

The effect of messages on vaccinations

Jake Bowers

August 17, 2021

Contents

Design	2
Data Setup	3
Weight Creation	5
Basic Descriptions and Visualization	7
Pre-specified analyses	11
RQ0: Is there any effect of condition assignment?	12
RQ1: Is there an effect of receiving a message as opposed to not receiving a message?	12
RQ2: Does any given message differ from control (focal tests)?	13
RQ3: Does epistemic humility help?	16
RQ5: How do social proof and appeals to the family interact?	16
RQ6: Did adaptive randomization increase vaccinations over fixed randomization?	17
Exploratory Analyses Not Pre-registered	18
Effects on vaccination within a week	18
Each iteration separately	20
Exploratory Analyses Pre-registered	22
EQ1: Do explicit appeals to the safety of vaccines increase responses in areas with higher proportions of Black or Latinx people? Message 2 vs. control	22
EQ2: Does the implication of choice through emphasis on a conspicuous advantage increase responses in areas with higher proportions of Republican people? Message 3 vs. control	24
EQ3: Do explicit appeals to ease of access increase responses in areas with higher proportions of Black or Latinx people? Message 5 vs. control	24
EQ4: Does epistemic humility increase responses in areas with higher proportions of either Black or Latinx people or Republican people? Message 4 versus 3	25
EQ5: Is there a day-of-week effect? Proportions of vaccinations collapsed across all messages by day.	27
EQ6: Is there an iteration effect? Some people were randomly assigned to have 3 weeks to schedule a vaccination and others only 1 week before the study ended. We explore whether there is a difference here.	27
Housekeeping	27

```

## The file expanding the aggregated data
wrkdat <- read_csv(file.path(DATA_DIR, "dat_indiv.csv"))
## This next makes an error in case the original data changed. Mostly to make us pay attention.
stopifnot(nrow(wrkdat) == 158103)

## The latest file from Kevin
wrkdat2 <- read_csv(file.path(DATA_DIR, "final_data_one_line_per_individual.csv"))
nrow(wrkdat2)

[1] 158103
stopifnot(nrow(wrkdat2) == 158103)
stopifnot(all(!is.na(wrkdat2$date_sent)))
## This next fails

wrkdat_design_tab <- with(wrkdat, table(date_sent, assigned_message, exclude = c()))
wrkdat2_design_tab <- with(wrkdat2, table(date_sent, assigned_message, exclude = c()))
stopifnot(all.equal(wrkdat_design_tab, wrkdat2_design_tab, exclude = c()))

wrkdat_outcomes_tab <- with(wrkdat, table(date_sent, is_vax_after_send))
wrkdat2_outcomes_tab <- with(wrkdat2, table(date_sent, is_vax_after_send))
stopifnot(all.equal(wrkdat_design_tab, wrkdat2_design_tab, exclude = c()))

## Add ZCTA data to wrkdat2:

zctadat <- read_csv(file.path(DEMO_DATA_DIR, "combined_demo_data_by_zcta.csv"))

Warning: Missing column names filled in: 'X1' [1]
## The first column, auto-named to X1 is noise
zctadat$X1 <- NULL

zctadat <- zctadat %>% mutate(
  pct_any_blk = any_black_population / total_population,
  pct_hisp = hispanic_population / total_population,
  pct_dem = dem_vote / total_vote,
  pct_gop = gop_vote / total_vote,
  pct_vote = total_vote / total_population
)

## Any mismatch in ZCTA ids?
## using https://stackoverflow.com/questions/19797954/function-to-find-symmetric-difference-opposite-of-intersection-in-r
sym_diff2 <- function(a, b) unique(c(setdiff(a, b), setdiff(b, a)))
sym_diff2(unique(wrkdat2$zcta), unique(zctadat$zcta))

[1] "00000"

## So some people with no ZCTA. We will address this in the supplementary analyses later
wrkdat3 <- left_join(wrkdat2, zctadat, by = "zcta")
stopifnot(nrow(wrkdat3) == nrow(wrkdat2))
## We will use wrkdat3 for the rest of the time

```

Design

This study randomly assigned 8 sms message types plus 1 control arm (no msg) (9 arms total) to roughly 160,000 (exactly 158,103) Rhode Islanders and recorded whether or not these people were vaccinated during the study period.

The randomization occurred each week and then, within arm, people were randomly assigned to a day on which they could be sent a text message. This nested randomization can be represented as complete randomization to one of 9 arms within each active day of the study. The table below, created by `with(wrkdat3, table(date_sent, assigned_message, exclude = c()))`, shows a pattern of assignment by day consistent with this idea — roughly equal

numbers assigned per message condition per day within each iteration.

The experiment also involve adaptive randomization, and the Thompson sampling algorithm assigned more people to arm 6 in the final week, for example. A block-randomized experiment can often have variation in probabilities assigned to treatment, and the analysis of such an experiment thus is no different whether there are changes in assignment probabilities between blocks or not.

Data Setup

We drop observations assigned to be sent a message on June 15, 2021 because those messages were not sent and because people were assigned to that date at random.

```
wrkdat3 <- wrkdat3 %>%
  filter(date_sent < "2021-06-15") %>%
  droplevels()
stopifnot(nrow(wrkdat3) == 142428)

## Some recoding to make things nicer for coin etc..
## Also coin wants factor variables for the CMH tests (since those are test of independence of contingency tables)
wrkdat3$messageF <- factor(wrkdat3$assigned_message)
wrkdat3$vaccinated <- as.numeric(wrkdat3$is_vax_after_send)
wrkdat3$vaccinatedF <- factor(wrkdat3$vaccinated)

wrkdat3$vac_in_week <- as.numeric(wrkdat3$is_within_one_week_after_send)
wrkdat3$vac_in_weekF <- factor(wrkdat3$vac_in_week)

## Checking that vaccinated in week is a subset of vaccinated
with(wrkdat3, table(vac_in_weekF, vaccinatedF, exclude = c()))
```

	vaccinatedF	
vac_in_weekF	0	1
0	139684	1453
1	0	1291

```
wrkdat3$date_sentF <- factor(wrkdat3$date_sent)
## A new variable that records the "any message" versus "no message" contrast
wrkdat3$not_control <- as.numeric(wrkdat3$messageF != "message_0")
with(wrkdat3, table(messageF, not_control, exclude = c()))
```

	not_control	
messageF	0	1
message_0	11327	0
message_1	0	10491
message_2	0	12440
message_3	0	11962
message_4	0	10110
message_5	0	15243
message_6	0	47058
message_7	0	12363
message_8	0	11434

```
wrkdat3$not_controlF <- factor(wrkdat3$not_control)
```

```
## This next does not involve all possible dates, only those existing in the data:
unique(wrkdat3$date_sent)
```

```
[1] "2021-06-03" "2021-06-08" "2021-05-26" "2021-06-11" "2021-06-10" "2021-06-02" "2021-05-28" "2021-06-14" "2021-06-04"
[10] "2021-05-27" "2021-06-09" "2021-06-07" "2021-05-25"
```

```
## Any missing dates will be assigned NA and the code will stop if any NA are detected
```

```
wrkdat3 <- wrkdat3 %>% mutate(iteration = case_when(
  date_sent <= "2021-05-28" ~ 1,
  date_sent >= "2021-06-02" & date_sent <= "2021-06-08" ~ 2,
  date_sent > "2021-06-08" ~ 3
```

```

))
stopifnot(any(!is.na(wrkdat3$iteration)))
## Inspect by hand
with(wrkdat3, table(date_sent, iteration, exclude = c()))

```

	iteration		
date_sent	1	2	3
2021-05-25	10003	0	0
2021-05-26	9999	0	0
2021-05-27	9999	0	0
2021-05-28	9999	0	0
2021-06-02	0	7941	0
2021-06-03	0	7941	0
2021-06-04	0	7948	0
2021-06-07	0	7942	0
2021-06-08	0	7937	0
2021-06-09	0	0	15682
2021-06-10	0	0	15679
2021-06-11	0	0	15679
2021-06-14	0	0	15679

Notice that assignment is consistent with complete randomization in iteration 1 (equal numbers assigned to each message that week). It diverges from uniform assignment in iteration 2 and 3 because we are using the ε -Thompson adaptive algorithm for assignment. Notice that within iteration, roughly equal numbers are allocated to each day within message type. This was also done at random, making this a study that can be treated as if it were block-randomized by day.

```

with(wrkdat3, table(iteration, assigned_message, exclude = c()))

```

	assigned_message								
iteration	message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
1	4445	4445	4445	4445	4444	4444	4444	4444	4444
2	3494	3069	5647	5053	2414	8502	1990	5591	3949
3	3388	2977	2348	2464	3252	2297	40624	2328	3041

```

with(wrkdat3, table(date_sentF, assigned_message, exclude = c()))

```

	assigned_message								
date_sentF	message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
2021-05-25	1112	1112	1112	1112	1111	1111	1111	1111	1111
2021-05-26	1111	1111	1111	1111	1111	1111	1111	1111	1111
2021-05-27	1111	1111	1111	1111	1111	1111	1111	1111	1111
2021-05-28	1111	1111	1111	1111	1111	1111	1111	1111	1111
2021-06-02	699	616	1131	1010	485	1698	398	1117	787
2021-06-03	697	611	1131	1010	481	1700	399	1122	790
2021-06-04	699	616	1132	1011	485	1701	397	1118	789
2021-06-07	698	612	1124	1011	482	1704	398	1121	792
2021-06-08	701	614	1129	1011	481	1699	398	1113	791
2021-06-09	847	745	587	616	813	575	10156	582	761
2021-06-10	847	744	587	616	813	574	10156	582	760
2021-06-11	847	744	587	616	813	574	10156	582	760
2021-06-14	847	744	587	616	813	574	10156	582	760

```

## Message types
## 0. Control
## 1. Ownership (baseline prompt)
## 2. Safety
## 3. Pros of vaccination (implicit choice): no hospitals
## 4. Epistemic humility + pros of vaccination (implicit choice): no hospitals
## 5. Access
## 6. Family concern
## 7. Social proof
## 8. Social proof + family concern

```

```

with(wrkdat3, table(iteration, exclude = FALSE))

```

```

iteration
  1    2    3
40000 39709 62719
with(wrkdat3, table(date_sent, exclude = FALSE))

date_sent
2021-05-25 2021-05-26 2021-05-27 2021-05-28 2021-06-02 2021-06-03 2021-06-04 2021-06-07 2021-06-08 2021-06-09 2021-06-10 2021-06-11
      10003         9999         9999         9999         7941         7941         7948         7942         7937         15682         15679         15679
2021-06-14
      15679

```

Weight Creation

A block-randomized study is a collection of mini-experiments. Overall tests and estimates involve some kind of combination of those block-level quantities, using weights to give larger and more informative blocks more weight and/or to target specific kinds of treatment effects. We pre-specified that we would use block-size weights because we know that these produce unbiased estimators. Right now, the `difference_in_means` command will use those estimators by default, but only for a comparison of two arms. To make things easier later on, we therefore create weights here to be used in our standard linear regression command (`lm_robust`) that we use for general estimation in randomized studies (because, in part, it uses randomization justified standard errors by default, and thus allows less typing than `lm`).

