Don’t Feed the Bugs

Reproducibility for Health Policy Research

Alyssa Bilinski
April 3, 2024

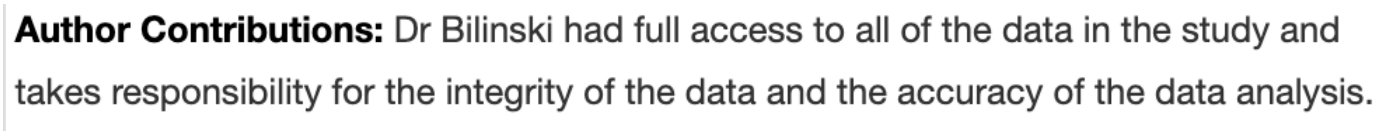
Does your code work?

Ya sure?

…how do you know?

Does your code work?

I should hope so.



Author Contributions: Dr Bilinski had full access to all of the data in the study and takes responsibility for the integrity of the data and the accuracy of the data analysis.

Ya sure?

…how do you know?

Well, it runs.
I checked and double-checked it.
The results "make sense."

1 **Complex code underpins much of our research.**

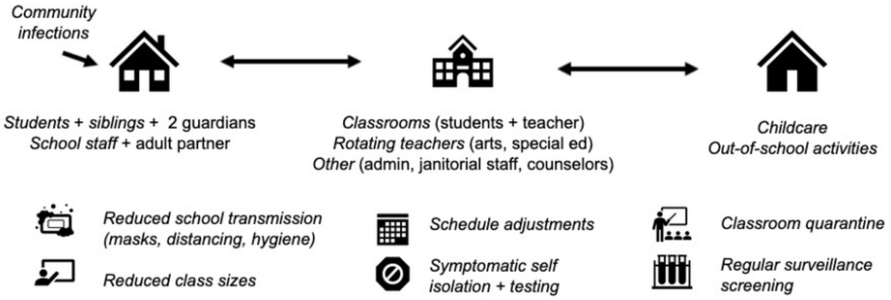
2 **In research, we implicitly assume that code meets quality standards because:**

Researchers really care.

There are many eyes on the output (co-authors, reviewers).

(Sometimes) code is public.

This felt inadequate.



Community infections

Students + *siblings* + 2 guardians
School staff + adult partner

Classrooms (students + teacher)
Rotating teachers (arts, special ed)
Other (admin, janitorial staff, counselors)

Childcare
Out-of-school activities

Reduced school transmission (masks, distancing, hygiene)

