Does Humor Impact Dating Success?

12/11/2023

1 Abstract

This study investigates the user personalities displayed on dating app profiles and their subsequent engagement levels. We aim to examine how varying personality portrayals in dating app profiles – independent of accompanying photos – have on user interactions and engagement. Comparing the effects of creating a neutral personality profile with a highly humorous personality profile will allow us to investigate dating app behavior. Our experiment generated the following results: dating profiles with humorous textual cues received more likes and matches as the same dating profiles without humorous textual cues with statistical significance. These results can be beneficial for online dating app users as they evaluate where to spend their time as they develop attractive dating profiles.

2 Introduction

Humor is a universal human experience that promotes bonding and positive social interaction. Historically, humor has been shown to decrease social tension and increase likability. The treatment involving the use of humor in the description is expected to change the measured behavior of humans who receive it. In the context of dating, where first impressions are critical, a humorous profile may pique interest by presenting someone as relatable, fun, and approachable, thereby increasing the chances of having dating success.

Our team is interested in investigating whether humorous expressions through textual cues affect dating success on a dating app. Current academic literature has evaluated this concept in similar, yet different, ways. First, a research study has evaluated how messages with different levels of humor impact a person's perceived attractiveness (Garove and Farley, 2015). Second, a research study evaluated how live humor and laughter in discourse are associated with perceived attractiveness (Hall, 2015). Lastly, a few research studies have evaluated attractiveness (Fiore et all, 2008) or first impressions (Zanden, 2021) specifically within dating apps. However, none of these research studies evaluate whether humorous textual cues within a dating app affect a person's attractiveness when primary cues like photographs and demographic information remain constant.

3 Experiment Concept

People spend hours daily on dating apps looking for potential partners. The selection process gets more complicated with the increase in the abundance of options available to them. While the superficial qualities and end goals are highly esteemed, a profile with a humorous caption tends to stand out among the numerous mundane ones as it indicates a witty personality, something which is not easily captured through just pictures and usual descriptions.

Our intervention variable is textual differences between the two Hinge dating profiles. Our subjects will experience a dating profile that is humorous and fun (treatment) and a dating profile that is plain and unimaginative (control). To reduce noise driven by confounding variables, we plan to ensure both profiles have the same characteristics and content apart from responses to question prompts. Controlling for these other factors is an important feature of this experience because while studies have shown that photos and demographics heavily influence dating profile experiences, this experiment is meant to focus on the causal effect personality traits expressed through text can have on someone's dating success on Hinge.

Due to limitations in timing and resources, we are utilizing only the Hinge application, but if we could replicate the results on other dating apps such as Tinder or Bumble, we hope to see similar results.

4 Experiment Design

The objective of this experiment is to explore the impact of incorporating a humorous personality into dating app profiles on user engagement. Specifically, we will compare the effects of creating a neutral personality profile with crafting a highly humorous and intriguing personality description on user interactions and engagement. We target the participants to be a diverse group of participants aged 18-25. Participants' profile photos will be the same to ensure that the study focuses solely on the impact of the humorous personality. Each of our team members will create fake profiles for both control and treatment groups based on the same photo. In the control group, the profile will be created as a neutral personality profile while in the treatment group, the profile will be created as a highly humorous and intriguing personality in the description. The fake profile will launch at the same time on the dating app for a specified period of time(e.g. 4-6 weeks). The number of likes and matches will be the key metric we are going to measure.

We have chosen two locations in the United States— New York City and San Francisco to introduce the fictitious dating profiles. The underlying assumption is that users of the dating app from these two U.S. locations will exhibit similar reactions regardless of whether a humorous personality is presented or not. Our intention is to keep other demographic differences to a minimum to strengthen this assumption. Moreover, to prevent a single participant from encountering two identical profiles simultaneously, we have made the deliberate decision to launch identical profiles in different locations. For instance, one profile (the control group) will be launched in San Francisco, while the other profile (the treatment group) will be launched in Los Angeles. This collection of data will serve as a foundational reference point, aiding us in ensuring that the two groups—those exposed to a neutral personality profile and those exposed to a humorous personality description—are comparable in terms of their demographic characteristics. It also safeguards against the possibility of a single individual viewing both profiles concurrently.

Since the control/treatment profile will be launched at the same time, the engagement metrics, such as the number of matches, likes, and comments will be zero as a baseline. The following variables are measured after the treatment (ie. activating the fake profile) to assess the impact on user engagement.

Number of profile likes: How many users have liked the profile Number of comments: How many users have commented on the profile Number of matches: How many users have matched with the profile

Dating apps necessitate mutual likes for a match to occur, for example, if one profile swipes right on five people, their maximum potential matches will be limited to five. We will standardize the maximum number of matches each profile can receive after the treatment. This will ensure that our created profile's actions (how much time we swipe right) do not unduly influence the effectiveness of our measures.

Through these variables both before and after administering the treatment, we can assess how the humor personality influenced user engagement on dating apps.

Due to the nature of this experiment creating fake dating profiles for our sample, our team will make up the participants of this experiment. Each team member will create 2 (or 4) dating profiles - one for control and one for treatment. These profiles will be identical in everything except for the profile question prompts, which will be neutral in control and humorous in treatment.

5 Data Cleaning

The provided dataset records essential metrics—swipes, likes, matches, and comments—for various user profiles on distinct dates, classified into control and treatment groups. It encompasses information pertaining to user profiles within a dating application (refer to the Appendix for a detailed description of the table columns).

We will employ two main tables for our data analysis. The primary focus will be on the 'exp_d' table for evaluating the metric of matches, which is our key criterion. In this table, we expand 10 daily swipes into each row, creating one sample per row. This expansion is justified by the fact that matches can only occur when swipes take place. Consequently, for the evaluation of the likes metric, we will have a total of 2752 samples.

Regarding the metrics of likes and comments, given that they are not strictly contingent on individual swipes, individuals using Hinge might express their interest by liking or commenting on the profile descriptions we've created. In this context, our unit of measurement for one sample will be based on the date of our experiment. This spans a timeframe of 36 days, from October 19th to November 14th, yielding a total of 294 samples.