Here we add weights to the data set since each block (`date_sent` involved different assignments to treatment (actually it was each `iteration` but the sms were then divided at random into days for sending and we are then treating each moment of administering the treatment as a block).

I'm doing this slow to convince myself, and perhaps, others that (1) there are different ways to weight blocks and (2) that what we are seeing from the canned R commands with weights makes sense. During the weight creation we will be doing some analyses of the effects of the study, they are mainly to help us ensure that we creating the weights correctly. A cleaner analysis is below, under "Pre-specified analyses".

```

## The two arm version:
## Creating the weights following the examples in the randomizr vignettes
block_m_each_bin <- with(wrkdat3, table(date_sentF, not_control, exclude = c()))
declared_randomization_twoarm <- declare_ra(blocks = wrkdat3$date_sentF, block_m = block_m_each_bin[, "1"])
wrkdat3$IPW_weight_bin <- 1 / obtain_condition_probabilities(declaration = declared_randomization_twoarm, assignment = wrkdat3$not_control)
## unique(wrkdat3$IPW_weight_bin)
## Now doing this by hand, following Gerber and Green Chap 3 (creating regression weights to reflect block-size weighting)
wrkdat3 <- wrkdat3 %>%
  group_by(date_sentF) %>%
  mutate(
    nb = n(),
    p_not_control = mean(not_control),
    nbwt_bin = ifelse(not_control == 1, 1 / p_not_control, 1 / (1 - p_not_control)),
  ) %>%
  ungroup()

stopifnot(all.equal(wrkdat3$IPW_weight_bin, wrkdat3$nbwt_bin))

lm_bin0 <- difference_in_means(vaccinated ~ not_control, blocks = date_sentF, data = wrkdat3)
lm_bin1 <- lm_robust(vaccinated ~ not_control, data = wrkdat3, weights = nbwt_bin)
lm_bin2 <- lm_robust(vaccinated ~ not_control, data = wrkdat3, weights = IPW_weight_bin)
lm_bin1

```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.02061	0.001312	15.711	1.406e-55	0.018038	0.023180	142426
not_control	-0.00147	0.001366	-1.077	2.817e-01	-0.004147	0.001207	142426

```
stopifnot(all.equal(lm_bin0$coef, lm_bin1$coef["not_control"]))
stopifnot(all.equal(lm_bin0$coef, lm_bin2$coef["not_control"]))

## Now for the precision weighted version
## lm_bin3 <- lm_robust(vaccinated~not_control,fixed_effects=~date_sentF,data=wrkdat3)
## lm_bin3
```

We cannot use difference_of_means for a multi-armed treatment, but we follow the same general approach:

```
### Multiple arm version
block_m_each <- with(wrkdat3, table(date_sentF, messageF, exclude = c()))
block_prob_each <- block_m_each / rowSums(block_m_each)
declared_randomization_multarm <- declare_ra(blocks = wrkdat3$date_sentF, block_m_each = block_m_each, conditions = sort(unique(
wrkdat3$IPW_weight_multarm <- 1 / obtain_condition_probabilities(declaration = declared_randomization_multarm, assignment = wrkdat3$
## unique(wrkdat3$IPW_weight_multarm)

stopifnot(all.equal(sort(unique(1 / wrkdat3$IPW_weight_multarm)), sort(unique(block_prob_each))))

lm_multarm_ipw <- lm_robust(vaccinated ~ messageF, data = wrkdat3, weights = IPW_weight_multarm)
## lm1_mult_fe <- lm_robust(vaccinated~messageF,data=wrkdat3,fixed_effects=~date_sentF)
lm_multarm_ipw
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.02060861	0.001312	15.71145	1.406e-55	0.018038	0.0231795	142419
messageFmessage_1	-0.00208199	0.001842	-1.13002	2.585e-01	-0.005693	0.0015291	142419
messageFmessage_2	-0.00002266	0.001898	-0.01194	9.905e-01	-0.003742	0.0036968	142419
messageFmessage_3	0.00232052	0.002012	1.15339	2.488e-01	-0.001623	0.0062638	142419
messageFmessage_4	-0.00222375	0.001863	-1.19395	2.325e-01	-0.005874	0.0014267	142419
messageFmessage_5	0.00015034	0.001886	0.07972	9.365e-01	-0.003546	0.0038467	142419
messageFmessage_6	-0.00260175	0.001719	-1.51333	1.302e-01	-0.005971	0.0007679	142419
messageFmessage_7	-0.00316289	0.001824	-1.73444	8.284e-02	-0.006737	0.0004113	142419
messageFmessage_8	-0.00012584	0.001844	-0.06826	9.456e-01	-0.003739	0.0034876	142419

```
## lm1_mult_fe
```

```
## Creating the weights by hand to verify understanding:
wrkdat3 <- wrkdat3 %>%
  group_by(date_sentF) %>%
  mutate(
    p_m_0 = mean(messageF == "message_0"),
    p_m_1 = mean(messageF == "message_1"),
    p_m_2 = mean(messageF == "message_2"),
    p_m_3 = mean(messageF == "message_3"),
    p_m_4 = mean(messageF == "message_4"),
    p_m_5 = mean(messageF == "message_5"),
    p_m_6 = mean(messageF == "message_6"),
    p_m_7 = mean(messageF == "message_7"),
    p_m_8 = mean(messageF == "message_8"),
    nbwt_mult = as.numeric(messageF == "message_0") / p_m_0 +
      as.numeric(messageF == "message_1") / p_m_1 +
      as.numeric(messageF == "message_2") / p_m_2 +
      as.numeric(messageF == "message_3") / p_m_3 +
      as.numeric(messageF == "message_4") / p_m_4 +
      as.numeric(messageF == "message_5") / p_m_5 +
      as.numeric(messageF == "message_6") / p_m_6 +
      as.numeric(messageF == "message_7") / p_m_7 +
      as.numeric(messageF == "message_8") / p_m_8
  ) %>%
  ungroup()
## Verify that the IPW weights using randomizr are the same as those we created by hand
stopifnot(all.equal(sort(unique(wrkdat3$IPW_weight_multarm)), sort(unique(wrkdat3$nbwt_mult))))
```

Now estimate effects by first aggregating to the block level and then weighting (this is just to check that we can get the same numbers as when we use lm_robust etc.):

```

wrkdat3_b <- wrkdat3 %>%
  group_by(date_sentF) %>%
  summarize(
    nb = n(),
    effect_1 = mean(vaccinated[messageF == "message_1"]) - mean(vaccinated[messageF == "message_0"]),
    effect_2 = mean(vaccinated[messageF == "message_2"]) - mean(vaccinated[messageF == "message_0"]),
    effect_3 = mean(vaccinated[messageF == "message_3"]) - mean(vaccinated[messageF == "message_0"]),
    effect_4 = mean(vaccinated[messageF == "message_4"]) - mean(vaccinated[messageF == "message_0"]),
    effect_5 = mean(vaccinated[messageF == "message_5"]) - mean(vaccinated[messageF == "message_0"]),
    effect_6 = mean(vaccinated[messageF == "message_6"]) - mean(vaccinated[messageF == "message_0"]),
    effect_7 = mean(vaccinated[messageF == "message_7"]) - mean(vaccinated[messageF == "message_0"]),
    effect_8 = mean(vaccinated[messageF == "message_8"]) - mean(vaccinated[messageF == "message_0"]),
    effect_any_msg = mean(vaccinated[messageF != "message_0"]) - mean(vaccinated[messageF == "message_0"])
  )
## The ATE over all is just the weighted average of the block ATEs
simp_est <- wrkdat3_b %>% summarize(across(
  .cols = contains("effect"),
  .fns = function(x) {
    weighted.mean(x, w = nb)
  }
))
## Test the block-based algorithm itself
stopifnot(all.equal(with(wrkdat3_b, weighted.mean(x = effect_1, w = nb)), simp_est$effect_1))
stopifnot(all.equal(with(wrkdat3_b, weighted.mean(x = effect_6, w = nb)), simp_est$effect_6))

## Verifying that we get the same answer with explicit weighting (building up from the block-level estimates)
## versus regression weights
## The "any message" effect differs slightly but everything else is identical
rbind(by_hand = simp_est, by_lm = lm_multarm_ipw$coef[-1])