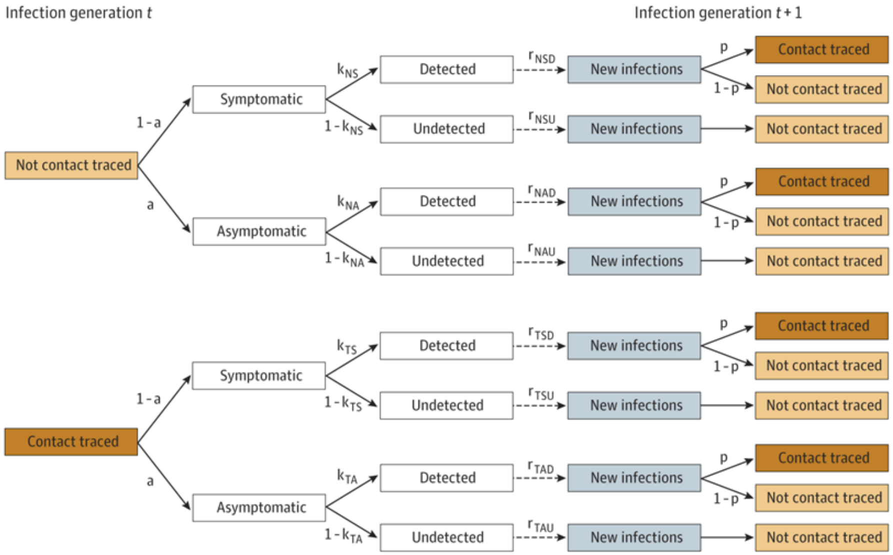
Reduced class sizes

Schedule adjustments

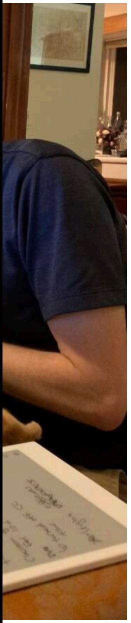
Symptomatic self isolation + testing

Classroom quarantine

Regular surveillance screening



This felt inadequate.



So, I called a software engineer.

Software engineers are:

- Formally trained to write code
- Primarily focused on writing high quality code

So, I called a software engineer.

Researchers	Software Engineers
Code is the easy part.	Code is a very hard part.
Check outputs.	Test each unit.
One or two coders	Highly collaborative
Final version	Iterative

Today, I’ll argue that our approach to reproducibility should adapt insights from software engineering.

The current paradigm is insufficient to support high-quality research.

Even when it “works out,” it’s both hard on analysts and hard to communicate quality-assurance procedures.

Today, I'll argue that our approach to reproducibility should adapt insights from software engineering.

We'll be keeping it simple.

Does your code do what you want/think it does?
Do results get to where they need to go?

Collaborators

John Giardina
Massachusetts General Research Institute

Luke Massa
Tripadvisor

Gray Babbs
Brown University

Mindset

In research, coding can seem "easy" relative to generating a question and learning statistical methods.

But actually…writing code that does what you want is **really, really hard**.

Mindset

There are always bugs in your code.

The key is a system that avoids or stomps on the **ones that matter**.

Mindset

1 **Minimize opportunities for errors.**

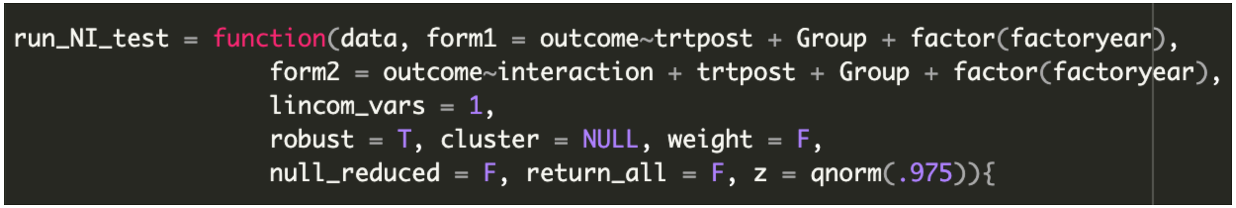
2 **Assume guilty until proven innocent.**

Mindset

1 Minimize opportunities for errors.

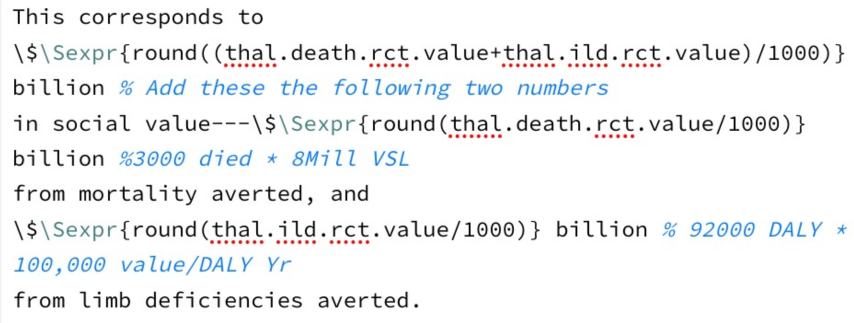
Avoid copy/paste.