6 Randomization Check

A randomization check was performed to evaluate whether our treatment and control groups make up representative samples. To perform this check two regression models were developed. The first model: null_mod is the simplest and captures the intercept. The second model: full_mod is more complex and includes profile characteristic variables such as: Who created the profile (Owner), The profile's name (Profile), The location – San Francisco or New York – the profile is active (City), The day the sample was collected (Day), and the number of swipes taken on the profile (Swipes).

The first simple model illustrates a 0.502 intercept to support the assumption that both treatment and control groups make up representative samples. The The second complex model with five covariates illustrates a 0.412 intercept and captures a small pattern; The City variable has a statistically significant correlation with treatment outcomes. This suggests that different profile locations (San Francisco or New York) contribute to sample differences between the treatment group and the control group.

```
stargazer(null_mod, full_mod,
          type = "latex",
         covariate.labels = c("Owner", "Profile", "City - San Francisco", "Day", "Swipes", "Intercept")
          )
```

% Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac at gmail.com % Date and time: Tue, Dec 12, 2023 - 12:53:43 AM

Table 1:

	Dependent variable:	
	Treat_binary	
	(1)	(2)
Owner		0.002
		(0.009)
Profile		−0.0002
		(0.007)
City - San Francisco		0.167^{***}
		(0.019)
Day		0.001
		(0.001)
Swipes		−0.001
		(0.003)
Intercept	0.502^{***}	0.412^{***}
	(0.010)	(0.039)
Observations	2,742	2,742
R^2	0.000	0.028
Adjusted R^2	0.000	0.026
Residual Std. Error	0.500 (df = 2741)	0.493 (df = 2736)
F Statistic		15.773^{***} (df = 5; 2736)
Note:		*p<0.1; **p<0.05; ***p<0.01

However, while the City variable is statistical significant the impact seems impractically significant at 0.167. To further illustrate the minimal impact the City variable has on treatment assignment, we've run an F-test between our two models (null mod, full mod). As suspected, while City location is statistically significant, the difference does not result in material differences in model performance. While variance is higher with our simple model, it still generates the same outcomes as the complex model for 97% of samples.

```
anova_mod <- anova(full_mod, null_mod, test = 'F')
anova_mod
```

```
## Analysis of Variance Table
##
## Model 1: Treat_binary ~ 1 + OwnerNum + ProfileNum + City + ExpDay_profile +
##     Swipes
## Model 2: Treat_binary ~ 1
```

```
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1   2736 666.29
## 2   2741 685.49 -5   -19.206 15.773 2.428e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

7 Data Exploration & Analytics

7.1 Design Assumptions

Our experiment design for the dating profiles consists of multiple assumptions. We limited our experiment profile features to 25-30 year old caucasian males in two of the biggest cities in the country, New York and San Francisco to ensure consistency in our results. While majority of the factors were kept consistent across all the accounts, some of them were prone to affect the potential outcomes more than the others. So we measured the variance of outcomes across the following factors to justify our assumptions.

7.1.1 Variance within Treatment and Control

The six sets of treatment and control accounts are distributed equivalently across the two selected cities. All of the vitals and virtues are kept exactly the same across each set except for the prompt and the location. While the difference in prompt is the treatment, the location of the accounts were blocked to ensure randomizations that keep treatment and control similar. Moreover this also reduced non-compliances, maening someone liking or matching a control account is unlikely to come across the treatment and vice versa. Before interpreting the results of statistical analysis of the datasets, we looked at the variances between the the control and treatment accounts. So far the following values are similar enough to support our assumption that they will allow for more confident interpretation of the differences in means of control and treatment. accounts.

```
tc_match_var <- d[,var(Matches), by = Treat]
tc_like_var <- d[, var(Likes), by = Treat]
tc_comments_var <- d[, var(Comments), by = Treat]

tc_match_var
```

```
##        Treat        V1
## 1:   Control 0.7276596
## 2: Treatment 1.7115673
```

```
tc_like_var
```

```
##        Treat       V1
## 1:   Control 6.049949
## 2: Treatment 9.866996
```

```
tc_comments_var
```

```
##        Treat        V1
## 1:   Control 0.1517730
## 2: Treatment 0.6755069
```

7.1.2 Variance within Locations

We have three treatment accounts and three control accounts in New York, and three treatment accounts and three control accounts in San Francisco.We assumed that both of the cities are similar in terms of popularity of the Hinge app, and should attract subjects in a similar way. However, to ensure the dispersion of values within each dataset is comparable we calculated the variance as per both treatment and control.

```
city_like_var_t <-  d[Treat == "Treatment", var(Likes), by = City]
city_like_var_c <-  d[Treat == "Control", var(Likes), by = City]

city_comment_var_t <-  d[Treat == "Treatment", var(Comments), by = City]
city_comment_var_c <-  d[Treat == "Control", var(Comments), by = City]
```

```r
city_match_var_t <-  d[Treat == "Treatment", var(Matches), by = City]
city_match_var_c <-  d[Treat == "Control", var(Matches), by = City]

city_like_var_t
```

```
##             City       V1
## 1: San Francisco 12.49858
## 2:      New York  6.61449
```

```r
city_like_var_c
```

```
##             City       V1
## 1:      New York 9.377215
## 2: San Francisco 1.479235
```

```r
city_comment_var_t
```

```
##             City        V1
## 1: San Francisco 0.3276899
## 2:      New York 1.0936013
```

```r
city_comment_var_c
```

```
##             City         V1
## 1:      New York 0.09113924
## 2: San Francisco 0.22185792
```

```r
city_match_var_t
```

```
##             City       V1
## 1: San Francisco 1.847468
## 2:      New York 1.563458
```

```r
city_match_var_c
```

```
##             City        V1
## 1:      New York 0.8213608
## 2: San Francisco 0.6158470
```

7.1.3 Variance within Profile Owners

The six sets of profiles while similar in most aspects differ from each other in a few ways such as different name, age, profile pictures, type of humor and promts. Since these play a crucial role in determining the likelihood of a profile being liked or matched, we computed and compared the variance of the dataset from each owner before adding them in the same group for further analysis.