# A tibble: 2 x 9
  effect_1 effect_2 effect_3 effect_4 effect_5 effect_6 effect_7 effect_8 effect_any_msg
*   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
1 -0.00208 -0.0000227 0.00232 -0.00222 0.000150 -0.00260 -0.00316 -0.000126 -0.00147
2 -0.00208 -0.0000227 0.00232 -0.00222 0.000150 -0.00260 -0.00316 -0.000126 -0.00208

```

Ok. So, I'm satisfied with the regression weights as doing their job, and we can move on to basic descriptives and the pre-registered analyses using tools like `lm_robust`. Weighting the different blocks is implicit (and optimal from a statistical testing perspective) in `cmh_test`.

Basic Descriptions and Visualization

Here is a rough plot to show the proportions vaccinated by message by date plus binomial 95% confidence intervals for those proportions. These are not effects, just proportions. Each panel shows a gray vertical line at the proportion vaccinated in the control condition.

mtype	mtypeF							
	Control	Ownership	Safety	Preventing bad outcomes	Epistemic humility+no bad outcomes	Access		
Access	0	0	0		0	0	13	
Control	13	0	0		0	0	0	
Epistemic humility+no bad outcomes	0	0	0		0	13	0	
Family concern	0	0	0		0	0	0	
Ownership	0	13	0		0	0	0	
Preventing bad outcomes	0	0	0	13		0	0	
Safety	0	0	13	0		0	0	
Social proof	0	0	0	0		0	0	
Social proof+family concern	0	0	0	0		0	0	

mtype	mtypeF		
	Family concern	Social proof	Social proof+family concern
Access	0	0	0
Control	0	0	0
Epistemic humility+no bad outcomes	0	0	0
Family concern	13	0	0
Ownership	0	0	0

Preventing bad outcomes	0	0	0
Safety	0	0	0
Social proof	0	13	0
Social proof+family concern	0	0	13

		messageF								
mtype		message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
Access		0	0	0	0	0	13	0	0	0
Control		13	0	0	0	0	0	0	0	0
Epistemic humility+no bad outcomes		0	0	0	0	13	0	0	0	0
Family concern		0	0	0	0	0	0	13	0	0
Ownership		0	13	0	0	0	0	0	0	0
Preventing bad outcomes		0	0	0	13	0	0	0	0	0
Safety		0	0	13	0	0	0	0	0	0
Social proof		0	0	0	0	0	0	0	13	0
Social proof+family concern		0	0	0	0	0	0	0	0	13

		messageF								
mtypeF		message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
Control		13	0	0	0	0	0	0	0	0
Ownership		0	13	0	0	0	0	0	0	0
Safety		0	0	13	0	0	0	0	0	0
Preventing bad outcomes		0	0	0	13	0	0	0	0	0
Epistemic humility+no bad outcomes		0	0	0	0	13	0	0	0	0
Access		0	0	0	0	0	13	0	0	0
Family concern		0	0	0	0	0	0	13	0	0
Social proof		0	0	0	0	0	0	0	13	0
Social proof+family concern		0	0	0	0	0	0	0	0	13

Proportion vaccinated by date and message arm (these are the point values in the above plot):

```
prop_vac_arr <- matrix(plotdat1$prop_vac,
  nrow = 13, byrow = TRUE,
  dimnames = list(sort(unique(as.character(plotdat1$date_sent))), sort(unique(plotdat1$messageF)))
)

zapsmall(prop_vac_arr, digits = 2)
```

	message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
2021-05-25	0.045	0.032	0.046	0.045	0.029	0.055	0.031	0.027	0.037
2021-05-26	0.037	0.047	0.039	0.032	0.032	0.026	0.028	0.027	0.032
2021-05-27	0.027	0.023	0.038	0.030	0.029	0.030	0.033	0.030	0.032
2021-05-28	0.034	0.029	0.025	0.025	0.029	0.045	0.031	0.021	0.041
2021-06-02	0.020	0.021	0.026	0.027	0.025	0.016	0.020	0.020	0.025
2021-06-03	0.014	0.020	0.022	0.021	0.027	0.017	0.025	0.020	0.027
2021-06-04	0.017	0.013	0.017	0.018	0.023	0.016	0.015	0.023	0.020
2021-06-07	0.013	0.008	0.008	0.014	0.012	0.011	0.008	0.015	0.020
2021-06-08	0.027	0.021	0.013	0.015	0.017	0.019	0.018	0.018	0.016
2021-06-09	0.014	0.011	0.017	0.016	0.012	0.028	0.013	0.022	0.018
2021-06-10	0.011	0.011	0.009	0.021	0.011	0.014	0.012	0.014	0.011
2021-06-11	0.014	0.013	0.014	0.019	0.012	0.005	0.009	0.003	0.007
2021-06-14	0.011	0.008	0.010	0.019	0.004	0.002	0.008	0.003	0.005

Can we see any patterns in the extent to which any given message, sent any given day, elicited more vaccinations than control? Not really, each message was, in absolute terms, sometimes better than and sometimes worse than control. No message stands out for always being worse or better than control (and notice this little table weighs all days equally).

```
prop_diffs <- apply(prop_vac_arr, 2, function(x) {
  x - prop_vac_arr[, 1]
})

prop_diffs
```

	message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
2021-05-25	0	-0.0125899	0.0008993	0.0000000	-0.0161611	0.0099415	-0.01436097	-0.017961	-0.00806034
2021-05-26	0	0.0099010	0.0018002	-0.0045005	-0.0054005	-0.0108011	-0.00900090	-0.009901	-0.00540054
2021-05-27	0	-0.0045005	0.0108011	0.0027003	0.0018002	0.0027003	0.00630063	0.002700	0.00540054


```

2021-05-28      0 -0.0054005 -0.0090009 -0.0090009 -0.0054005  0.0108011 -0.00360036 -0.013501  0.00630063
2021-06-02      0  0.0010753  0.0056124  0.0067041  0.0047137 -0.0041276  0.00007189 -0.000333  0.00538435
2021-06-03      0  0.0052927  0.0077571  0.0064449  0.0126798  0.0027116  0.01071545  0.006152  0.01223508
2021-06-04      0 -0.0041804 -0.0003829  0.0006368  0.0055130 -0.0007065 -0.00205403  0.006088  0.00311145
2021-06-07      0 -0.0047240 -0.0048869  0.0009537 -0.0004459 -0.0017437 -0.00535629  0.002271  0.00730804
2021-06-08      0 -0.0059315 -0.0138180 -0.0122673 -0.0104721 -0.0076809 -0.00951620 -0.009135 -0.01066924
2021-06-09      0 -0.0034294  0.0028681  0.0020661 -0.0018675  0.0136584 -0.00067809  0.008169  0.00422920
2021-06-10      0  0.0001270 -0.0021079  0.0104782  0.0004444  0.0033115  0.00128840  0.003120 -0.00009942
2021-06-11      0 -0.0007268 -0.0005390  0.0053129 -0.0018675 -0.0089412 -0.00491204 -0.010731 -0.00758870
2021-06-14      0 -0.0025612 -0.0004043  0.0088548 -0.0069357 -0.0088836 -0.00304401 -0.007189 -0.00536258

```

```
signs_prop_diffs <- sign(prop_diffs)
```

```
signs_prop_diffs
```

```

      message_0 message_1 message_2 message_3 message_4 message_5 message_6 message_7 message_8
2021-05-25      0      -1       1       0       -1       1       -1       -1       -1
2021-05-26      0       1       1      -1      -1      -1      -1      -1      -1
2021-05-27      0      -1       1       1       1       1       1       1       1
2021-05-28      0      -1      -1      -1      -1       1      -1      -1       1
2021-06-02      0       1       1       1       1      -1       1      -1       1
2021-06-03      0       1       1       1       1       1       1       1       1
2021-06-04      0      -1      -1       1       1      -1      -1       1       1
2021-06-07      0      -1      -1       1      -1      -1      -1       1       1
2021-06-08      0      -1      -1      -1      -1      -1      -1      -1      -1
2021-06-09      0      -1       1       1      -1       1      -1       1       1
2021-06-10      0       1      -1       1       1       1       1       1      -1
2021-06-11      0      -1      -1       1      -1      -1      -1      -1      -1
2021-06-14      0      -1      -1       1      -1      -1      -1      -1      -1

```

```
## Number of times greater than control
```

```

gt_control <- apply(signs_prop_diffs, 2, function(x) {
  sum(x > 0)
})
lt_control <- apply(signs_prop_diffs, 2, function(x) {
  sum(x < 0)
})

```

```
rbind(gt_control, lt_control)
```

```

      message_0 message_1 message_2 message_3 message_4 message_5 message_6 message_7 message_8
gt_control      0       4       6       9       5       6       4       6       7
lt_control      0       9       7       3       8       7       9       7       6

```

What about patterns in the extent to which one message was higher ranked in a given day? Not really, the average rank out of 9 messages is about 4,5 or 6 for each message across the days. The number of times that a message is lowest or highest ranked is not huge (out of 13 days): like less than 3 days out of 13 does one message appear worst or best.

```

prop_rank <- t(apply(prop_vac_arr, 1, function(x) {
  rank(x)
})))

```

```
prop_rank
```

```

      message_0 message_1 message_2 message_3 message_4 message_5 message_6 message_7 message_8
2021-05-25      6.5      4.0      8.0      6.5      2.0       9       3       1      5.0
2021-05-26      7.0      9.0      8.0      6.0      4.5       1       3       2      4.5
2021-05-27      2.0      1.0      9.0      5.0      3.0       5       8       5      7.0
2021-05-28      7.0      4.5      2.5      2.5      4.5       9       6       1      8.0
2021-06-02      3.0      5.0      8.0      9.0      6.0       1       4       2      7.0
2021-06-03      1.0      3.0      6.0      5.0      9.0       2       7       4      8.0
2021-06-04      5.0      1.0      4.0      6.0      8.0       3       2       9      7.0
2021-06-07      6.0      3.0      2.0      7.0      5.0       4       1       8      9.0
2021-06-08      9.0      8.0      1.0      2.0      4.0       7       5       6      3.0
2021-06-09      4.0      1.0      6.0      5.0      2.0       9       3       8      7.0
2021-06-10      3.0      4.0      1.0      9.0      5.0       8       6       7      2.0
2021-06-11      8.0      6.0      7.0      9.0      5.0       2       4       1      3.0

```

2021-06-14	8.0	6.0	7.0	9.0	3.0	1	5	2	4.0
------------	-----	-----	-----	-----	-----	---	---	---	-----

```
apply(prop_rank, 2, mean)
```

message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
5.346	4.269	5.346	6.231	4.692	4.692	4.385	4.308	5.731

```
apply(prop_rank, 2, function(x) {
  sum(x == min(x))
})
```

message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
1	3	2	1	2	3	1	3	1

```
apply(prop_rank, 2, function(x) {
  sum(x == max(x))
})
```

message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
1	1	1	4	1	3	1	1	1

As a reminder about the sample sizes in each message and day:

```
with(wrkdatt3, table(date_sent, messageF, exclude = c()))
```

	messageF								
date_sent	message_0	message_1	message_2	message_3	message_4	message_5	message_6	message_7	message_8
2021-05-25	1112	1112	1112	1112	1111	1111	1111	1111	1111
2021-05-26	1111	1111	1111	1111	1111	1111	1111	1111	1111
2021-05-27	1111	1111	1111	1111	1111	1111	1111	1111	1111
2021-05-28	1111	1111	1111	1111	1111	1111	1111	1111	1111
2021-06-02	699	616	1131	1010	485	1698	398	1117	787
2021-06-03	697	611	1131	1010	481	1700	399	1122	790
2021-06-04	699	616	1132	1011	485	1701	397	1118	789
2021-06-07	698	612	1124	1011	482	1704	398	1121	792
2021-06-08	701	614	1129	1011	481	1699	398	1113	791
2021-06-09	847	745	587	616	813	575	10156	582	761
2021-06-10	847	744	587	616	813	574	10156	582	760
2021-06-11	847	744	587	616	813	574	10156	582	760
2021-06-14	847	744	587	616	813	574	10156	582	760

A figure with proportion vaccinated at all, and another with proportion vaccinated within a week.

