Supplementary Information for A Retrospective Bayesian Model for Measuring Covariate Effects on Observed COVID-19 Test and Case Counts

Robert Kubinec

April 1st, 2020

1 Materials and Methods

In this document I put forward a model for the rate of infected people $I_{ct} \in (0, 1)$ in a given country/region c given the the number of cases a_{ct} and the number of tests q_{ct} for all outbreak time periods $t \in T$. The outcome is here defined as the tests and cases reported on a given day rather than the cumulative total for ease of modeling purposes (i.e., the lagged difference of the cumulative total). I assume that I_{ct} is an unobserved Beta-distributed variable with a 3-order polynomial time trend of the number of post-outbreak time periods $T_O < T_A$, where an outbreak begins at the first reported case in a given area. By using a single time function, the model assumes that the coronavirus follows a similar pattern of infection across countries, as appears to be the case (i.e., all countries are infected with the same virus).

The unobserved infection rate I_{ct} is a function of the following parameters:

$$Pr(I_{ct}|T = t) \sim Beta(\alpha_1 + \alpha_c + \beta_{O1} \sum_{c=1}^{C} \mathbf{1}(a_{ct'} > 0) \forall t' \in t' < t + \mathbf{1}\beta_{S1} + \tag{1}$$

$$\beta_{I1} t_o + \beta_{I2} t_o^2 + \beta_{I3} t_o^3 + \mathbf{1}\beta_{S2}, \phi) \tag{2}$$

Where $g(\cdot)$ is the inverse logit function, α_c are country-level intercepts (less one for identification), $\beta_{O1} \sum_{c=1}^{C} \mathbf{1}(a_{ct_a - 1} > 0) \forall t' \in t' < t$ is the sum of countries with at least one case of infection in the world at any previous time point, and the three β_{Ii} are polynomial coefficients of post-outbreak time points t_o. The sum of countries parameter measures the spread of the virus due to travel across state borders, and as such

will increase rapidly as more states are infected. By contrast, the polynomial time trends represent possible within-state transmission of the virus, which can cause exponentially growing case counts compared to the linear sum of infected countries parameter. Finally, the parameter ϕ is a dispersion parameter governing how precise the infection rate estimate is.

Given these two primary ways that the infection rate can increase, there are two ways that possible suppression measures targeted at the virus can enter the model. The first is a constant factor, $\mathbf{1}\beta_{S1}$, which is an indicator function that is equal to **1** if a country has taken suppression measures and **0** otherwise. This constant parameter is meant to capture the ability of countries or regions to stop the spread of transmission due to foreign travelers arriving in the country/region, which occurs at a roughly constant level over time.

The second way suppression measures enter the model is through $\mathbf{1}\beta_{S2}$, which can increase over time as the virus increases. This parameter reflects possible social distancing measures which will grow more effective as domestic transmission of the virus increases (i.e. as the polynomial time trend takes off). As such, it is assumed that any deviation from the common domestic virus transmission pattern is due to these time-varying suppression measures.

Once the outbreak has started in country c, which is indicated by a positive case count in time $t-1$, a time counter $t_O = 1$ starts for that country and increases by 1 for each following time point until T.

Given this specification of the infection rate process, we can then move to the generation of the observed data, tests q_{ct_a} and cases a_{ct_a}. The infection rate is assumed to influence both of these numbers. First, an increasing number of infections is associated with more tests as countries try to identify who may have the virus. Furthermore, a rising infection rate is associated with a higher ratio of positive results (reported cases) conditional on the number of tests. I model both of these observed indicators, tests and cases, jointly to simultaneously adjust for the infection rate's influence on both factors.

To model the number of tests, I assume that each country has an unobserved level of testing parameter, $\beta_{cq} > 0$, indicating how strongly each country is willing and able to perform tests as a factor of the unobserved infection rate. The number of observed tests q_{ct} for a given time point t and country c is distributed as a Binomial proportion of the countries' population c_p:

$$q_{ct} \sim B(c_p, g(\alpha_2 + \beta_{cq} I_{ct})) \tag{3}$$

The parameter β_{cq} serves to scale the infection rate I_{ct} so that an increasing infection rate has heterogeneous effects on the number of tests by country. The intercept α_2 indicates how many tests would be performed in a country with an infection rate of zero. It is assumed that this number is quite low, though not necessarily zero.

Given the parameter β_{cq}, a country could test almost no one or test far more than are actually infected depending on their willingness to impose tests. However, the number of tests is increasing in I_{ct} conditional on a country's willingness to test people. That is, regardless of how much a country wants to test people, as the outbreak grows the number of tests will increase though at very different rates.