```r
owner_like_var_t <-  d[Treat == "Treatment", var(Likes), by = Owner]
owner_like_var_c <-  d[Treat == "Control", var(Likes), by = Owner]

owner_comment_var_t <-  d[Treat == "Treatment", var(Comments), by = Owner]
owner_comment_var_c <-  d[Treat == "Control", var(Comments), by = Owner]

owner_match_var_t <-  d[Treat == "Treatment", var(Matches), by = Owner]
owner_match_var_c <-  d[Treat == "Control", var(Matches), by = Owner]

owner_like_var_t
```

```
##         Owner        V1
## 1:      Brian 17.113978
## 2:      Quazi 11.628571
## 3: Erin Smith 10.357143
## 4:         KT  1.328042
## 5:       Luka  3.092803
```

```r
owner_like_var_c
```

```
##         Owner         V1
## 1:      Brian  3.4946237
## 2:      Quazi 14.5571429
## 3: Erin Smith  2.0640394
## 4:         KT  1.4761905
## 5:       Luka  0.9667339
```

```r
owner_comment_var_t
```

```
##         Owner        V1
## 1:      Brian 0.1612903
## 2:      Quazi 0.6333333
## 3: Erin Smith 1.8916256
## 4:         KT 0.1944444
## 5:       Luka 0.2178030
```

```r
owner_comment_var_c
```

```
##         Owner         V1
## 1:      Brian 0.13978495
## 2:      Quazi 0.09047619
## 3: Erin Smith 0.26108374
## 4:         KT 0.03571429
## 5:       Luka 0.19354839
```

```r
owner_match_var_t
```

```
##         Owner       V1
## 1:      Brian 1.692473
## 2:      Quazi 1.228571
## 3: Erin Smith 1.544335
## 4:         KT 1.941799
## 5:       Luka 1.530303
```

```r
owner_match_var_c
```

```
##         Owner        V1
## 1:      Brian 0.3892473
## 2:      Quazi 0.8476190
## 3: Erin Smith 0.5344828
## 4:         KT 1.0621693
## 5:       Luka 0.6925403
```

7.1.4 Variance within Humor Type

Most of the profiles used humor generated by AI tools like ChatGPT, however one set of profiles did not: Owner = "KT" and Owner = "Erin". In order to ensure consistency, the above variance comparisons among the owners also ensures the assumption that the humor type in the prompts regardless of the source are similar enough to be grouped together.

```r
dbr <- d[Owner == "Brian"]
plotl1 <- ggplot(dbr, aes(x = Date, y = Likes, colors = Date)) +
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (Brian)", x = "Date", y = "Likes")

der <- d[Owner == "Erin Smith"]
plotl2 <- ggplot(der, aes(x = Date, y = Likes, colors = Date)) +
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (Erin)", x = "Date", y = "Likes")

dkt <- d[Owner == "KT"]
plotl3 <- ggplot(dkt, aes(x = Date, y = Likes, colors = Date)) +
```

```r
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (KT)", x = "Date", y = "Likes")

dlu <- d[Owner == "Luka"]
plotl4 <- ggplot(dlu, aes(x = Date, y = Likes, colors = Date)) +
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (Luka)", x = "Date", y = "Likes")

dqu <- d[Owner == "Quazi"]
plotl5 <- ggplot(dqu, aes(x = Date, y = Likes, colors = Date)) +
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (Quazi)", x = "Date", y = "Likes")


stacked_like_plots <- plot_grid(plotl1, plotl2, plotl3, plotl4, plotl5, ncol = 1)
stacked_like_plots
```


```r
dbr <- d[Owner == "Brian"]
plotm1 <- ggplot(dbr, aes(x = Date, y = Matches, colors = Date)) +
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (Brian)", x = "Date", y = "Matches")

der <- d[Owner == "Erin Smith"]
plotm2 <- ggplot(der, aes(x = Date, y = Matches, colors = Date)) +
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (Erin)", x = "Date", y = "Matches")

dkt <- d[Owner == "KT"]
plotm3 <- ggplot(dkt, aes(x = Date, y = Matches, colors = Date)) +
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (KT)", x = "Date", y = "Matches")
```

```r
dlu <- d[Owner == "Luka"]
plotm4 <- ggplot(dlu, aes(x = Date, y = Matches, colors = Date)) +
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (Luka)", x = "Date", y = "Matches")

dqu <- d[Owner == "Quazi"]
plotm5 <- ggplot(dqu, aes(x = Date, y = Matches, colors = Date)) +
  geom_line(color = "blue", size = 5) +
  labs(title = "Likes over Time (Erin)", x = "Date", y = "Matches")