```
plotdat1$line_thick <- ifelse(plotdat1$messageF == "message_0", 2, 1)
plotdat1$control_msg <- plotdat1$messageF == "message_0"

g_prop_vac <- ggplot(plotdat1, aes(x = date_sent, y = prop_vac, group = mtypeF, color = mtypeF, size = control_msg)) +
  geom_point() +
  geom_line() +
  guides(color = guide_legend(title = "Message")) +
  scale_colour_brewer(type = "div") +
  ylab("Vaccinated by June 22") +
  xlab("Date Assigned Message") +
  scale_size_manual(values = c(0.5, 1.2), guide = "none") +
  scale_linetype_manual(values = c("solid", "dashed"), guide = "none") +
  theme_classic(base_family = "Open Sans") +
  theme(
    text = element_text(size = 16),
    axis.text.x = element_text(angle = 0, hjust = 1),
    legend.position = c(.6, 0.8)
  )

# g_prop_vac

g_prop_vac_smooth <- ggplot(plotdat1, aes(x = date_sent, y = prop_vac, group = mtypeF, color = mtypeF, size = control_msg)) +
  geom_point() +
  geom_smooth(se = FALSE, method = "loess", span = 2 / 3, method.args = list(degree = 1, family = "symmetric")) +
  guides(color = guide_legend(title = "Message")) +
  scale_colour_brewer(type = "div") +
```

```

ylab("Vaccinated by June 22") +
xlab("Date Assigned Message") +
scale_size_manual(values = c(0.5, 1.2), guide = "none") +
scale_linetype_manual(values = c("solid", "dashed"), guide = "none") +
theme_classic(base_family = "Open Sans") +
theme(
  text = element_text(size = 16),
  axis.text.x = element_text(angle = 0, hjust = 1),
  legend.position = c(.6, 0.8)
)

# g_prop_vac_smooth

## Trying to break lines in between iterations. Not working well.
## blah <- tidyrr::complete(plotdat1, date_sent = seq(min(date_sent), max(date_sent), by = "day"))

g_prop_vac_in_week <- ggplot(plotdat1, aes(x = date_sent, y = prop_vac_in_week, group = mtypeF, color = mtypeF, size = control_m)) +
  geom_point() +
  geom_path() +
  # facet_wrap(~iteration, scales="free")+
  guides(color = guide_legend(title = "Message")) +
  scale_colour_brewer(type = "div") +
  ylab("Vaccinated within a Week of Message Assignment") +
  xlab("Date Assigned Message") +
  scale_size_manual(values = c(0.5, 1.2), guide = "none") +
  scale_linetype_manual(values = c("solid", "dashed"), guide = "none") +
  theme_classic(base_family = "Open Sans") +
  theme(
    text = element_text(size = 16),
    axis.text.x = element_text(angle = 0, hjust = 1)
  )

# g_prop_vac_in_week

g_prop_vac_smooth_in_week <- ggplot(plotdat1, aes(x = date_sent, y = prop_vac_in_week, group = mtypeF, color = mtypeF, size = control_m)) +
  geom_point() +
  geom_smooth(se = FALSE, method = "loess", span = 2 / 3, method.args = list(degree = 1, family = "symmetric")) +
  guides(color = guide_legend(title = "Message")) +
  ylab("Vaccinated within a Week of Message Assignment") +
  xlab("Date Assigned Message") +
  # geom_vline(xintercept=as.Date(c("2021-05-31", "2021-06-08")))+
  scale_size_manual(values = c(0.5, 1.2), guide = "none") +
  scale_colour_brewer(type = "div") +
  scale_linetype_manual(values = c("solid", "dashed"), guide = "none") +
  theme_classic(base_family = "Open Sans") +
  theme(
    text = element_text(size = 16),
    axis.text.x = element_text(angle = 0, hjust = 1)
  )

# g_prop_vac_smooth_in_week

ggsave(file = "prop_vac.png", path = OUTPUT_DIR, plot = g_prop_vac, type = "cairo-png", dpi = 300)
ggsave(file = "prop_vac_in_week.png", path = OUTPUT_DIR, plot = g_prop_vac_in_week, type = "cairo-png", dpi = 300)
ggsave(file = "prop_vac_smooth.png", path = OUTPUT_DIR, plot = g_prop_vac_smooth, type = "cairo-png", dpi = 300)
ggsave(file = "prop_vac_smooth_in_week.png", path = OUTPUT_DIR, plot = g_prop_vac_smooth_in_week, type = "cairo-png", dpi = 300)

```

Pre-specified analyses

These analyses were registered at <https://osf.io/pkhae/>.

RQ0: Is there any effect of condition assignment?

The following suggests that we have some evidence of differences among the messages:

```
## This is the asymptotic approx to the randomization inference
rq0_asym <- cmh_test(vaccinatedF ~ messageF | date_sentF, data = wrkdat3, distribution = asymptotic())
## This next is the permutation approx to the randomization inference
set.seed(12345)
rq0_perm <- cmh_test(vaccinatedF ~ messageF | date_sentF, data = wrkdat3, distribution = approximate(nresample = 10000, parallel = TRUE))
rq0_asym
```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```
data: vaccinatedF by
      messageF (message_0, message_1, message_2, message_3, message_4, message_5, message_6, message_7, message_8)
      stratified by date_sentF
chi-squared = 13, df = 8, p-value = 0.1
rq0_perm
```

Approximative Generalized Cochran-Mantel-Haenszel Test

```
data: vaccinatedF by
      messageF (message_0, message_1, message_2, message_3, message_4, message_5, message_6, message_7, message_8)
      stratified by date_sentF
chi-squared = 13, p-value = 0.1
pvalue(rq0_asym)
```

```
[1] 0.1153
thetab <- with(wrkdat3, table(messageF, vaccinatedF, date_sentF))

rq0a <- mantelhaen.test(thetab)
rq0a
```

Cochran-Mantel-Haenszel test

```
data: thetab
Cochran-Mantel-Haenszel M^2 = 13, df = 8, p-value = 0.1
```

RQ1: Is there an effect of receiving a message as opposed to not receiving a message?

The below shows little evidence of effect of “any message” versus “control”.

```
rq1_asym <- cmh_test(vaccinatedF ~ not_controlF | date_sentF, data = wrkdat3, distribution = asymptotic())
rq1_perm <- cmh_test(vaccinatedF ~ not_controlF | date_sentF, data = wrkdat3, distribution = approximate(nresample = 10000, parallel = TRUE))
rq1_asym
```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```
data: vaccinatedF by not_controlF (0, 1)
      stratified by date_sentF
chi-squared = 1.2, df = 1, p-value = 0.3
rq1_perm
```

Approximative Generalized Cochran-Mantel-Haenszel Test

```
data: vaccinatedF by not_controlF (0, 1)
      stratified by date_sentF
chi-squared = 1.2, p-value = 0.3
pvalue(rq1_asym)
```

```
[1] 0.2663
```

We can show the estimated difference in proportion here:

```
rq1_est <- difference_in_means(vaccinated ~ not_controlF, blocks = date_sentF, data = wrkdat3)
rq1_est
```

Design: Blocked

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
not_controlF	-0.00147	0.001363	-1.079	0.2806	-0.004141	0.001201	142402

```
rq1_week_est <- difference_in_means(vaccinated ~ not_controlF, blocks = iteration, data = wrkdat3)
rq1_week_est
```

Design: Blocked

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
not_controlF	-0.001473	0.001363	-1.081	0.2797	-0.004145	0.001198	142422

```
lm_bin0 <- difference_in_means(vaccinated ~ not_control, blocks = date_sentF, data = wrkdat3)
lm_bin1 <- lm_robust(vaccinated ~ not_control, data = wrkdat3, weights = nbwt_bin)
lm_bin2 <- lm_robust(vaccinated ~ not_control, data = wrkdat3, weights = IPW_weight_bin)
lm_bin1
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.02061	0.001312	15.711	1.406e-55	0.018038	0.023180	142426
not_control	-0.00147	0.001366	-1.077	2.817e-01	-0.004147	0.001207	142426

```
stopifnot(all.equal(lm_bin0$coef, lm_bin1$coef["not_control"]))
stopifnot(all.equal(lm_bin0$coef, lm_bin2$coef["not_control"]))
stopifnot(all.equal(lm_bin0$coef[[1]], rq1_est$coef[["not_controlF"]]))
```

```
## Proportion vaccinated by end in the "not control" combination condition
sum(coef(lm_bin1))
```

```
[1] 0.01914
```

RQ2: Does any given message differ from control (focal tests)?