Given the number of observed tests q_{ct}, I can then generate the number of observed cases a_{ct} as a binomial proportion of the number of tests q_{ct}:

$$a_{ct} \sim B(q_{ct}, g(\alpha_3 + \beta_a I_{ct_a})) \tag{4}$$

where $g(\cdot)$ is again the inverse logit function, α_3 is an intercept that indicates how many cases would test positive with an infection rate of zero (equal to the false positive rate of the test), and $\beta_a > 0$ is a parameter that determines how hard it is to find the infected people and test them as opposed to people who are not actually infected. The multiplication of this parameter and the infection rate determines the observed number of cases a_{ct} as a proportion of the number of observed tests q_{ct}.

To summarize the model, infection rates determine how many tests a country is likely to undertake and also the number of positive tests they receive conditional on a certain number of tests. This simultaneous adjustment helps takes care of mis-interpreting the observed data by not taking into account varying testing rates, which is likely why some policy makers argue that the epidemiological models are wrong.

Because sampling from a model with a hierarchical Beta parameter can be difficult, we can simplify the final likelihood by combining the beta distribution and the binomial counts into a beta-binomial model for tests:

$$q_{ct_a} \sim BB(c_p, g(\alpha_2 + \beta_q I_{ct_a}), \phi_q) \tag{5}$$

and cases:

$$a_{ct_a} \sim BB(q_{ct}, g(\alpha_3 + \beta_a I_{ct_a}), \phi_a) \tag{6}$$

For estimation purposes, the infection rates I_{ct} are put in to the beta-binomial model on the logit scale instead of transforming them to (0,1), but they can be transformed to proportions post-estimation with the inverse logit function.

1.1 Identification

This model contains an unobserved latent process I_{ct_a}, and as such there are further constraints necessary in order to have a unique scale and rotation of the latent variable. First, sign restrictions are put on the suppression measure parameters β_{S1} and β_{S2} so that they are strictly negative. Second, positivity constraints are put on the parameters β_a and β_q that govern the relationship between the infection rate and test and case counts. It is assumed that infection rates will increase both the number of tests and the number of cases relative to the number of tests, though at necessarily different rates.

The other important restriction is to fix the intercept for the cases model α_3 to a fixed value of 0.1, or -2.2 on the logit scale. The reason for this restriction is because the intercept necessarily equals the false positive rate of a COVID-19 test. While there is still ongoing research as to the false positive rate, it is clear that it is likely in the area of 1 in 10 false positives for rapid tests.[1] As such, by pinning this value, we can lower-bound the number of cases we are likely to see given an infection rate of zero, providing a helpful lower-bound for the estimate.

As I will show, no other identification restrictions are necessary to estimate the model beyond weakly informative priors assigned to parameters. These are:

$$\beta_a \sim E(.1) \tag{7}$$

$$\beta_{qc} \sim E(\sigma_q) \tag{8}$$

$$\sigma_q \sim E(.1) \tag{9}$$

$$\beta_{Si} \sim N(0, 2) \tag{10}$$

$$\alpha_c \sim N(0, 3) \tag{11}$$

$$\beta_{Ii} \sim N(0, 10) \tag{12}$$

$$\alpha_1 \sim N(0, 10) \tag{13}$$

$$\alpha_2 \sim N(0, 10) \tag{14}$$

The one prior to note is that a hierarchical regularizing prior is put on the varying testing adjustment parameters β_{qc} for regularization purposes due to the limited data available to inform the parameter.

Other than these weakly informative priors, the model is identified, as I show in the next section. However, it is important to emphasize that there is no information in the model that identifies the *true* number of infected

[1] For a discussion, see https://www.realclearpolitics.com/articles/2020/03/18/the_perils_of_mass_coronavirus_testing_142693.html.

people. Rather, the infection rate is a latent process, and as such it is not known exactly what scale to assign to it without further information. However, both the relative growth in infection rates are identified, along with the effect of suppression measures, so *the model is useful without being fully identified.* Furthermore, by incorporating insights from SIR/SEIR models I can also identify the latent scale with reasonable informative priors, as I show in the data analysis section. The SIR/SEIR models are not "doomsday" predictions but rather rigorous models of the underlying infection process.