Functions



```
run_NI_test = function(data, form1 = outcome~trtpost + Group + factor(factoryear),
                       form2 = outcome~interaction + trtpost + Group + factor(factoryear),
                       lincom_vars = 1,
                       robust = T, cluster = NULL, weight = F,
                       null_reduced = F, return_all = F, z = qnorm(.975)){
```

Automatic updates



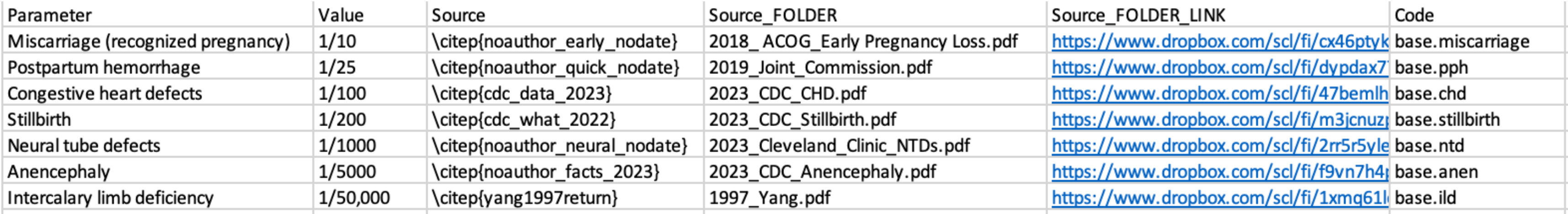
```
This corresponds to
\$\Sexpr{round((thal.death.rct.value+thal.ild.rct.value)/1000)}
billion % Add these the following two numbers
in social value---\$\Sexpr{round(thal.death.rct.value/1000)}
billion %3000 died * 8Mill VSL
from mortality averted, and
\$\Sexpr{round(thal.ild.rct.value/1000)} billion % 92000 DALY *
100,000 value/DALY Yr
from limb deficiencies averted.
```

Mindset

1 Minimize opportunities for errors.

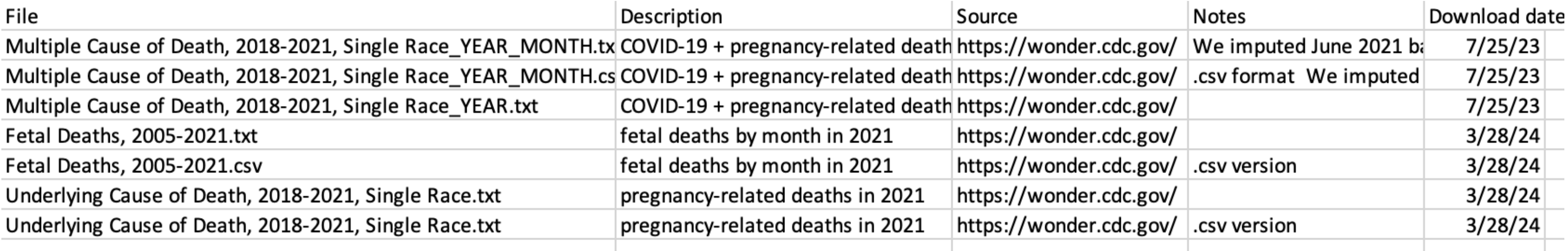
When unavoidable,
make checking easy.