stacked_match_plots <- plot_grid(plotm1, plotm2,plotm3,plotm4,plotm5, ncol = 1)
stacked_match_plots
```

7.2 Challenges and Limitations

We resorted to using artificial intelligence generated pictures for profile owners to avoid any privacy invasion of an actual person or possible accusation opf cat fishing. As advanced as AI is, almost half of our accounts got flagged at one point due to Hinge Trust & Service policy violation during the four week duration of the experiment. Some comments indicated users were able to tell that the accounts were counterfeit. The first set got deleted on day 2, so we re-created similar accounts. The second set got flagged around day 7 and again we followed similar recovery plan. The third set got banned on week 3, and demanded proof of the fake profile owner data were using, so that one could not be recovered.

Our only interaction on the profiles was to like 10 accounts in a row per day. This limited interaction reduced our chances of getting more matches and likes as the profile visibility is dependent on how interactive the users are, and other than swipes, it is difficult to maintain consistency of interactions between each user.

8 Experimentation

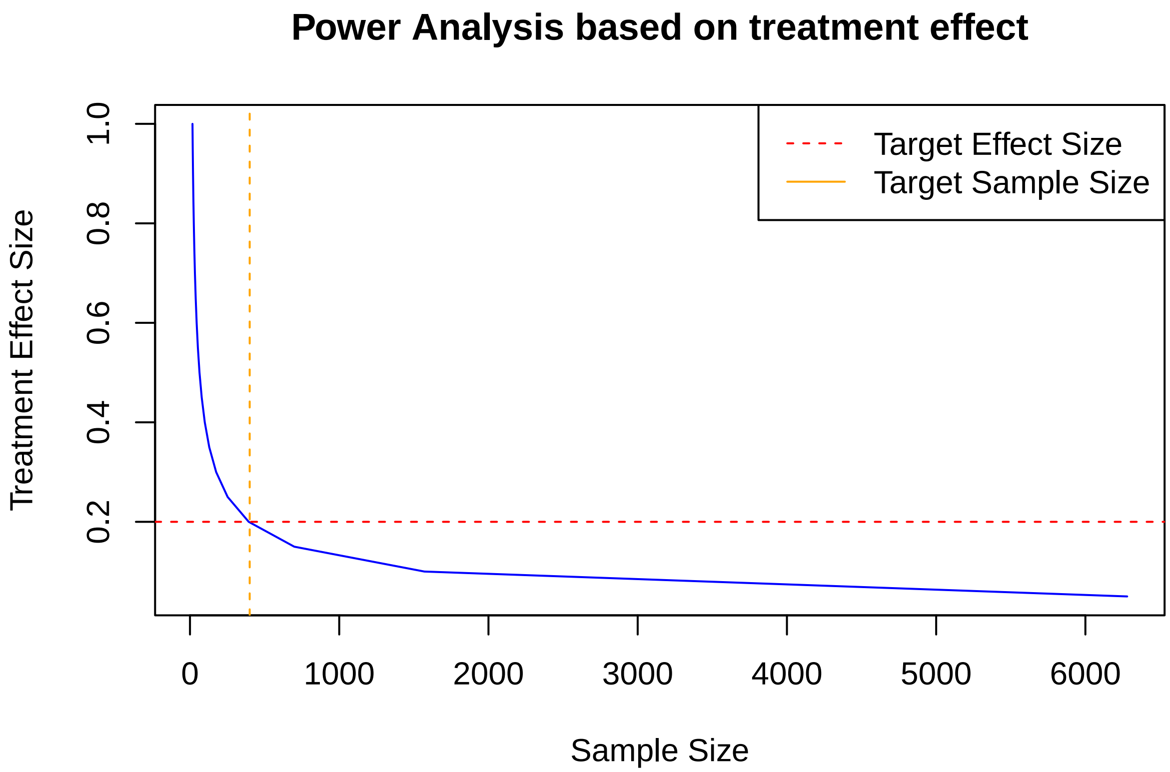
In this section, we dived deeper into the analysis of our experiment, aiming to elucidate the impact of incorporating a humorous personality into dating app profiles while accounting for various control variables. Our primary objective remains consistent: to assess the effect of the treatment variable, denoting the presence of a highly humorous and intriguing personality description within the profile, on user engagement metrics, specifically, the number of likes and matches.

We performed a power analysis simulation before the experiment, in order to ensure a high probability (80%) of detecting a treatment effect of 20%. In this case, a treatment effect of 20% suggests that the study is designed to detect a 20% difference between groups, which could be a treatment group and a control group. A sample size of 400 is recommended for the statistical test being conducted.

```
# Plot the results
plot(sample_sizes, treatment_effect_sizes, type = "l", col = "blue",
     xlab = "Sample Size", ylab = "Treatment Effect Size",
     main = "Power Analysis based on treatment effect")

# Add a horizontal line for the desired power
abline(h = target_effect_size, col = "red", lty = 2)
abline(v = target_sample_size, col = "orange", lty = 2)

# Add legend
legend("topright", legend = c("Target Effect Size", "Target Sample Size"),
       col = c("red", "orange"), lty = c(2, 1))
```



This experiment was conducted for 45 days. However, due to unforeseen issues with profiles being removed unexpectedly, we were not able to collect data for all of those days. We were able to get a total of 283 samples in terms of unique profiles collecting data per day. To analyze the number of matches each profile received, we expanded the data for that model so that each sample is actually a single "swipe" or "like" that the profile sent in a day. This means that the outcome variable for each sample is a binary indicator variable indicating whether or not that swipe garnered a match or not. The total number of samples for our match model is 2742.

Analyzing the likes that a profile received is a bit more complicated than that. It is unclear exactly how Hinge decides to show certain profiles to users, so it is impossible to determine the total number of users that actually encoutered our profiles in a given day. Therefore, we will analyze the lkes on a per day sample. That is to say that a single sample represents a single day in our experiment and the outcome variable is the nubmer of likes that a profile received. Thhe total number of samples for our likes model is 283.

8.1 Number of Matches Experiment - Simple

```
match_d <- exp_d[, c('match',"Treat_binary")]
# Perform a t-test
```

```r
t_test_matches <- t.test(match ~ Treat_binary, d=match_d)
t_test_matches
```

```
##
##  Welch Two Sample t-test
##
## data:  match by Treat_binary
## t = -3.3309, df = 2676.3, p-value = 0.0008773
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -0.06248673 -0.01617828
## sample estimates:
## mean in group 0 mean in group 1
##      0.08784773      0.12718023
```

Using a t-test, we compared the mean matches between two groups. The t-value is approximately, -3.3309, and the p-value is 0.0008773. The results suggest a statistically significant difference in means between control and treatment, as indicated by the small p-value and the confidence interval(-0.06248673 -0.01617828) that does not include zero. The negative t-value suggests that the mean in treatment is higher than the mean in control."

```r
mod <- lm(match ~ Treat_binary, data=match_d)
robust_mod <- coeftest(mod, vcov = vcovHC(mod, type = "HC3"))

stargazer(mod, type = "text", header=FALSE,
          title = "Simple Regression Resutls - Matches",
          se = list(sqrt(diag(vcovHC(mod, type = "HC3")))))
```

```
##
## Simple Regression Resutls - Matches
## ===============================================
##                         Dependent variable:
##                     ---------------------------
##                                match
## -----------------------------------------------
## Treat_binary                 0.039***
##                               (0.012)
##
## Constant                     0.088***
##                               (0.008)
##
## -----------------------------------------------
## Observations                   2,742
## R2                             0.004
## Adjusted R2                    0.004
## Residual Std. Error      0.309 (df = 2740)
## F Statistic          11.082*** (df = 1; 2740)
## ===============================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
```

The 'Treat_binary' coefficient (0.039) indicates that, on average, there is a 0.039-unit increase in the 'match' variable for treatment group, which is 'Treat_binary'= 1. The p-value associated with 'Treat_binary' is highly significant (***p<0.01), suggesting that 'Treat_binary' is a significant predictor of 'match.' Since we are exploring whether there is a significant difference between control and treatment groups in matches, the t-test and a simple linear regression produce a very similar result.

8.2 Number of Matches Expriment - Comphrehesive

To further enhance the analysis of the impact of humor on dating app matches, we include additional variables, such as dating app profile, which can capture and account for potential influence on the number of matches, and the city or days of the week. I believe that this helps to isolate the true effect of treat_binary (humor) on matches, reducing extraneous noise and improving the accuracy the estimates. Additionally, including more variables allows you to understand how different factors interact and contribute to the outcome. This creates a more nuanced picture of the relationships between humor and matches, rather than just a single, isolated effect. Lastly, we will be able to reduce the risk of omitted variable bias and obtain more reliable results.