1.2 Simulation

Because this model is fully generative, we can simulate it using Monte Carlo methods. The simulation is very important as it is the only way to demonstrate that the model is globally identified and can in fact capture unobserved parameters like suppression effects and relative infection rates. The following R code generates data from the model and plots the resulted unobserved infection rate and observed values for tests and cases:

```
# simulation parameters
num_country <- 100
# unobserved country-level heterogeneity for outbreak onset
country_int <- c(0,sort(rnorm(num_country-1,0,.25)))
time_points <- 100
# allows for linear growth that later becomes explosive
polynomials <- c(.03,0.007,-0.0001)

# factor that determines how many people a country is willing/able to test

country_test <- rnorm(num_country,5,0.25)

# size of countries

country_pop <- rpois(num_country,10000)

# assumt t=1 is unmodeled = exogenous start of the infection

t1 <- c(1,rep(0,num_country-1))
```

```r
# create a suppression coefficient
# first is for preventing domestic transmission from occuring
# second is for preventing further domestic transmission once it starts

suppress1 <- -0.8
suppress2 <- -0.05

# high value of phi = high over-time stability
phi <- c(300,300)

# countries that suppress or don't suppress

suppress_measures <- as.numeric(runif(num_country)>0.5)

# parameter governing how hard it is to find infected people and test them
# strictly positive

finding <- 1.5

# recursive function to generate time-series data by country

out_poly <- function(time_pt,end_pt,time_counts,tested,case_count,rate_infected,pr_domestic) {

  if(time_pt==1) {
    time_counts <- as.matrix(c(1,rep(0,num_country-1)))
    rate_infected <- as.matrix(c(.0001,rep(0,num_country-1)))
    tested <- as.matrix(rep(0,num_country))
    case_count <- as.matrix(c(1,rep(0,num_country-1)))
  }

  # if at time = t infected, start time tracker at t
  # need to know how many countries have reported at least one case = infection start

  world_count <- sum(case_count[,time_pt]>0)
```

```
if(time_pt==1) {

  rate_infected_new <-  plogis(-5 + time_counts[,time_pt]*polynomials[1] +
                                 suppress1*suppress_measures +
                 suppress2*suppress_measures*time_counts[,time_pt] +
                 .05*sum(world_count) +
    (time_counts[,time_pt]^2)*polynomials[2] +
      (time_counts[,time_pt]^3)*polynomials[3])

  # conservative time counter that only starts when first case is recorded

  time_counts_new <- ifelse(time_counts[,time_pt]>0 | case_count[,time_pt]>0,time_counts[,time_pt]+1,

} else {

  rate_infected_new <- plogis(-5 + time_counts[,time_pt]*polynomials[1] +
                 suppress1*suppress_measures +
                 suppress2*suppress_measures*time_counts[,time_pt] +
                 .05*sum(world_count) +
    (time_counts[,time_pt]^2)*polynomials[2] +
      (time_counts[,time_pt]^3)*polynomials[3])

  # conservative time counter that only starts when first case is recorded

  time_counts_new <- ifelse(time_counts[,time_pt]>0 | case_count[,time_pt]>0,time_counts[,time_pt]+1,
}

# of these, need to calculated a set number tested
mu_test <- plogis(-7 + country_test*rate_infected_new)
tested_new <- rbbinom(num_country,country_pop,mu_test*phi,(1-mu_test)*phi)

# determine case count as percentage number tested
```

```
  # this is what we always observe
  mu_case <- plogis(-2.19 + finding*rate_infected_new)
  case_count_new <- rbbinom(num_country,tested_new,mu_case*phi,(1-mu_case)*phi)



  if(time_pt<end_pt) {
    out_poly(time_pt=time_pt+1,
             end_pt=end_pt,
             time_counts=cbind(time_counts,
                               time_counts_new),
             rate_infected=cbind(rate_infected,rate_infected_new),
             tested=cbind(tested,tested_new),
             case_count=cbind(case_count,case_count_new))
  } else {
    return(list(time_counts=time_counts,
                tested=tested,
                rate_infected=rate_infected,
                case_count=case_count))
  }

}

check1 <- out_poly(1,time_points)

check1 <- lapply(check1, function(c) {
  colnames(c) <- as.numeric(1:time_points)
  c
})

all_out <- bind_rows(list(time_counts=as_tibble(check1$time_counts),
                          `Proportion Population\nInfected`=as_tibble(check1$rate_infected),
                          `Number of Cases`=as_tibble(check1$case_count),
                          `Proportion of Cases from Domestic Transmission`=as_tibble(check1$pr_domestic
                          `Number of Tests`=as_tibble(check1$tested)),.id="Series")
```

```r
all_out$country <- rep(paste0("country_",1:num_country),times=length(check1))
all_out$suppress_measures <- factor(rep(suppress_measures,times=length(check1)),labels=c("No","Yes"))

all_out %>%
  gather(key = "time_id",value="indicator",-Series,-country,-suppress_measures) %>%
  mutate(time_id=as.numeric(time_id)) %>%
  filter(!(Series %in% c("time_counts"))) %>%
  ggplot(aes(y=indicator,x=time_id)) +
  geom_line(aes(colour=suppress_measures,group=country),alpha=0.3) +
  xlab("Days Since Outbreak") +
  ylab("") +
  facet_wrap(~Series,scales="free_y") +
  theme(panel.background = element_blank(),
        panel.grid=element_blank(),
        strip.background = element_blank(),
        strip.text = element_text(face="bold"),
        legend.position = "top")
```

Figure S1 shows one line for each country's trajectory from a total of 100 countries and 100 time points. As can be seen, the green lines indicating suppression effects and the red lines indicating no suppression diverge substantially over time. However, the numbers of observed tests and cases show far more random noise due to the difficulty in inferring the true rate from the observed data. It is possible that some countries simply want to test more, and end up with more cases, or that the infection rate is in fact higher. As such, this model is able to incorporate that measurement uncertainty between the true (unobserved) rate and the observed indicators, tests and cases.