Overall, we have approx 2% of the control group getting vaccinated (weighted average across the days), and very small differences from that rate for each message — all less than 1/3 pct point different in magnitude from the control group.

```
rq2_est <- lm_robust(vaccinated ~ messageF, weights = IPW_weight_multarm, data = wrkdat3)
rq2_est
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.02060861	0.001312	15.71145	1.406e-55	0.018038	0.0231795	142419
messageFmessage_1	-0.00208199	0.001842	-1.13002	2.585e-01	-0.005693	0.0015291	142419
messageFmessage_2	-0.00002266	0.001898	-0.01194	9.905e-01	-0.003742	0.0036968	142419
messageFmessage_3	0.00232052	0.002012	1.15339	2.488e-01	-0.001623	0.0062638	142419
messageFmessage_4	-0.00222375	0.001863	-1.19395	2.325e-01	-0.005874	0.0014267	142419
messageFmessage_5	0.00015034	0.001886	0.07972	9.365e-01	-0.003546	0.0038467	142419
messageFmessage_6	-0.00260175	0.001719	-1.51333	1.302e-01	-0.005971	0.0007679	142419
messageFmessage_7	-0.00316289	0.001824	-1.73444	8.284e-02	-0.006737	0.0004113	142419
messageFmessage_8	-0.00012584	0.001844	-0.06826	9.456e-01	-0.003739	0.0034876	142419

```
## In percentage point differences from message_0 (except for Intercept which is proportion vaccinated (on average, weighted by
zapsmall(rq2_est$coef * 100)
```

	(Intercept)	messageFmessage_1	messageFmessage_2	messageFmessage_3	messageFmessage_4	messageFmessage_5	messageFmessage_6
	2.0609	-0.2082	-0.0023	0.2321	-0.2224	0.0150	-0.2602
messageFmessage_7	-0.3163	-0.0126					

```
## Adding the fixed effects estimates (biased, but more precise/statistically powerful)
```

```
rq2_fe_est <- lm_robust(vaccinated ~ messageF, fixed_effects = ~date_sentF, data = wrkdat3)
rq2_fe_est
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_1	-0.0022322	0.002007	-1.1125	0.26595	-0.006165	0.0017006	142407
messageFmessage_2	-0.0002051	0.001966	-0.1044	0.91688	-0.004058	0.0036474	142407

```

messageFmessage_3 0.0005308 0.002000 0.2654 0.79071 -0.003390 0.0044514 142407
messageFmessage_4 -0.0029133 0.002013 -1.4469 0.14792 -0.006860 0.0010330 142407
messageFmessage_5 -0.0006247 0.001874 -0.3334 0.73881 -0.004297 0.0030474 142407
messageFmessage_6 -0.0027722 0.001567 -1.7690 0.07689 -0.005844 0.0002992 142407
messageFmessage_7 -0.0034071 0.001902 -1.7915 0.07321 -0.007135 0.0003204 142407
messageFmessage_8 0.0004312 0.002011 0.2144 0.83020 -0.003510 0.0043723 142407

```

The unadjusted tests of independence of each message versus control using permutation approximations to the randomization inference and the Cochran-Mantel-Haenszel test for 2x2xK experiments show no differences between any message and control at $\alpha = .05$.

```

test_msgs <- function(msg1, msg2) {
  ## msg1 and msg2 are strings indicating message assignment in messageF
  effect_test <- cmh_test(vaccinatedF ~ messageF | date_sentF,
    data = wrkdat3,
    subset = wrkdat3$messageF %in% c(msg1, msg2),
    distribution = approximate(nresample = 10000, parallel = "multicore", ncpu = 6)
  )
  return(pvalue(effect_test)[1])
}

message_test_ps <- sapply(levels(wrkdat3$messageF)[-1], function(msg) {
  test_msgs("message_0", msg)
})

message_test_ps

```

```

message_1 message_2 message_3 message_4 message_5 message_6 message_7 message_8
0.2797 1.0000 0.6950 0.1812 0.9313 0.1039 0.0538 0.8993

```

We specified that we would report adjusted p-values, although it is hardly necessary since we are not reporting any discoveries.

The FDR adjustments (direct and q-values):

```

cbind(message_test_ps, fdr_adjusted = p.adjust(message_test_ps, method = "fdr"))

```

	message_test_ps	fdr_adjusted
message_1	0.2797	0.5594
message_2	1.0000	1.0000
message_3	0.6950	1.0000
message_4	0.1812	0.4832
message_5	0.9313	1.0000
message_6	0.1039	0.4156
message_7	0.0538	0.4156
message_8	0.8993	1.0000

Here are the q-values (same as the “adjusted p-values” above) (not clearly worth diving into since we have no effects but including a link to an explanation here <https://www.bioconductor.org/packages/devel/bioc/vignettes/qvalue/inst/doc/qvalue.pdf>)

```

library(qvalue)
rq2_qvals <- qvalue(message_test_ps, lambda = seq(0.05, 0.65, 0.05))
rq2_qvals$qvalues

```

```

message_1 message_2 message_3 message_4 message_5 message_6 message_7 message_8
0.5594 1.0000 1.0000 0.4832 1.0000 0.4156 0.4156 1.0000

```

A figure showing the results from rq2 estimation with 95% confidence intervals.

```

rq2plot_dat <- tidy(rq2_est)

rq2plot_dat$term <- c("Control", paste("M", 1:8, " v Ctrl", sep = ""))

rq2plot_dat$mtype <- c("Control", "Ownership", "Safety", "Preventing bad outcomes", "Epistemic humility+no bad outcomes", "Access")

total_n_by_arm <- table(wrkdat3$messageF)

```

```

rq2plot_dat$n_arm <- total_n_by_arm

rq2plot_dat <- rq2plot_dat %>%
  filter(term != "Control") %>%
  arrange(estimate)
rq2plot_dat$termF <- factor(rq2plot_dat$term, levels = rq2plot_dat$term)

rq2xlim <- range(c(rq2plot_dat$conf.low, rq2plot_dat$conf.high + .001))

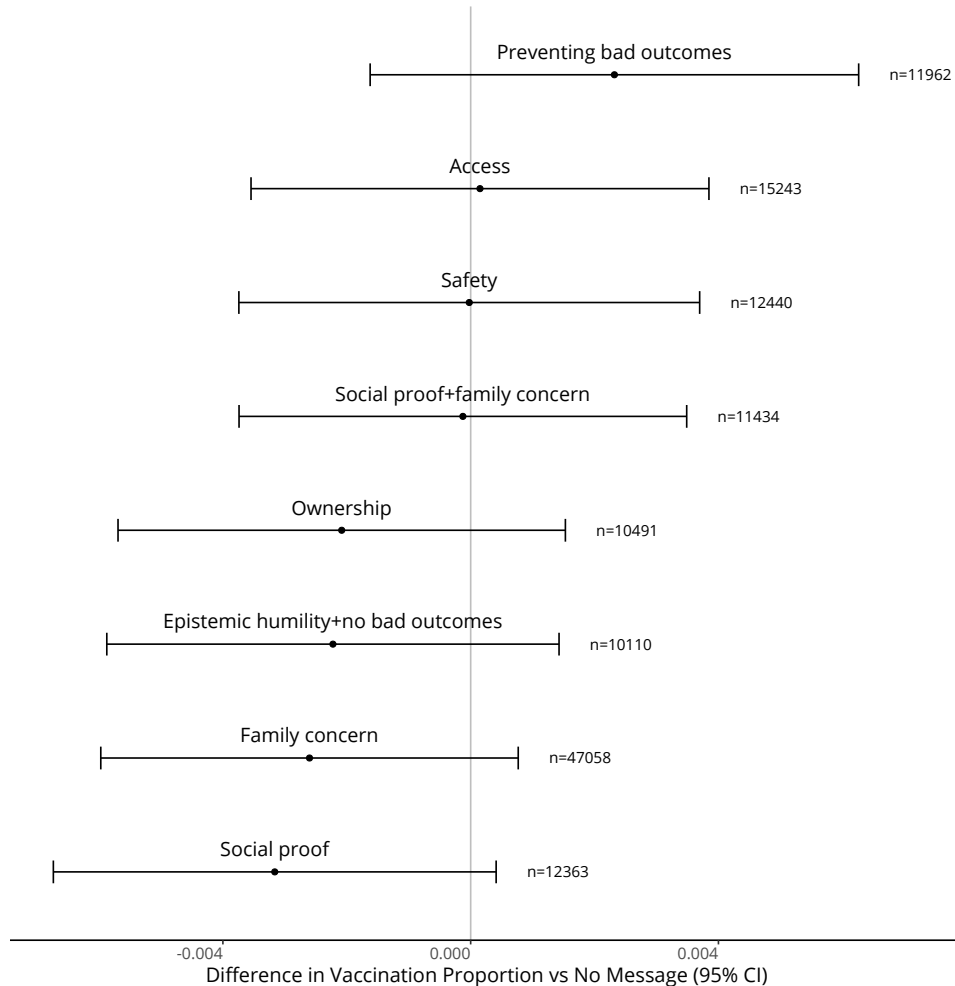
rq2plot <- ggplot(rq2plot_dat, aes(x = estimate, y = termF)) +
  geom_vline(aes(xintercept = 0), color = "grey") +
  geom_point() +
  geom_errorbarh(mapping = aes(xmin = conf.low, xmax = conf.high), height = .2) +
  xlab("Difference in Vaccination Proportion vs No Message (95% CI)") +
  geom_text(aes(label = mtype), check_overlap = TRUE, nudge_y = .2, family = "Open Sans", size = 4.5) +
  geom_text(aes(label = paste("n=", n_arm, sep = ""), x = conf.high), check_overlap = TRUE, nudge_x = .001, family = "Open Sans", size = 4.5) +
  ylab("") +
  theme_classic(base_family = "Open Sans") +
  xlim(rq2xlim) +
  theme(
    text = element_text(size = 13),
    axis.line.y = element_blank(),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.text.x = element_text(angle = 0, hjust = 1)
  )

# print(rq2plot)

ggsave(file = "rq2plot.pdf", path = OUTPUT_DIR, plot = rq2plot, device = cairo_pdf)
ggsave(file = "rq2plot.png", path = OUTPUT_DIR, plot = rq2plot, type = "cairo-png", dpi = 300)

include_graphics(here(OUTPUT_DIR, "rq2plot.pdf"))

```



RQ3: Does epistemic humility help?

Message 4 vs. 3 (CMH test, difference of proportions estimator). Only very small differences between those two arms.

```
rq3_est <- difference_in_means(vaccinated ~ messageF, blocks = date_sent, data = wrkdat3, subset = wrkdat3$messageF %in% c("message_3", "message_4"))
rq3_est
```

Design: Blocked

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_4	-0.003194	0.002074	-1.54	0.1235	-0.00726	0.0008707	22046

```
rq3_test <- test_msgs("message_3", "message_4")
rq3_test
```

[1] 0.1057

RQ5: How do social proof and appeals to the family interact?