```
mod_1 <- lm(match ~ Treat_binary + CityNum, data=exp_d)
robust_mod_1 <- coeftest(mod_1, vcov = vcovHC(mod_1, type = "HC3"))

mod_2 <- lm(match ~ Treat_binary + CityNum + factor(ProfileNum), data=exp_d)
robust_mod_2 <- coeftest(mod_2, vcov = vcovHC(mod_2, type = "HC3"))

mod_3 <- lm(match ~ Treat_binary + CityNum + factor(ProfileNum) + factor(weekday), data=exp_d)
robust_mod_3 <- coeftest(mod_3, vcov = vcovHC(mod_3, type = "HC3"))

mod_4 <- lm(match ~ Treat_binary + CityNum + factor(ProfileNum) + factor(weekday) + ExpDay_profile, data=exp_
robust_mod_4 <- coeftest(mod_4, vcov = vcovHC(mod_4, type = "HC3"))

stargazer(
  mod_1,
  mod_2,
  mod_3,
  mod_4,
  type = "latex", header=FALSE, column.sep.width = ".2pt",
  title = "Regression Results - Matches",
  font.size="footnotesize", no.space=TRUE,
  se = list(
    sqrt(diag(vcovHC(mod_1, type = "HC3"))),
    sqrt(diag(vcovHC(mod_2, type = "HC3"))),
    sqrt(diag(vcovHC(mod_3, type = "HC3"))),
    sqrt(diag(vcovHC(mod_4, type = "HC3")))
  ))
```

8.3 Matches Model Interpretation

Model 1: We included city data (limited to San Francisco and New York City) to assess its potential influence on dating app matches. As anticipated, incorporating humor into profiles significantly increased matches, regardless of city location. This aligns with our assumption that individuals in these two major cities would exhibit similar behavior regarding online dating preferences.

Model 2: By incorporating "profile_id" as a categorical variable in our analysis, we were able to demonstrate that all six profiles exhibited statistically significant differences in their average number of matches compared to the control group. This finding provides valuable insights into how individual profile characteristics influence dating app success and highlights the importance of considering individual variability in such research.

Model 3: This model further expands by incorporating the day of the week the user profile was active (coded as a factor with seven levels). Consistent with the previous model, the results reaffirm that incorporating humor significantly increases the number of matches received, and individual profile characteristics remain influential factors in dating app success. However, similar to city location, weekday does not appear to have a statistically significant impact on match count.

Model 4: This model expands upon previous iterations by incorporating "ExpDay_profile," which signifies the duration since a profile's creation. This addition enables us to examine the relationship between the number of days a profile is active on the app (exposure days) and the number of matches received. The negative coefficient (-0.00238) associated with ExpDay_profile reveals that as the exposure days increase, the average number of matches received decreases. This finding suggests the potential presence of a decay effect, where the initial boost experienced by new profiles gradually diminishes over time. Consequently, the importance of profile refresh or update strategies to sustain user engagement and maintain match potential is highlighted.

8.4 Number of Likes Experiment - Simple

```
like_d <- d[, c('Likes',"Treat_binary")]
```

Table 2: Regression Results - Matches

	Dependent variable:			
	match			
	(1)	(2)	(3)	(4)
Treat_binary	0.039***	0.040***	0.040***	0.040***
	(0.012)	(0.012)	(0.012)	(0.012)
CityNum	0.001	0.001	0.001	0.001
	(0.012)	(0.012)	(0.012)	(0.012)
factor(ProfileNum)2		0.108***	0.107***	0.079**
		(0.033)	(0.033)	(0.034)
factor(ProfileNum)3		0.067***	0.067***	0.065***
		(0.017)	(0.017)	(0.017)
factor(ProfileNum)4		0.034**	0.035**	0.037**
		(0.015)	(0.015)	(0.015)
factor(ProfileNum)5		0.089***	0.088***	0.076***
		(0.020)	(0.020)	(0.020)
factor(ProfileNum)6		0.080***	0.080***	0.069***
		(0.021)	(0.021)	(0.021)
factor(weekday)1			0.001	0.001
			(0.019)	(0.019)
factor(weekday)2			0.029	0.027
			(0.022)	(0.022)
factor(weekday)3			0.018	0.016
			(0.022)	(0.022)
factor(weekday)4			0.013	0.009
			(0.020)	(0.020)
factor(weekday)5			0.026	0.021
			(0.022)	(0.022)
factor(weekday)6			0.024	0.027
			(0.021)	(0.021)
ExpDay_profile				−0.002***
				(0.001)
Constant	0.087***	0.036***	0.021	0.060***
	(0.011)	(0.013)	(0.018)	(0.022)
Observations	2,742	2,742	2,742	2,742
R^2	0.004	0.017	0.018	0.022
Adjusted R^2	0.003	0.014	0.013	0.017
Residual Std. Error	0.309 (df = 2739)	0.308 (df = 2734)	0.308 (df = 2728)	0.307 (df = 2727)
F Statistic	5.541*** (df = 2; 2739)	6.556*** (df = 7; 2734)	3.798*** (df = 13; 2728)	4.315*** (df = 14; 2727)
Note:				*p<0.1; **p<0.05; ***p<0.01

```r
t_test_likes <- t.test(Likes ~ Treat_binary, d=like_d)
t_test_likes
```

```
##
##  Welch Two Sample t-test
##
## data:  Likes by Treat_binary
## t = -1.6386, df = 266.54, p-value = 0.1025
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -1.2094231  0.1107316
## sample estimates:
## mean in group 0 mean in group 1
##        1.992908        2.542254
```

We conducted a t-test to compare the mean number of matches between the treatment and control groups. The resulting t-value was approximately -1.6386, and the corresponding p-value was found to be 0.1025. These outcomes suggest that there is no statistically significant difference in the number of matches between the two groups.

Furthermore, the 95% confidence interval for the difference in means, (-1.2094231 to 0.1107316), encompasses zero. This observation signifies that we do not have sufficient evidence to establish any statistical significance in the comparison of mean matches between the treatment and control groups.