The advantage of this model is that it allows for the testing of covariates that affect the true infection rate without requiring more heavy-duty approaches like SEIR/SIR. The intention is to have a more parsimonious model that can see how the effect of variables like suppression measures have on different countries/states infection numbers over time. I am currently collecting this data as part of the CoronaNet project.

The model could be further extended with more complicated initial infection processes and inter-country transmission processes, such as spatial modeling, but for the purposes of this exposition I do not look further at such extensions.

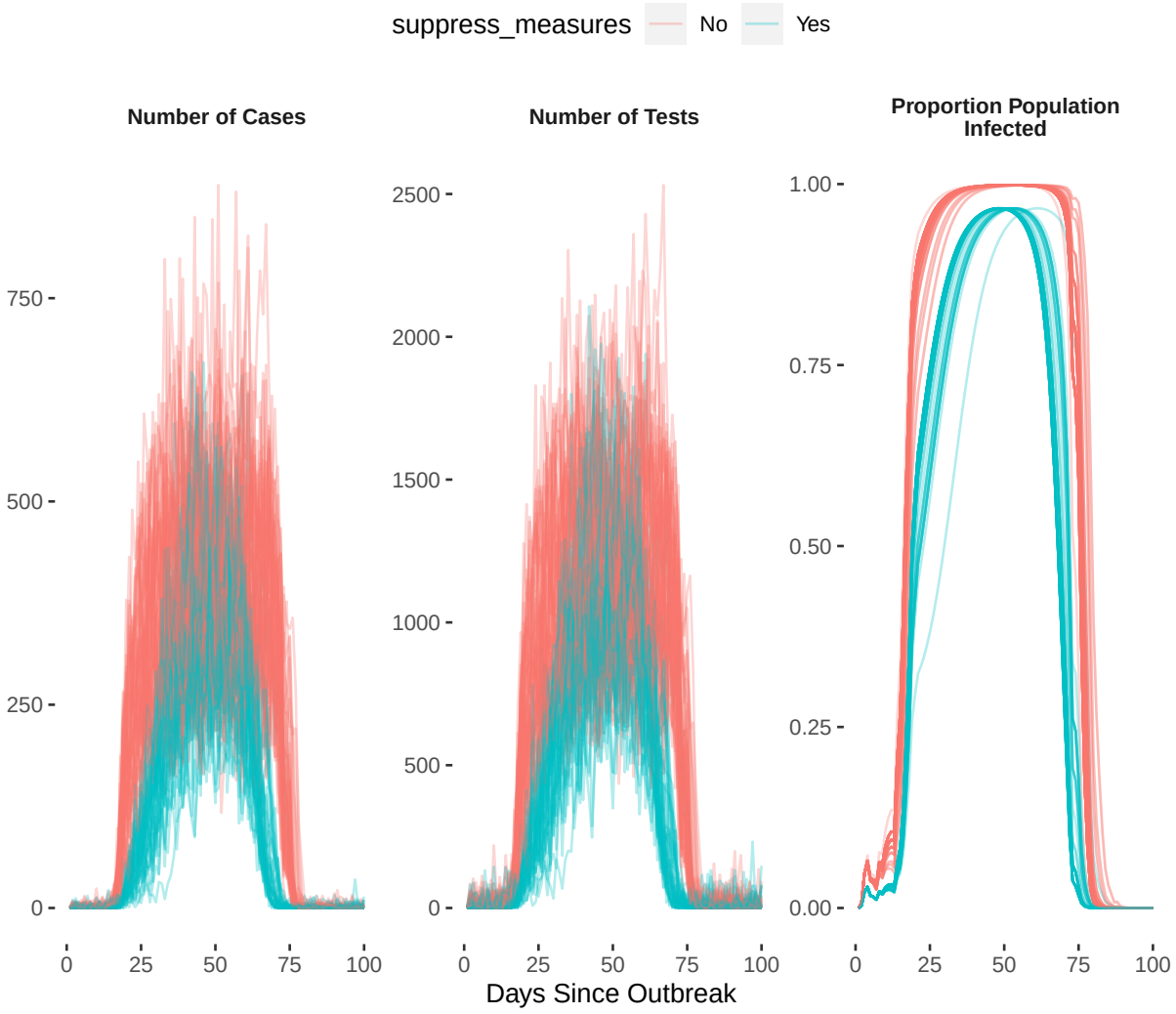

Fig. S1: Simulation of Observed Tests and Cases Given Unobserved Infectious Process