We will test the overall hypothesis of no difference between 6 (family concern), 7 (social proof), and 8 (family concern + social proof). If we reject this, we test 6 versus 8 and 7 versus 8.

```
rq5_overall <- cmh_test(vaccinatedF ~ messageF | date_sentF, data = wrkdat3, subset = wrkdat3$messageF %in% c("message_6", "message_7", "message_8"))
rq5_overall_perm <- cmh_test(vaccinatedF ~ messageF | date_sentF, data = wrkdat3, subset = wrkdat3$messageF %in% c("message_6", "message_7", "message_8"))
rq5_overall
```


Asymptotic Generalized Cochran-Mantel-Haenszel Test

```
data: vaccinatedF by
      messageF (message_6, message_7, message_8)
      stratified by date_sentF
chi-squared = 5.1, df = 2, p-value = 0.08
rq5_overall_perm
```

Approximative Generalized Cochran-Mantel-Haenszel Test

```
data: vaccinatedF by
      messageF (message_6, message_7, message_8)
      stratified by date_sentF
chi-squared = 5.1, p-value = 0.08
```

So, since we have a marginal rejection, we do the other tests. The differences are still quite small.

```
test_msgs("message_6", "message_7")
```

```
[1] 0.5753
```

```
test_msgs("message_6", "message_8")
```

```
[1] 0.1674
```

```
test_msgs("message_7", "message_8")
```

```
[1] 0.0235
```

```
rq_5a_est <- difference_in_means(vaccinated ~ messageF, blocks = date_sent, data = wrkdat3, subset = wrkdat3$messageF %in% c("message_6", "message_7"))
rq_5b_est <- difference_in_means(vaccinated ~ messageF, blocks = date_sent, data = wrkdat3, subset = wrkdat3$messageF %in% c("message_6", "message_8"))
rq_5c_est <- difference_in_means(vaccinated ~ messageF, blocks = date_sent, data = wrkdat3, subset = wrkdat3$messageF %in% c("message_7", "message_8"))
```

```
rq_5a_est
```

Design: Blocked

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_7	-0.0002589	0.001726	-0.15	0.8807	-0.003642	0.003124	59395

```
rq_5b_est
```

Design: Blocked

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_8	0.0009202	0.001569	0.5863	0.5577	-0.002156	0.003996	58466

```
rq_5c_est
```

Design: Blocked

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_8	0.004312	0.001919	2.247	0.02463	0.0005511	0.008073	23771

RQ6: Did adaptive randomization increase vaccinations over fixed randomization?

We also will report the effect of using adaptive randomization versus fixed randomization on total vaccinations — since we withheld 25% of each of the three weeks experimental pools for fixed randomization and adapted the other 100 – 25%. Our aim in this study was to (1) learn about which messages worked best but also (2) increase vaccination. The fixed randomization maximized statistical power to detect effects whereas the adaptive randomization increased the numbers of people exposed to more effective messages.

It looks like slightly more people were vaccinated in the non-adaptive arm of the study.

```
with(wrkd3, table(date_sent, is_chosen_from_uniform, exclude = c()))
```

```

      is_chosen_from_uniform
date_sent FALSE TRUE
2021-05-25    0 10003
2021-05-26    0  9999
2021-05-27    0  9999
2021-05-28    0  9999
2021-06-02  5947  1994
2021-06-03  5969  1972
2021-06-04  5919  2029
2021-06-07  5933  2009
2021-06-08  5988  1949
2021-06-09 10437  5245
2021-06-10 10503  5176
2021-06-11 10501  5178
2021-06-14 10499  5180

```

```
rq6_est <- difference_in_means(vaccinated ~ is_chosen_from_uniform, blocks = date_sentF, data = wrkd3, subset = wrkd3$date_sent > 2021-06-07)
rq6_est
```

Design: Blocked

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
is_chosen_from_uniform	0.0009958	0.0008211	1.213	0.2252	-0.0006135	0.002605	102410

```
rq6_cmh_perm <- cmh_test(vaccinatedF ~ factor(is_chosen_from_uniform) | date_sentF, data = wrkd3, subset = wrkd3$date_sent > 2021-06-07)
rq6_cmh_perm
```

Approximative Generalized Cochran-Mantel-Haenszel Test

```

data: vaccinatedF by
      factor(is_chosen_from_uniform) (FALSE, TRUE)
      stratified by date_sentF
chi-squared = 1.5, p-value = 0.2

```

Exploratory Analyses Not Pre-registered

Effects on vaccination within a week

The experiment ran during a time of national campaigns in favor of vaccination. The control group in our experiment would have been exposed to this, and thus, might have gotten vaccinated for reasons other than a nudge from a text message.

No strong evidence that people were likely to be vaccinated within a week in “any message” versus control or versus any given message.

```
rq7_test <- cmh_test(vac_in_weekF ~ not_controlF | date_sentF, data = wrkd3)
rq7_test
```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```

data: vac_in_weekF by not_controlF (0, 1)
      stratified by date_sentF
chi-squared = 0.00041, df = 1, p-value = 1

```

```
rq7a_test <- cmh_test(vac_in_weekF ~ messageF | date_sentF, data = wrkd3)
rq7a_test
```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```

data: vac_in_weekF by
      messageF (message_0, message_1, message_2, message_3, message_4, message_5, message_6, message_7, message_8)
      stratified by date_sentF
chi-squared = 8.7, df = 8, p-value = 0.4

```

```
rq7a_est <- lm_robust(vac_in_week ~ messageF, weights = IPW_weight_multarm, data = wrkdat3)
rq7a_est
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.009127748	0.0009266	9.851260	6.884e-23	0.0073117	0.010944	142419
messageFmessage_1	0.000124368	0.0013438	0.092548	9.263e-01	-0.0025095	0.002758	142419
messageFmessage_2	0.000262444	0.0013718	0.191314	8.483e-01	-0.0024263	0.002951	142419
messageFmessage_3	0.002280338	0.0014842	1.536381	1.244e-01	-0.0006287	0.005189	142419
messageFmessage_4	-0.001289688	0.0012946	-0.996239	3.191e-01	-0.0038270	0.001248	142419
messageFmessage_5	-0.000005737	0.0013428	-0.004273	9.966e-01	-0.0026376	0.002626	142419
messageFmessage_6	-0.000254321	0.0011945	-0.212902	8.314e-01	-0.0025956	0.002087	142419
messageFmessage_7	-0.000963766	0.0013199	-0.730162	4.653e-01	-0.0035508	0.001623	142419
messageFmessage_8	0.000933308	0.0013295	0.702014	4.827e-01	-0.0016724	0.003539	142419

```
rq7plot_dat <- tidy(rq7a_est)
```

```
rq7plot_dat$term <- c("Control", paste("M", 1:8, " v Ctrl", sep = ""))
```

```
rq7plot_dat$mtype <- c("Control", "Ownership", "Safety", "Preventing bad outcomes", "Epistemic humility+no bad outcomes", "Access to information")
```

```
total_n_by_arm <- table(wrkdat3$messageF)
```

```
rq7plot_dat$n_arm <- total_n_by_arm
```

```
rq7plot_dat <- rq7plot_dat %>% filter(term != "Control")
```

```
rq7plot_dat$termF <- factor(rq7plot_dat$term, levels = rq2plot_dat$term)
```

```
with(rq7plot_dat, table(term, termF, exclude = c()))
```

term	termF									
	M7 v Ctrl	M6 v Ctrl	M4 v Ctrl	M1 v Ctrl	M8 v Ctrl	M2 v Ctrl	M5 v Ctrl	M3 v Ctrl		
M1 v Ctrl	0	0	0	1	0	0	0	0		
M2 v Ctrl	0	0	0	0	0	1	0	0		
M3 v Ctrl	0	0	0	0	0	0	0	1		
M4 v Ctrl	0	0	1	0	0	0	0	0		
M5 v Ctrl	0	0	0	0	0	0	1	0		
M6 v Ctrl	0	1	0	0	0	0	0	0		
M7 v Ctrl	1	0	0	0	0	0	0	0		
M8 v Ctrl	0	0	0	0	1	0	0	0		

```
rq7plot <- ggplot(rq7plot_dat, aes(x = estimate, y = termF)) +
  geom_vline(aes(xintercept = 0), color = "grey") +
  geom_point() +
  geom_errorbarh(mapping = aes(xmin = conf.low, xmax = conf.high), height = .2) +
  xlab("Difference in Vaccination Proportion vs No Message within a Week (95% CI)") +
  geom_text(aes(label = mtype), check_overlap = TRUE, nudge_y = .2, family = "Open Sans", size = 4.5) +
  ylab("") +
  theme_classic(base_family = "Open Sans") +
  xlim(rq2xlim) +
  theme(
    text = element_text(size = 13),
    axis.line.y = element_blank(),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.text.x = element_text(angle = 0, hjust = 1)
  )
```

```
# print(rq7plot)
```

```
ggsave(file = "rq7plot.pdf", path = OUTPUT_DIR, plot = rq7plot, device = cairo_pdf)
```

```
ggsave(file = "rq7plot.png", path = OUTPUT_DIR, plot = rq7plot, type = "cairo-png", dpi = 300)
```

```
rq2_rq7_plot <- ggarrange(rq2plot, rq7plot, nrow = 1)
```

Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x\$label)): font family 'Open Sans' not found in PostScript font database

Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x\$label)): font family 'Open Sans' not found in PostScript font database

Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x\$label)): font family 'Open Sans' not found in PostScript font database

Warning in grid.Call(C_stringMetric, as.graphicsAnnot(x\$label)): font family 'Open Sans' not found in PostScript font database

```
# print(rq2_rq7_plot)

ggsave(file = "rq2_rq7_plot.pdf", path = OUTPUT_DIR, plot = rq2_rq7_plot, device = cairo_pdf, width = 12, height = 6)
ggsave(file = "rq2_rq7_plot.png", path = OUTPUT_DIR, plot = rq2_rq7_plot, type = "cairo-png", dpi = 300, width = 12, height = 6)
```

We looked at both overall vaccination and vaccination within a week (only available for those assigned in the first week). The following table shows that we have no strong arguments against the claim that our messages were the same as control in regards either outcome. (Not adjusting p-values here because this is exploratory work and because we have so few small p-values).

