.05
##           power = 0.0943829
##     alternative = two.sided
```

The Wilcoxon test results showed that the difference in mean recall between the two groups was not significant (p = 0.4862 > 0.05). Therefore, the null hypothesis could not be rejected, indicating that there was no significant difference in recall performance between the Treatment and Control groups. The effect size was small (r = 0.116), suggesting a minimal difference between the groups. Listed out the outcome, the Treatment group had a slightly lower mean recall (0.0489) compared to the Control group (0.0677), but the gap was negligible. Moreover, the statistical power of the test was only 0.094, far below the conventional threshold of 0.8, implying that the current sample size was insufficient to detect such a small effect reliably.

Precision

Draw the distribution graph of precision rate to check whether it conforms to the normal distribution

```
ggplot(df_precision_avg, aes(x = precision_avg, fill = Variant)) +
  geom_histogram(alpha = 0.6, bins = 10, position = "identity") +
  facet_wrap(~Variant, scales = "free") +
  theme_minimal()
```




```
df_list <- split(df_precision_avg, df_precision_avg$Variant)

lapply(df_list, function(x) shapiro.test(x$precision_avg))
```

```
## $Control
##
##  Shapiro-Wilk normality test
##
## data:  x$precision_avg
## W = 0.90444, p-value = 0.2449
##
##
## $Treatment
##
##  Shapiro-Wilk normality test
##
## data:  x$precision_avg
## W = 0.93272, p-value = 0.07222
```

The control group conforms to the normal distribution, while the experimental group does not. Therefore, the Wilcox test is used to check whether there is a difference between the two groups.

```
wilcox.test(precision_avg ~ Variant, data = df_precision_avg,
            exact = FALSE)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  precision_avg by Variant
## W = 72, p-value = 0.02202
## alternative hypothesis: true location shift is not equal to 0
```

```
df_precision_avg |> wilcox_effsize(precision_avg ~ Variant)
```

```
## # A tibble: 1 × 7
##   .y.           group1  group2    effsize    n1    n2 magnitude
## * <chr>         <chr>   <chr>       <dbl> <int> <int> <ord>
## 1 precision_avg Control Treatment   0.374    10    28 moderate
```

```
df_precision_avg |>
  group_by(Variant) |>
  reframe(avg=mean(precision_avg))
```

```
## # A tibble: 2 × 2
##   Variant      avg
##   <chr>      <dbl>
## 1 Control   -0.06
## 2 Treatment  0.075
```

```
r <- 0.3742511
d <- 2*r / sqrt(1 - r^2)
pwr.t2n.test(d = d, n1 = 10, n2 = 28, sig.level = 0.05,
             alternative = "two.sided")
```

```
##
##      t test power calculation
##
##              n1 = 10
##              n2 = 28
##               d = 0.8071607
##       sig.level = 0.05
##           power = 0.5684556
##     alternative = two.sided
```

The Wilcox test results showed that the difference in average precision between the two groups was statistically significant (p = 0.022 < 0.05). Therefore, the null hypothesis was rejected, indicating a significant difference in precision performance between the Treatment and Control groups. The effect size was moderate (r = 0.374), suggesting a statistically significant difference between the groups. The Treatment group achieved a higher mean precision (0.075) compared to the Control group (-0.060), indicating an improvement under the Treatment condition. However, the statistical power of the test was only 0.568, which is below the conventional threshold of 0.8, suggesting that the current sample size may still be insufficient for reliably detecting effects of this magnitude.

Calculate the Sample Needed

Since this is an unbalanced dataset, the data in the treatment group is much larger than that in the control group. Therefore, I try to fix the number of treatment groups and calculate how many data sets are needed in the control group to achieve power = 0.8.

For Recall

```
#r <- 0.1156434
#d <- 2*r / sqrt(1 - r^2)
#pwr.t2n.test(d = d, n1 = NULL, n2 = 30, sig.level = 0.05, power = 0.8,alternative = "two.sided")
```

The calculation could not be completed because the effect size was too small. At such a small magnitude, the control group sample size would need to approach infinity for the power to reach 0.8.

For Precision

```
r <- 0.3742511
d <- 2*r / sqrt(1 - r^2)
pwr.t2n.test(d = d, n1 = NULL, n2 = 28,
             sig.level = 0.05, power = 0.8,
             alternative = "two.sided")
```

```
##
##      t test power calculation
##
##              n1 = 22.71269
##              n2 = 28
##               d = 0.8071607
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
```

For precision, given that the Treatment group is fixed at 28 observations, a total of 23 observations would be required to achieve a statistical power of 0.8. In other words, an additional 13 operators using the old model for 10 days would be needed to ensure sufficient power for detecting the observed effect.