1.3 Estimation

I can then fit an empirical model using Stan, a Markov Chain Monte Carlo sampler, to model the unobserved infection rate given the simulated observed data:

```
# all data from simulation
# primarily case and test counts

# need to make centered, ortho-normal polynomials

ortho_time <- poly(scale(1:time_points),degree=3)

init_vals <- function() {
  list(phi1=100,
       phi2=100)
}

sim_data <- list(time_all=time_points,
                 num_country=num_country,
                 country_pop=country_pop,
                 cases=check1$case_count,
                 phi_scale=1/300,
                 ortho_time=ortho_time,
                 tests=check1$tested,
                 count_outbreak=as.numeric(scale(apply(check1$time_counts,2,function(c) sum(c>0)))),
                 time_outbreak=check1$time_counts,
                 suppress=suppress_measures)

if(run_model) {

  pan_model <- stan_model("corona_tscs_betab.stan")

# run model

pan_model_est <- sampling(pan_model,data=sim_data,chains=2,cores=2,iter=1200,warmup=800,init=init_vals)
```

```
saveRDS(pan_model_est,"data/pan_model_est.rds")
} else {
  pan_model_est <- readRDS("data/pan_model_est.rds")
}
```

Now I can access the estimated infection rates and plot them:

```
all_est <- as.data.frame(pan_model_est,"num_infected_high") %>%
  mutate(iter=1:n()) %>%
  gather(key="variable",value="estimate",-iter) %>%
  group_by(variable) %>%
  mutate(estimate=plogis(estimate)) %>%
  summarize(med_est=quantile(estimate,.5),
            high_est=quantile(estimate,.95),
            low_est=quantile(estimate,.05)) %>%
  mutate(country_num=as.numeric(str_extract(variable,"(?<=\\[)[1-9][0-9]?0?")),
         time_point=as.numeric(str_extract(variable,"[1-9][0-9]?0?(?=\\])")))

all_est <- left_join(all_est,tibble(country_num=1:num_country,
                                    suppress_measures=factor(suppress_measures,labels=c("No","Yes"))),by

all_est %>%
  ggplot(aes(y=med_est,x=time_point)) +
  geom_ribbon(aes(ymin=low_est,
  ymax=high_est,
  group=country_num,
  fill=suppress_measures),alpha=0.5) +
  theme_minimal() +
  scale_color_brewer(type="div") +
  ylab("Proportion of Population Infected") +
  xlab("Days Since Outbreak Start") +
  theme(panel.grid = element_blank(),
        legend.position = "top")
```