```r
mod <- lm(Likes ~ Treat_binary, data=d)
robust_mod <- coeftest(mod, vcov = vcovHC(mod, type = "HC3"))

stargazer(mod, type = "latex", header=FALSE,
          title = "Simple Regression Results - Likes",
          se = list(sqrt(diag(vcovHC(mod, type = "HC3")))))
```

Table 3: Simple Regression Results - Likes

	Dependent variable:
	Likes
Treat_binary	0.549
	(0.336)
Constant	1.993***
	(0.208)
Observations	283
R^2	0.009
Adjusted R^2	0.006
Residual Std. Error	2.822 (df = 281)
F Statistic	2.680 (df = 1; 281)
Note:	*p<0.1; **p<0.05; ***p<0.01

We conducted a regression analysis to examine the impact of treatment, on the number of 'Likes' received in dating app profiles. The coefficient for the treatment variable is 0.549, indicating that profiles with a humorous personality description (in the treatment group) tend to receive approximately 0.549 more 'Likes' compared to profiles without such descriptions (in the control group). However, this result does not reach statistical significance ($p > 0.1$), suggesting that the effect of the treatment on 'Likes' is not statistically conclusive.

The constant term in the model is 1.993, representing the estimated number of 'Likes' for profiles in the control group. The overall model explains a small portion of the variability in 'Likes' (R-squared = 0.009) and has a limited ability to predict the number of 'Likes' based on the treatment variable. The F-statistic is 2.680, with a corresponding p-value greater than 0.1, indicating that the model does not provide strong evidence for the relationship between the treatment variable and 'Likes.'

In summary, while there is a positive association between the treatment and the number of 'Likes,' our results do not demonstrate statistical significance. Therefore, the impact of a humorous personality description on the number of 'Likes' in dating app profiles remains inconclusive based on this analysis.

8.5 Number of Likes Expriment - Comphrehesive

Building upon the foundation laid in the earlier section, where we examined the simple relationships between the treatment variable and our outcome variables, Likes and Matches, we recognize the need to consider a broader context. By doing so, we aim to control for potential confounding variables that might otherwise obscure the true impact of the treatment variable. These control variables encompass demographic and contextual factors, allowing us to explore how the humorous personality description affects user engagement while holding other relevant factors constant.

```r
mod_1 <- lm(Likes ~ Treat_binary + Swipes, data=d)
robust_mod_1 <- coeftest(mod_1, vcov = vcovHC(mod_1, type = "HC3"))

mod_2 <- lm(Likes ~ Treat_binary + Swipes + CityNum, data=d)
robust_mod_2 <- coeftest(mod_2, vcov = vcovHC(mod_2, type = "HC3"))

mod_3 <- lm(Likes ~ Treat_binary + Swipes + CityNum + factor(Owner), data=d)
robust_mod_2 <- coeftest(mod_3, vcov = vcovHC(mod_3, type = "HC3"))

mod_4 <- lm(Likes ~ Treat_binary + Swipes + CityNum + factor(Owner) +
              factor(DOW), data=d)
robust_mod_4 <- coeftest(mod_4, vcov = vcovHC(mod_4, type = "HC3"))

mod_5 <- lm(Likes ~ Treat_binary + Swipes + CityNum + factor(Owner) +
              factor(DOW) + ExpDay, data=d)
robust_mod_5 <- coeftest(mod_5, vcov = vcovHC(mod_5, type = "HC3"))