```

    data = wrkdat3,
    subset = wrkdat3$messageF %in% c(msg1, msg2) & wrkdat3$iteration == the_iteration,
    distribution = asymptotic() # approximate(nresample = 10000, parallel = "multicore", ncpu = 6)
  )
  return(pvalue(effect_test)[1])
}

msg_by_iteration <- as_tibble(expand.grid(iteration = 1:3, messageF = levels(wrkdat3$messageF)[-1], stringsAsFactors = FALSE))

test_msgs2(msg1 = "message_0", msg2 = msg_by_iteration$messageF[1], the_iteration = 3)

```

```
[1] 0.5406
```

```

set.seed(12345)
msg_by_iteration <- msg_by_iteration %>%
  rowwise() %>%
  mutate(p_vs_ctrl = test_msgs2("message_0", messageF, iteration)) %>%
  arrange(iteration, messageF)

msg_by_iteration <- msg_by_iteration %>%
  rowwise() %>%
  mutate(p_vac_week_vs_ctrl = test_msgs2("message_0", messageF, iteration, thefmla = vac_in_weekF ~ messageF | date_sentF))

msg_by_iteration <- msg_by_iteration %>% mutate(p_vac_week_vs_ctrl = ifelse(p_vac_week_vs_ctrl == p_vs_ctrl, NA, p_vac_week_vs_ctrl))
print(msg_by_iteration, n = 100)

```

```
# A tibble: 24 x 4
```

```
# Rowwise:
```

	iteration	messageF	p_vs_ctrl	p_vac_week_vs_ctrl
	<int>	<chr>	<dbl>	<dbl>
1	1	message_1	0.414	0.624
2	1	message_2	0.777	0.776
3	1	message_3	0.485	0.554
4	1	message_4	0.0950	0.151
5	1	message_5	0.432	0.999
6	1	message_6	0.174	0.368
7	1	message_7	0.00849	0.185
8	1	message_8	0.911	0.458
9	2	message_1	0.601	0.188
10	2	message_2	0.685	0.655
11	2	message_3	0.870	0.127
12	2	message_4	0.510	0.215
13	2	message_5	0.369	0.357
14	2	message_6	0.741	0.250
15	2	message_7	0.732	0.298
16	2	message_8	0.289	0.0548
17	3	message_1	0.541	0.731
18	3	message_2	0.988	0.893
19	3	message_3	0.0394	0.127
20	3	message_4	0.321	0.203
21	3	message_5	0.944	0.680
22	3	message_6	0.318	0.517
23	3	message_7	0.567	0.523
24	3	message_8	0.405	0.469

Nor is there strong evidence that “any message” was better than control, even when we assess the relationships for each iteration separately:

```

rq8_iteration1_test <- cmh_test(vaccinatedF ~ not_controlF | date_sentF, data = wrkdat3, subset = wrkdat3$iteration == 1)
rq8_iteration1_test

```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```

data: vaccinatedF by not_controlF (0, 1)
      stratified by date_sentF
chi-squared = 1, df = 1, p-value = 0.3

```

```
rq8_iteration2_test <- cmh_test(vaccinatedF ~ not_controlF | date_sentF, data = wrkdat3, subset = wrkdat3$iteration == 2)
rq8_iteration2_test
```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```
data: vaccinatedF by not_controlF (0, 1)
      stratified by date_sentF
chi-squared = 0.0055, df = 1, p-value = 0.9
```

```
rq8_iteration3_test <- cmh_test(vaccinatedF ~ not_controlF | date_sentF, data = wrkdat3, subset = wrkdat3$iteration == 3)
rq8_iteration3_test
```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```
data: vaccinatedF by not_controlF (0, 1)
      stratified by date_sentF
chi-squared = 0.57, df = 1, p-value = 0.5
```

Also looking at vaccinations within a week for the first iteration

```
rq9_iteration1_test <- cmh_test(vac_in_weekF ~ not_controlF | date_sentF, data = wrkdat3, subset = wrkdat3$iteration == 1)
rq9_iteration1_test
```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```
data: vac_in_weekF by not_controlF (0, 1)
      stratified by date_sentF
chi-squared = 0.36, df = 1, p-value = 0.6
```

```
rq9_iteration2_test <- cmh_test(vac_in_weekF ~ not_controlF | date_sentF, data = wrkdat3, subset = wrkdat3$iteration == 2)
rq9_iteration2_test
```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```
data: vac_in_weekF by not_controlF (0, 1)
      stratified by date_sentF
chi-squared = 1.9, df = 1, p-value = 0.2
```

```
rq9_iteration3_test <- cmh_test(vac_in_weekF ~ not_controlF | date_sentF, data = wrkdat3, subset = wrkdat3$iteration == 3)
rq9_iteration3_test
```

Asymptotic Generalized Cochran-Mantel-Haenszel Test

```
data: vac_in_weekF by not_controlF (0, 1)
      stratified by date_sentF
chi-squared = 0.34, df = 1, p-value = 0.6
```

Exploratory Analyses Pre-registered

These analysis all compare effects of messages as they might vary for people who live in different kinds of places (using ZCTA as the place).

EQ1: Do explicit appeals to the safety of vaccines increase responses in areas with higher proportions of Black or Latinx people? Message 2 vs. control

We cannot detect any simple linear differential effect of pct black or latinx on the message 2 versus control comparison.

```
wrkdat3_eq1 <- wrkdat3 %>%
  filter(messageF %in% c("message_0", "message_2") & zcta != "00000") %>%
  droplevels()
dim(wrkdat3_eq1)
```

```
[1] 23249    44
```

```
table(wrkdat3_eq1$date_sent, wrkdat3_eq1$messageF, exclude = c())
```

	message_0	message_2
2021-05-25	1089	1087
2021-05-26	1095	1099
2021-05-27	1087	1087
2021-05-28	1092	1088
2021-06-02	688	1108
2021-06-03	677	1107
2021-06-04	675	1109
2021-06-07	679	1095
2021-06-08	686	1105
2021-06-09	830	572
2021-06-10	823	576
2021-06-11	831	570
2021-06-14	819	575

```
make_weights <- function(dat) {
  block_m_each <- with(dat, table(date_sentF, messageF, exclude = c()))
  block_prob_each <- block_m_each / rowSums(block_m_each)
  declared_randomization <- declare_ra(blocks = dat$date_sentF, block_m_each = block_m_each, conditions = sort(unique(dat$messageF)))
  IPW_weight <- 1 / obtain_condition_probabilities(declaration = declared_randomization, assignment = dat$messageF)
  stopifnot(all.equal(sort(unique(1 / IPW_weight)), sort(unique(block_prob_each))))
  return(IPW_weight)
}
```

```
wrkdat3_eq1$IPW_eq1 <- make_weights(wrkdat3_eq1)
## So, good that I didn't use the multi-arm weights.
with(wrkdat3_eq1, cor(IPW_eq1, IPW_weight_multarm))
```

```
[1] 0.352
```

```
eq1_blk_estA <- lm_robust(vaccinated ~ messageF * pct_any_blk, data = wrkdat3_eq1, weights = IPW_eq1)
## Just including Fixed Effects for curiosity. We will report estA
eq1_blk_estB <- lm_robust(vaccinated ~ messageF * pct_any_blk, data = wrkdat3_eq1, fixed_effects = ~date_sentF)
```

```
eq1_blk_estA
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.0188928	0.002016	9.370867	7.829e-21	0.014941	0.022845	23245
messageFmessage_2	-0.0002165	0.002783	-0.077775	9.380e-01	-0.005672	0.005239	23245
pct_any_blk	0.0538385	0.018998	2.833943	4.602e-03	0.016602	0.091075	23245
messageFmessage_2:pct_any_blk	-0.0002156	0.026120	-0.008253	9.934e-01	-0.051413	0.050982	23245

```
eq1_blk_estB
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_2	-0.0002700	0.002751	-0.09816	0.921810	-0.005662	0.005122	23233
pct_any_blk	0.0553735	0.018627	2.97276	0.002954	0.018863	0.091884	23233
messageFmessage_2:pct_any_blk	0.0004877	0.025627	0.01903	0.984818	-0.049744	0.050719	23233

```
eq1_lat_estA <- lm_robust(vaccinated ~ messageF * pct_hisp, data = wrkdat3_eq1, weights = IPW_eq1)
## Just including Fixed Effects for curiosity. We will report estA
eq1_lat_estB <- lm_robust(vaccinated ~ messageF * pct_hisp, data = wrkdat3_eq1, fixed_effects = ~date_sentF)
```

```
eq1_lat_estA
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.018012	0.001849	9.7392	2.264e-22	0.014387	0.021637	23245
messageFmessage_2	0.001080	0.002534	0.4262	6.700e-01	-0.003887	0.006048	23245
pct_hisp	0.034689	0.009438	3.6756	2.378e-04	0.016191	0.053187	23245
messageFmessage_2:pct_hisp	-0.007924	0.012534	-0.6322	5.272e-01	-0.032492	0.016643	23245

```
eq1_lat_estB
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_2	0.000861	0.002514	0.3424	0.7320161	-0.004067	0.005789	23233
pct_hisp	0.033746	0.009169	3.6804	0.0002334	0.015774	0.051717	23233
messageFmessage_2:pct_hisp	-0.006495	0.012236	-0.5309	0.5955128	-0.030478	0.017487	23233

EQ2: Does the implication of choice through emphasis on a conspicuous advantage increase responses in areas with higher proportions of Republican people? Message 3 vs. control

No detectable difference in effects.

```
wrkdat3_eq2 <- wrkdat3 %>%
  filter(messageF %in% c("message_0", "message_3") & zcta != "00000") %>%
  droplevels()
dim(wrkdat3_eq2)
```

```
[1] 22772    44
table(wrkdat3_eq2$date_sent, wrkdat3_eq2$messageF, exclude = c())
```

	message_0	message_3
2021-05-25	1089	1080
2021-05-26	1095	1096
2021-05-27	1087	1091
2021-05-28	1092	1084
2021-06-02	688	986
2021-06-03	677	980
2021-06-04	675	989
2021-06-07	679	996
2021-06-08	686	995
2021-06-09	830	603
2021-06-10	823	599
2021-06-11	831	602
2021-06-14	819	600

```
wrkdat3_eq2$IPW_eq2 <- make_weights(wrkdat3_eq2)

eq2_gop_estA <- lm_robust(vaccinated ~ messageF * pct_gop, data = wrkdat3_eq2, weights = IPW_eq2)
## Just including Fixed Effects for curiosity. We will report estA
eq2_gop_estB <- lm_robust(vaccinated ~ messageF * pct_gop, data = wrkdat3_eq2, fixed_effects = ~date_sentF)

eq2_gop_estA
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.033399	0.00480	6.9582	3.541e-12	0.02399	0.042808	22768
messageFmessage_3	-0.002249	0.00661	-0.3402	7.337e-01	-0.01520	0.010707	22768
pct_gop	-0.027623	0.01238	-2.2321	2.562e-02	-0.05188	-0.003366	22768
messageFmessage_3:pct_gop	0.009509	0.01705	0.5576	5.771e-01	-0.02392	0.042935	22768

```
eq2_gop_estB
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_3	-0.001908	0.006526	-0.2924	0.76996	-0.01470	0.0108826	22756
pct_gop	-0.024921	0.012234	-2.0370	0.04166	-0.04890	-0.0009412	22756
messageFmessage_3:pct_gop	0.008442	0.016846	0.5012	0.61627	-0.02458	0.0414618	22756