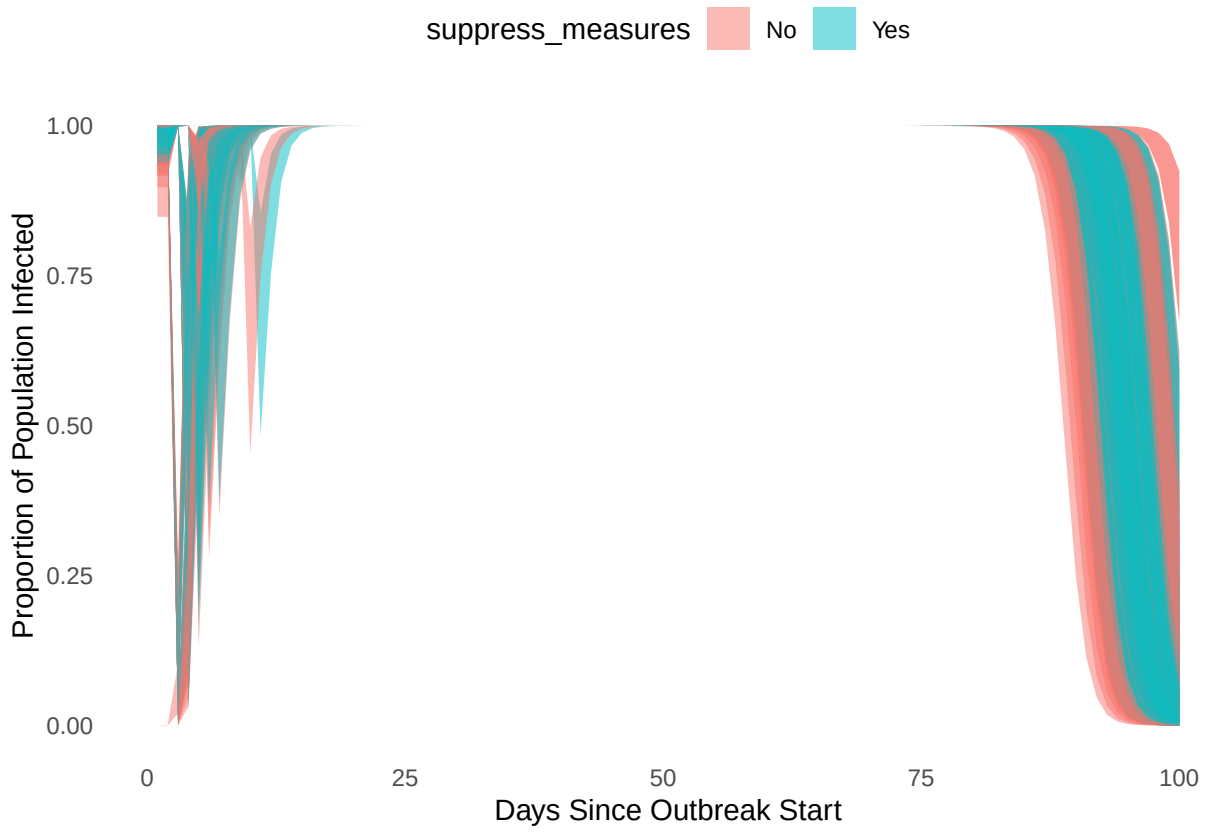

Fig. S2: Estimated Simulated Infection Rates

We see very good recovery of the estimates from the model given the observed data in Figure S2. The estimates disappear near the top of the plot as the true infected rate approaches 100% and there is little variability in the data. This plots shows the 5% - 95% high posterior density interval over time for each country. As can be seen, the suppression effect can be clearly distinguished even with residual uncertainty from not directly observing the infection rate, i.e., "flattening the curve." We also can learn what the effect of the suppression measure is:

```
require(bayesplot)


mcmc_hist(as.array(pan_model_est,c("suppress_effect[1]","suppress_effect[2]")))
```

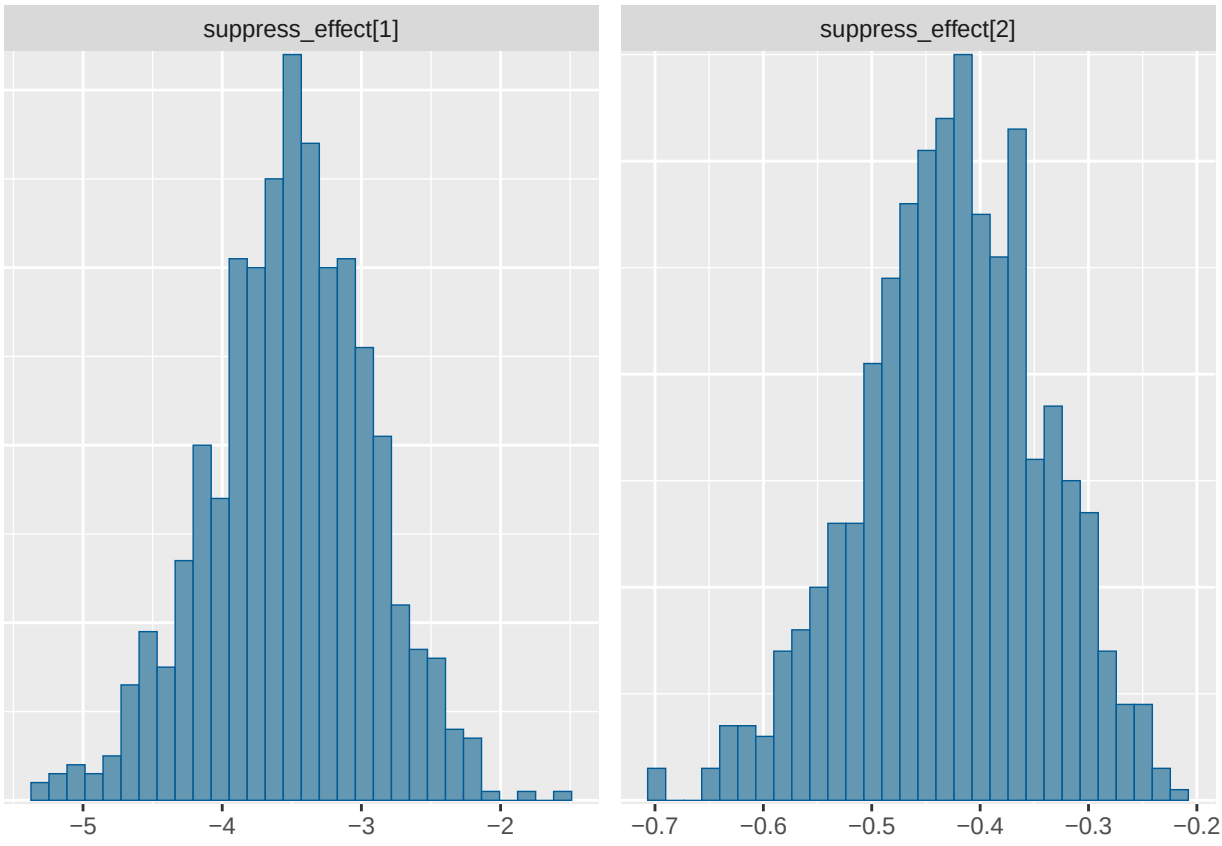

Fig. S3: Recovered Simulated Virus Suppression Parameters

We can see in Figure S3 despite the uncertainty in not perfectly observing the infection rate, we can still get a precise credible interval on the suppression effect. The effect is not the same as the "true" value due to the use of orthonormal polynomials for estimation purposes, but the effect is clearly distinguishable.