stargazer(
  mod_1,
  mod_2,
  mod_3,
  mod_4,
  mod_5,
  type = "latex", header=FALSE, column.sep.width = ".2pt",
  title = "Regression Results - Likes",
  font.size="footnotesize", no.space=TRUE,
  se = list(
    sqrt(diag(vcovHC(mod_1, type = "HC3"))),
    sqrt(diag(vcovHC(mod_2, type = "HC3"))),
    sqrt(diag(vcovHC(mod_3, type = "HC3"))),
    sqrt(diag(vcovHC(mod_4, type = "HC3"))),
    sqrt(diag(vcovHC(mod_5, type = "HC3")))
  ))
```

The coefficient for our binary treatment variable varies between 0.549 and 0.598 across different models, suggesting that profiles featuring a humorous description tend to attract more "Likes" compared to profiles without humor. However, it's worth noting that not all p-values across the models consistently reach statistical significance, with only 3 out of 5 falling below the 0.05 threshold.

The profile owner variable indicates the individual who created the profile (i.e., the group member responsible for the user's creation). Notably, profiles owned by Quazi consistently exhibit positive coefficients ranging from 3.290 to 3.665 across models. This implies that profiles created by Quazi have a highly statistically significant impact on the number of "Likes" received in these models at the 0.01 significance level.

Time also emerges as a significant factor in this experiment. In models that consider the day of the week, Thursday demonstrates a coefficient between 0.932 and 0.984, achieving statistical significance at the 0.05 level. Furthermore, the day of the experiment itself proves to be highly significant at the 0.01 level, with a coefficient of -0.057. This suggests a negative effect on the number of 'Likes' a profile receives on later days during the experiment.

Table 4: Regression Results - Likes

	Dependent variable:				
	Likes				
	(1)	(2)	(3)	(4)	(5)
Treat_binary	0.549	0.589*	0.593**	0.590**	0.598**
	(0.337)	(0.321)	(0.290)	(0.292)	(0.283)
Swipes	0.023	0.023	0.028	0.031	0.016
	(0.038)	(0.038)	(0.027)	(0.029)	(0.022)
CityNum		−0.030	−0.031	−0.030	−0.031
		(0.032)	(0.029)	(0.029)	(0.028)
factor(Owner)Erin Smith			0.695	0.714	1.032**
			(0.528)	(0.522)	(0.492)
factor(Owner)KT			−0.638	−0.590	−0.377
			(0.443)	(0.452)	(0.410)
factor(Owner)Luka			0.037	0.082	0.415
			(0.450)	(0.442)	(0.366)
factor(Owner)Quazi			3.647***	3.665***	3.290***
			(0.700)	(0.697)	(0.684)
factor(DOW)Monday				0.475	0.522
				(0.446)	(0.445)
factor(DOW)Saturday				0.122	0.156
				(0.422)	(0.401)
factor(DOW)Sunday				−0.067	0.086
				(0.392)	(0.405)
factor(DOW)Thursday				0.984**	0.932**
				(0.498)	(0.468)
factor(DOW)Tuesday				0.710	0.765
				(0.638)	(0.606)
factor(DOW)Wednesday				0.532	0.511
				(0.523)	(0.498)
ExpDay					−0.057***
					(0.013)
Constant	1.770***	2.502**	1.894**	1.449*	2.753***
	(0.410)	(1.005)	(0.775)	(0.824)	(0.796)
Observations	283	283	283	283	283
R^2	0.010	0.013	0.242	0.258	0.321
Adjusted R^2	0.003	0.002	0.223	0.222	0.285
Residual Std. Error	2.827 (df = 280)	2.828 (df = 279)	2.495 (df = 275)	2.496 (df = 269)	2.393 (df = 268)
F Statistic	1.408 (df = 2; 280)	1.201 (df = 3; 279)	12.571*** (df = 7; 275)	7.204*** (df = 13; 269)	9.035*** (df = 14; 268)
Note:					*p<0.1; **p<0.05; ***p<0.01

In summary, the inclusion of humor in dating app profiles is linked to an increase in the number of "Likes" and is statistically significant. Additional factors, such as specific profile identifiers and the timing of the experiment, including the day of the week and the day of the experiment, also exert influence on "Likes." However, the impact of factors like the day of the week, the user's city location, and the number of daily swipes often appears less pronounced and frequently fails to reach statistical significance in the presented models.

8.6 Experimentation Results

The results from both the matches and likes analysis indicate that having a humorous personality description significantly influences user engagement on dating apps, particularly through an increase in both matches and likes. The control variables in these models do not exhibit significant effects on being in the treatment group, reinforcing the importance of personality traits expressed through text in shaping user interactions and experiences on dating apps.

8.7 Heterogeneous Treatment Effects (HTE)

There are three groups in particular for our experiment to break down: treatment effects by Owner, City, and Time. Owner will measure if there is any disparate impacts by the different team members, essentially looking for if any owners are more humorous than others. City will measure if the selection of New York as the treatment or control had any impact on the results. Time will measure any impact by weeks since opening the account (there is an internet theory that Hinge will boost the likes you receive when first opening an account).

```r
#Create week since start
starting_dates <- d %>%
  group_by(Profile_Name, City, Treat) %>%
  summarise(start_date = min(Date)) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'Profile_Name', 'City'. You can override
## using the `.groups` argument.
```

```r
d <- d %>%
  left_join(starting_dates, by = c("Profile_Name", "City", "Treat"))

d <- d %>%
  mutate(
    weeks_since_start = ceiling((as.numeric(Date - start_date) + 6) / 7)
  )

# Convert dataset to a wide format
# Reformat data into a wide format
final_dataset <- d %>%
  # Create separate datasets for likes and matches
  mutate(likes_control = ifelse(Treat == "Control", Likes, NA),
         likes_treat = ifelse(Treat == "Treatment", Likes, NA),
         matches_control = ifelse(Treat == "Control", Matches, NA),
         matches_treat = ifelse(Treat == "Treatment", Matches, NA),
         City_Control = ifelse(Treat == "Control", City, NA),
         City_Treat = ifelse(Treat == "Treatment", City, NA)) %>%
  # Group by Owner, Profile_Name, Date, and weeks_since_start
  group_by(Owner, Profile_Name, Date, weeks_since_start) %>%
  # Summarize to collapse into single rows
  summarise(likes_control = sum(likes_control, na.rm = TRUE),
            likes_treat = sum(likes_treat, na.rm = TRUE),
            matches_control = sum(matches_control, na.rm = TRUE),
            matches_treat = sum(matches_treat, na.rm = TRUE),
            City_Control = first(na.omit(City_Control)),
            City_Treat = first(na.omit(City_Treat))) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'Owner', 'Profile_Name', 'Date'. You can
## override using the `.groups` argument.
```

```r
# calculate TE for each row
final_dataset <- final_dataset %>%
  mutate(treatment_effect_likes = likes_treat - likes_control,
         treatment_effect_matches = matches_treat - matches_control)
```

9 HTE by Owner

```r
# average by owner
average_treatment_effect <- final_dataset %>%
  group_by(Owner) %>%
  summarise(average_effect_likes = mean(treatment_effect_likes, na.rm = TRUE),
            average_effect_matches = mean(treatment_effect_matches, na.rm = TRUE)) %>%
  ungroup()


average_treatment_effect
```

```
## # A tibble: 5 x 3
##   Owner      average_effect_likes average_effect_matches
##   <chr>                     <dbl>                  <dbl>
## 1 Brian                     1.03                   0.226
## 2 Erin Smith                1.28                   0.517
## 3 KT                        0                      0.25
## 4 Luka                      0.606                  0.273
## 5 Quazi                    -0.429                  0.810
```

From a simple average treatment effect by owner, there seems to be significant discrepancy between owners. For likes, Quazi's treatment had a negative impact, while Erin's treatment had the highest impact. Interestingly, Quazi's treatment had the highest impact on matches, indicating that perhaps there is some complexity to Quazi's humor.

```r
# average by owner
# For likes
anova_likes <- aov(treatment_effect_likes ~ Owner, data = final_dataset)
summary(anova_likes)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Owner         4   51.1  12.787   2.147 0.0783 .
## Residuals   137  815.8   5.955
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# For matches
anova_matches <- aov(treatment_effect_matches ~ Owner, data = final_dataset)
summary(anova_matches)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Owner         4    6.0   1.501   0.762  0.552
## Residuals   137  269.7   1.969
```

Performing an ANOVA on likes and matches, we can see that the ANOVA test for likes is almost statistically significant with a p-value of 0.0783. The p-value suggests that there are differences in the average number of likes between different owners, but these differences are not statistically significant at the 5% level (though they are close). It implies that while there is some variation in likes attributable to the different owners, this variation might be due to chance.

The ANOVA test for matches is not statistically significant at all with a p-value of 0.552, indicating any differences are probably due to chance.

10 HTE By City

We next want to investigate whether the city had a significant impact on treatments. While the last test was interesting to measure which owner was the most humorous, this is the most important test for our team. We made a significant assumption that city selection would not impact the likes or matches received, so now we must confirm this.