EQ3: Do explicit appeals to ease of access increase responses in areas with higher proportions of Black or Latinx people? Message 5 vs. control

No detectable differences. Magnitude of moderation is large-ish given this phenomenon (on order of 1 or 2 pts, but negative).

```
wrkdat3_eq3 <- wrkdat3 %>%
  filter(messageF %in% c("message_0", "message_5") & zcta != "00000") %>%
  droplevels()
dim(wrkdat3_eq3)
```

```
[1] 25934    44
table(wrkdat3_eq3$date_sent, wrkdat3_eq3$messageF, exclude = c())
```


	message_0	message_5
2021-05-25	1089	1084
2021-05-26	1095	1086
2021-05-27	1087	1088
2021-05-28	1092	1091
2021-06-02	688	1655
2021-06-03	677	1645
2021-06-04	675	1653
2021-06-07	679	1665
2021-06-08	686	1656
2021-06-09	830	559
2021-06-10	823	566
2021-06-11	831	558
2021-06-14	819	557

```

wrkdat3_eq3$IPW_eq3 <- make_weights(wrkdat3_eq3)

eq3_blk_estA <- lm_robust(vaccinated ~ messageF * pct_any_blk, data = wrkdat3_eq3, weights = IPW_eq3)
## Just including Fixed Effects for curiosity. We will report estA
eq3_blk_estB <- lm_robust(vaccinated ~ messageF * pct_any_blk, data = wrkdat3_eq3, fixed_effects = ~date_sentF)
eq3_blk_estA

```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.018567	0.002043	9.0861	1.098e-19	0.01456	0.022572	25930
messageFmessage_5	0.001946	0.002743	0.7095	4.780e-01	-0.00343	0.007322	25930
pct_any_blk	0.051307	0.019179	2.6752	7.473e-03	0.01372	0.088898	25930
messageFmessage_5:pct_any_blk	-0.026276	0.025041	-1.0493	2.940e-01	-0.07536	0.022805	25930

```
eq3_blk_estB
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_5	0.002159	0.002699	0.8001	0.423633	-0.00313	0.007449	25918
pct_any_blk	0.055429	0.018619	2.9771	0.002913	0.01894	0.091922	25918
messageFmessage_5:pct_any_blk	-0.026560	0.024224	-1.0964	0.272898	-0.07404	0.020921	25918

```

eq3_lat_estA <- lm_robust(vaccinated ~ messageF * pct_hisp, data = wrkdat3_eq3, weights = IPW_eq3)
## Just including Fixed Effects for curiosity. We will report estA
eq3_lat_estB <- lm_robust(vaccinated ~ messageF * pct_hisp, data = wrkdat3_eq3, fixed_effects = ~date_sentF)
eq3_lat_estA

```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.017433	0.001874	9.3041	1.455e-20	0.013760	0.021111	25930
messageFmessage_5	0.001536	0.002512	0.6114	5.410e-01	-0.003388	0.00646	25930
pct_hisp	0.034860	0.009734	3.5814	3.424e-04	0.015782	0.05394	25930
messageFmessage_5:pct_hisp	-0.011863	0.012585	-0.9426	3.459e-01	-0.036531	0.01280	25930

```
eq3_lat_estB
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_5	0.001597	0.002468	0.6474	0.5173976	-0.003239	0.006434	25918
pct_hisp	0.033723	0.009157	3.6828	0.0002311	0.015775	0.051671	25918
messageFmessage_5:pct_hisp	-0.011048	0.011843	-0.9328	0.3509152	-0.034261	0.012166	25918

EQ4: Does epistemic humility increase responses in areas with higher proportions of either Black or Latinx people or Republican people? Message 4 versus 3

No detectable differences in effect.

```

wrkdat3_eq4 <- wrkdat3 %>%
  filter(messageF %in% c("message_3", "message_4") & zcta != "00000") %>%
  droplevels()
dim(wrkdat3_eq4)

```

[1] 21577 44

```
table(wrkd3_eq4$date_sent, wrkd3_eq4$messageF, exclude = c())
```

	message_3	message_4
2021-05-25	1080	1081
2021-05-26	1096	1090
2021-05-27	1091	1084
2021-05-28	1084	1089
2021-06-02	986	475
2021-06-03	980	471
2021-06-04	989	475
2021-06-07	996	471
2021-06-08	995	464
2021-06-09	603	786
2021-06-10	599	799
2021-06-11	602	794
2021-06-14	600	797

```
wrkd3_eq4$IPW_eq4 <- make_weights(wrkd3_eq4)
```

```
eq4_gop_estA <- lm_robust(vaccinated ~ messageF * pct_gop, data = wrkd3_eq4, weights = IPW_eq4)
## Just including Fixed Effects for curiosity. We will report estA
eq4_gop_estB <- lm_robust(vaccinated ~ messageF * pct_gop, data = wrkd3_eq4, fixed_effects = ~date_sentF)
eq4_gop_estA
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.031424	0.004622	6.7992	1.079e-11	0.02237	0.040483	21573
messageFmessage_4	-0.002244	0.006677	-0.3360	7.369e-01	-0.01533	0.010844	21573
pct_gop	-0.018053	0.011944	-1.5115	1.307e-01	-0.04146	0.005358	21573
messageFmessage_4:pct_gop	-0.003277	0.017226	-0.1902	8.491e-01	-0.03704	0.030487	21573

eq4_gop_estB

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_4	-0.002618	0.006497	-0.4030	0.6870	-0.01535	0.010116	21561
pct_gop	-0.016735	0.011589	-1.4440	0.1488	-0.03945	0.005981	21561
messageFmessage_4:pct_gop	-0.002784	0.016703	-0.1667	0.8676	-0.03552	0.029954	21561

```
eq4_blk_estA <- lm_robust(vaccinated ~ messageF * pct_any_blk, data = wrkd3_eq4, weights = IPW_eq4)
## Just including Fixed Effects for curiosity. We will report estA
eq4_blk_estB <- lm_robust(vaccinated ~ messageF * pct_any_blk, data = wrkd3_eq4, fixed_effects = ~date_sentF)
eq4_blk_estA
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.019125	0.002084	9.17906	4.730e-20	0.015041	0.02321	21573
messageFmessage_4	-0.003302	0.002909	-1.13507	2.564e-01	-0.009003	0.00240	21573
pct_any_blk	0.067888	0.020681	3.28267	1.030e-03	0.027352	0.10842	21573
messageFmessage_4:pct_any_blk	-0.001599	0.028836	-0.05547	9.558e-01	-0.058120	0.05492	21573

eq4_blk_estB

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_4	-0.0036235	0.002836	-1.277497	0.201440	-0.009183	0.001936	21561
pct_any_blk	0.0613615	0.019686	3.117026	0.001829	0.022776	0.099947	21561
messageFmessage_4:pct_any_blk	-0.0001924	0.027278	-0.007054	0.994371	-0.053659	0.053274	21561

```
eq4_lat_estA <- lm_robust(vaccinated ~ messageF * pct_hisp, data = wrkd3_eq4, weights = IPW_eq4)
## Just including Fixed Effects for curiosity. We will report estA
eq4_lat_estB <- lm_robust(vaccinated ~ messageF * pct_hisp, data = wrkd3_eq4, fixed_effects = ~date_sentF)
eq4_lat_estA
```

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
(Intercept)	0.018922	0.001878	10.0774	7.851e-24	0.015242	0.022603	21573
messageFmessage_4	-0.002755	0.002664	-1.0345	3.009e-01	-0.007976	0.002465	21573
pct_hisp	0.038470	0.009790	3.9294	8.542e-05	0.019280	0.057660	21573
messageFmessage_4:pct_hisp	-0.004235	0.013966	-0.3032	7.617e-01	-0.031610	0.023140	21573

eq4_lat_estB

	Estimate	Std. Error	t value	Pr(> t)	CI Lower	CI Upper	DF
messageFmessage_4	-0.002767	0.002619	-1.0565	0.2907542	-0.007901	0.002367	21561

```
pct_hisp          0.036447  0.009411  3.8729 0.0001079  0.018001 0.054893 21561
messageFmessage_4:pct_hisp -0.005410  0.013201 -0.4098 0.6819606 -0.031284 0.020465 21561
```

EQ5: Is there a day-of-week effect? Proportions of vaccinations collapsed across all messages by day.

Since the randomization to message occurred **within day** and we have relatively few weeks, it is difficult to disentangle day of week effects from date effects. So, we only present descriptive information here.

```
summary(wrkdat3$date_sent)
```

```
      Min.      1st Qu.      Median      Mean      3rd Qu.      Max.
"2021-05-25" "2021-05-28" "2021-06-07" "2021-06-04" "2021-06-10" "2021-06-14"
```

```
table(weekdays(wrkdat3$date_sent))
```

```
Friday  Monday  Thursday  Tuesday Wednesday
33626   23621   33619    17940    33622
wrkdat3$weekday_sent <- weekdays(wrkdat3$date_sent)
```

```
wrkdat3_weekday <- wrkdat3 %>%
  group_by(weekday_sent) %>%
  summarize(
    prop_vac = mean(vaccinated),
    prop_vac_in_week = mean(vac_in_week), nweek = n()
  )
```

```
wrkdat3_weekday
```

```
# A tibble: 5 x 4
  weekday_sent prop_vac prop_vac_in_week nweek
  <chr>         <dbl>         <dbl> <int>
1 Friday       0.0181         0.00803 33626
2 Monday       0.00923       0.00703 23621
3 Thursday     0.0195         0.00943 33619
4 Tuesday      0.0294         0.0109 17940
5 Wednesday    0.0218         0.0102 33622
```

EQ6: Is there an iteration effect? Some people were randomly assigned to have 3 weeks to schedule a vaccination and others only 1 week before the study ended. We explore whether there is a difference here.

We addressed this analysis above in our analysis by day of week and iteration.

Housekeeping

Combining three plots into one:

```
combined_plot <- ggarrange(rq2_rq7_plot, NULL, plot1, nrow = 3, ncol = 1, heights = c(.8, .1, 1), widths = c(.5, 1, 1), align =
# print(combined_plot)

ggsave(file = "combined_plot.pdf", path = OUTPUT_DIR, plot = combined_plot, device = cairo_pdf, width = 18, height = 14)
ggsave(file = "combined_plot.png", path = OUTPUT_DIR, plot = combined_plot, type = "cairo-png", dpi = 300, width = 12, height =
```