The advantage of this model, as can be seen, is that with testing numbers and case counts, we can model the

effect of country-level (or region-level) variables on the unseen infection rate up to an unknown constant. It is far simpler than existing approaches while still permitting inference on these hard-to-quantify measures. It is no substitute for infectious disease modeling–for example, it produces no estimates of the susceptible versus recovered individuals in the population, nor death rates–but rather a way to measure the effect of different suppressive and social-distancing measures on disease outcomes as well as approximate disease trajectory.

There is, however, an important caveat to be made. The infection rate here is modeled as a latent variable, and as such the scale is largely determined by the priors. In this simulation we were able to know a priori what the relationship between all the coefficients and the true infection rate is, but in an applied setting, as I discuss next, we will not be able to do so without making further assumptions based on SIR/SEIR models. However, what is important is that the sign and relative rank of suppression covariates is identified *even if the true scale of the latent infection rate is unknown.*

2 Identifying the Latent Scale

To show how we can further attempt to identify the scale of the latent variable, instead of only relative differences between states, I show in this section how we can add in information from SEIR/SIR modeling to identify the total number of infected persons in the model. The crucial missing piece of information in the model is the ratio between the proportion of infected persons I_{ct} and the proportion of tests per state population q_{ct}:

$$\frac{q_{ct}}{I_{ct}} \tag{15}$$

Another way of framing the problem is to think of how much the number of tests should increase given a one-percentage increase in the infection rate. How many of these infected people will be tested? Without an idea of the true number of infected people, it is impossible to answer this question and thus identify the latent scale.

However, an increasing number of SIR/SEIR papers show that it is likely that as few as 10% of the total number of infected persons are actually recorded as cases, including in the United States. This number provides us with a conservative lower bound if we consider that the number of tests should be at least 10% of the total proportion of infected persons. In other words, we can consider adding the following information into the model:

$$\frac{q_{ct}}{I_{ct}} > 0.1 \tag{16}$$

Every percentage increase in the infection rate should result in at least a 0.1% increase in the total number of people tested as a percentage of the population. It is difficult to impose this constraint directly in the model; however, we can consider adding it as prior information if we can define a prior density over the ratio. First, for computational simplicity, we can consider a distribution over the log differences of the parameters:

$$\log q_{ct} - \log I_{ct} \tag{17}$$

By simulating from uniform distributions of possible rates for $\log q_{ct}$ and $\log I_{ct}$, it is possible to tell that the a realistic distribution of log differences with 0.1 as a lower bound is in fact very close a standard normal distribution:[2]

```
# this code generates a log distribution of a ratio of two probabilities.


f_y <- function(y, a, b, log = FALSE){
  lc <-  log(b-a)
  ans <- y  - lc
  if(!log) ans <- exp(ans)
  return(ans)
}


#
get_Y_var <- function(a, b){
  EXsq <- ( b*(log(b)^2 - 2*log(b) + 2) -  a*(log(a)^2 - 2*log(a) + 2) )/(b-a)
  EX <- (b*(log(b) - 1) - a*(log(a) - 1) ) /(b-a)
  ans <- EXsq - (EX)^2
  return(ans)
}
##
analytic_W <- function(w, a, b){  ## assumes i.i.d.
  c0 <- all(w > a/b, w < b/a)
  k <- (b-a)*(b-a)
  m <- max(a, a/w)
  M <- min(b, b/w)
```