```
average_treatment_effect_city <- final_dataset %>%
  group_by(City_Control, City_Treat) %>%
  summarise(average_effect_likes = mean(treatment_effect_likes, na.rm = TRUE),
            average_effect_matches = mean(treatment_effect_matches, na.rm = TRUE)) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'City_Control'. You can override using the
## `.groups` argument.
```

```
average_treatment_effect_city
```

```
## # A tibble: 3 x 4
##   City_Control  City_Treat    average_effect_likes average_effect_matches
##   <chr>         <chr>                        <dbl>                  <dbl>
## 1 New York      San Francisco                0.288                  0.388
## 2 San Francisco New York                     0.934                  0.377
## 3 <NA>          New York                     0                      1
```

We can immediately see that selecting New York as the treatment city provided approximately 0.6 more likes per day than picking San Francisco as a treatment. There seems to be very little variation in the average effect on matches. There is an additional row here which indicates a data defect. It seems there was a row that that did not have the control city marked.

Let's see if the treatment city is statistically significant.

```
# T-Test for Likes
t_test_likes_city <- t.test(treatment_effect_likes ~ interaction(City_Control, City_Treat),
                            data = final_dataset)
t_test_likes_city
```

```
##
##  Welch Two Sample t-test
##
## data:  treatment_effect_likes by interaction(City_Control, City_Treat)
## t = 1.6065, df = 138.45, p-value = 0.1105
## alternative hypothesis: true difference in means between group San Francisco.New York and group New York.S
## 95 percent confidence interval:
##  -0.1493147  1.4431672
## sample estimates:
## mean in group San Francisco.New York mean in group New York.San Francisco
##                            0.9344262                            0.2875000
```

```
# T-Test for Matches
t_test_matches_city <- t.test(treatment_effect_matches ~ interaction(City_Control, City_Treat),
                              data = final_dataset)
t_test_matches_city
```

```
##
##  Welch Two Sample t-test
##
## data:  treatment_effect_matches by interaction(City_Control, City_Treat)
## t = -0.045581, df = 138.66, p-value = 0.9637
## alternative hypothesis: true difference in means between group San Francisco.New York and group New York.S
## 95 percent confidence interval:
##  -0.4637907  0.4428891
## sample estimates:
## mean in group San Francisco.New York mean in group New York.San Francisco
##                            0.3770492                            0.3875000
```

The t-test for likes has a p-value of 0.11 which is not statistically significant. Similiar to the owner analysis, it implies that while there is some variation in likes attributable to the treatment city, this variation might be due to chance. As expected for our matches t-test, there is very a high p-value of 0.96, indicating that the variation is most likely due to chance.

11 HTE by Weeks

When creating our hinge accounts, we noticed that the number of likes was very high at first and then tapered off. We want to confirm if this has any impact on our treatment.

```
#HTE by weeks
average_treatment_effect_week <- final_dataset %>%
  group_by(weeks_since_start) %>%
  summarise(average_effect_likes = mean(treatment_effect_likes, na.rm = TRUE),
            average_effect_matches = mean(treatment_effect_matches, na.rm = TRUE)) %>%
  ungroup()

average_treatment_effect_week
```

```
## # A tibble: 6 x 3
##   weeks_since_start average_effect_likes average_effect_matches
##               <dbl>                <dbl>                  <dbl>
## 1                 1               1.25                   0.333
## 2                 2               1.18                   0.7
## 3                 3               0.257                  0.314
## 4                 4              -0.0625                 0.0312
## 5                 5               0.375                  0.25
## 6                 6               0.714                  1
```

Interestingly, we can see the average effect of likes does steadily decline but then rises back up, creating a U-shape. The matches behavior seems to not have any discerning pattern.

```
# ANOVA for Likes
anova_likes_week <- aov(treatment_effect_likes ~ weeks_since_start, data = final_dataset)
summary(anova_likes_week)
```

```
##                    Df Sum Sq Mean Sq F value Pr(>F)
## weeks_since_start   1   16.7  16.706   2.751 0.0994 .
## Residuals         140  850.2   6.073
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# ANOVA for Matches
anova_matches_week <- aov(treatment_effect_matches ~ weeks_since_start, data = final_dataset)
summary(anova_matches_week)
```

```
##                    Df Sum Sq Mean Sq F value Pr(>F)
## weeks_since_start   1   0.82  0.8191   0.417  0.519
## Residuals         140 274.88  1.9634
```

Performing an anova shows a p-value of .0994 for likes and a high p-value of .519 for matches. Again similiar to the other HTE's, it implies that while there is some variation in likes attributable to the week since the account opened, this variation might be due to chance

12 Appendix

Here is a summary of the data columns:

- Owner: The team member who owns the profile.
- OwnerNum (Categorical): Owner variable transformed into an integer variable
- Profile_Name: The name associated with the profile.
- ProfileNum(Categorical): Profile Name variable transformed into an integer. We have 6 profile name.
- City: The city where the profile is located.

- CityNum:Integer of city, San Francisco = 0, NYC = 1
- Treat: Indicates whether the profile is in the control group or the treatment group.
- Treat_binary: Logical condition into binary values (0 for "Control" and 1 for "Treatment"). Indicator for the treatment group (humorous profile).
- Date: The date of the recorded data.
- Swipes: The number of times the profile was swiped.we carry out a daily random allocation of 10 swipes in the profiles, both in the control and treatment groups. This measure serves a dual purpose: first, it keeps our profiles actively participating in the app, and second, it facilitates interaction with other profiles, increasing the visibility of our profiles to other users.
- Likes: The number of likes received by the profile.
- ExpDate_Profile: Date variable transformed into an integer based on the day number in the experiment by profile. Starting from one.