[2] I thank Luiz Max Carvalho for providing this code.

```
  soln <- function(L, U) ((U *abs(U))- (L *abs(L)))/(2*k)
  d0 <- soln(L = m, U = M)
  dens <- c0 * d0
  return(dens)
}
analytic_W <- Vectorize(analytic_W)
##
f_z <- function(z, a, b, log = FALSE){
  ans <- log(analytic_W(exp(z), a, b)) + z
  if(!log) ans <- exp(ans)
  return(ans)
}
f_z <- Vectorize(f_z)
################

M <- 1E6
a <-  .01
b <-  .5
Y <- log(runif(M, min = a, max = b))

#hist(Y, probability = TRUE)
# curve(f_y(x, a, b), min(Y), max(Y), add = TRUE, lwd = 2)
# abline(v = c(log(a), log(b)), lwd = 2, lty = 2)
# integrate(function(x) f_y(x, a, b), log(a), log(b))

###

Y1 <- log(runif(M, min = a, max = b))
Y2 <- log(runif(M, min = a, max = b))

`Log Difference` <- Y1 - Y2

# integrate(function(x) f_z(x, a, b), log(a)-log(b), log(b)-log(a))
```

```r
# integrate(function(x) x^2* f_z(x, a, b), log(a)-log(b), log(b)-log(a))
#
# 2*get_Y_var(a, b)
# var(`Log Difference`)


#
hist(`Log Difference`, breaks = 50, probability = TRUE)
curve(f_z(x, a, b), min(`Log Difference`), max(`Log Difference`), col = 2, add = TRUE)
curve(dnorm(x, 0, sqrt(2*get_Y_var(a, b))), min(`Log Difference`), max(`Log Difference`), col = 3, add =
abline(v = c(log(a)-log(b), log(b)-log(a)), lwd = 2, lty = 2)
```

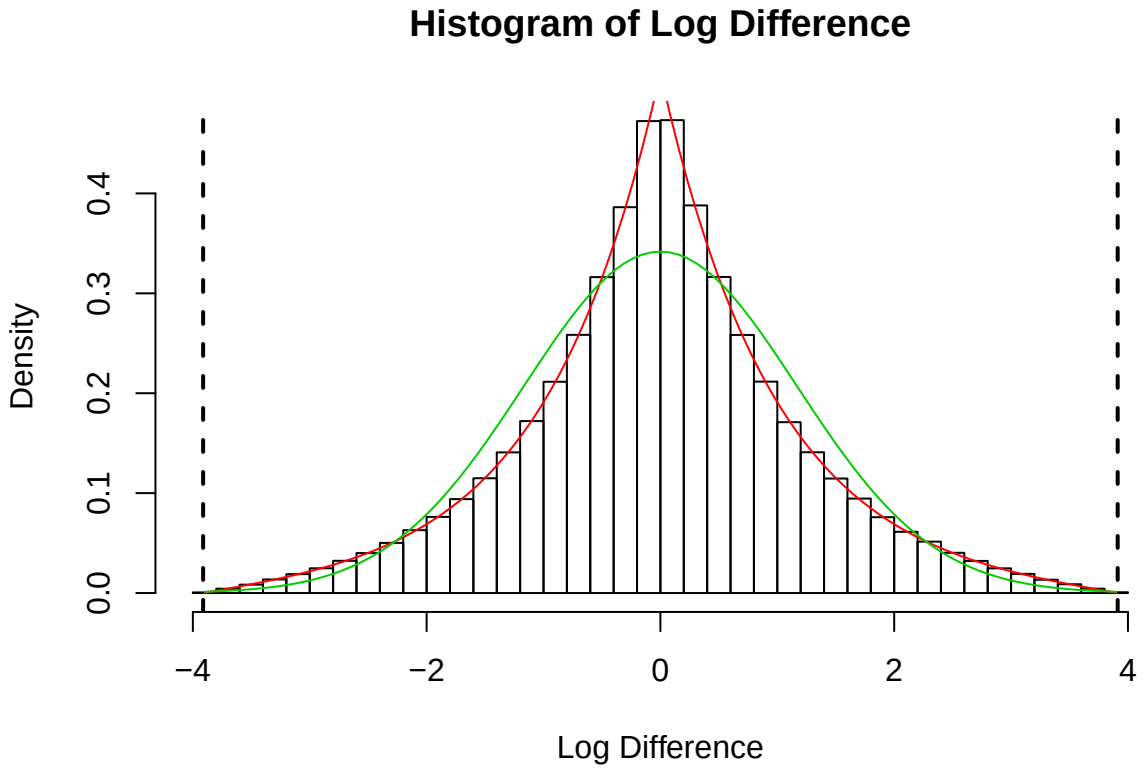

Fig. S4: Distribution of Log Difference in Probabilities

We can see in Figure S4 that the standard normal provides an approximate fit to the density of two probabilities where there is limited density on values below log − 2 or 0.14. On the other hand, the prior puts substantial mass on ratios of tests to infected that are quite high, such as log2, 7.38. This informative prior, which still allows for a range of possible ratios, allows me to use SEIR/SIR model conclusions while still

permitting uncertainty over the underlying relationship.

For these reasons, I add this term to the joint posterior for each time point t and country c:

$$\log q_{ct} - \log I_{ct} \sim N(0, 1) \tag{18}$$

I also add a log Jacobian adjustment to the posterior density to reflect the fact that this prior is a non-linear function of parameters I have already assigned priors to:

$$\log(1 - q_{ct}) + \log(1 - I_{ct}) \tag{19}$$

The Jacobian adjustment is derived as the joint derivative of the non-linear functions of the underlying parameters, which in this case are the natural log and the inverse logit function.