Linked source data



Parameter	Value	Source	Source_FOLDER	Source_FOLDER_LINK	Code
Miscarriage (recognized pregnancy)	1/10	\citep{noauthor_early_nodate}	2018_ ACOG_Early Pregnancy Loss.pdf	https://www.dropbox.com/scl/fi/cx46ptyk	base.miscarriage
Postpartum hemorrhage	1/25	\citep{noauthor_quick_nodate}	2019_Joint_Commission.pdf	https://www.dropbox.com/scl/fi/dypdax7	base.pph
Congestive heart defects	1/100	\citep{cdc_data_2023}	2023_CDC_CHD.pdf	https://www.dropbox.com/scl/fi/47bemlh	base.chd
Stillbirth	1/200	\citep{cdc_what_2022}	2023_CDC_Stillbirth.pdf	https://www.dropbox.com/scl/fi/m3jcnuz	base.stillbirth
Neural tube defects	1/1000	\citep{noauthor_neural_nodate}	2023_Cleveland_Clinic_NTDs.pdf	https://www.dropbox.com/scl/fi/2rr5r5yle	base.ntd
Anencephaly	1/5000	\citep{noauthor_facts_2023}	2023_CDC_Anencephaly.pdf	https://www.dropbox.com/scl/fi/f9vn7h4	base.anen
Intercalary limb deficiency	1/50,000	\citep{yang1997return}	1997_Yang.pdf	https://www.dropbox.com/scl/fi/1xmq61l	base.ild

ReadMes



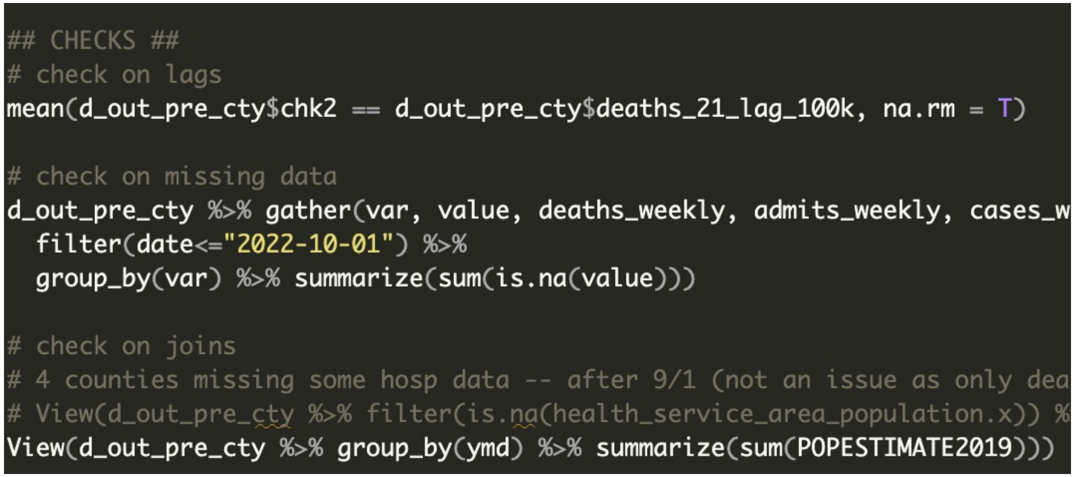
File	Description	Source	Notes	Download date
Multiple Cause of Death, 2018-2021, Single Race_YEAR_MONTH.tx	COVID-19 + pregnancy-related death	https://wonder.cdc.gov/	We imputed June 2021 ba	7/25/23
Multiple Cause of Death, 2018-2021, Single Race_YEAR_MONTH.cs	COVID-19 + pregnancy-related death	https://wonder.cdc.gov/	.csv format We imputed	7/25/23
Multiple Cause of Death, 2018-2021, Single Race_YEAR.txt	COVID-19 + pregnancy-related death	https://wonder.cdc.gov/		7/25/23
Fetal Deaths, 2005-2021.txt	fetal deaths by month in 2021	https://wonder.cdc.gov/		3/28/24
Fetal Deaths, 2005-2021.csv	fetal deaths by month in 2021	https://wonder.cdc.gov/	.csv version	3/28/24
Underlying Cause of Death, 2018-2021, Single Race.txt	pregnancy-related deaths in 2021	https://wonder.cdc.gov/		3/28/24
Underlying Cause of Death, 2018-2021, Single Race.txt	pregnancy-related deaths in 2021	https://wonder.cdc.gov/	.csv version	3/28/24

Mindset

2 Assume guilty until proven innocent.

What could have gone wrong, and how could I check that it didn't?

- Check range of values
  - Missing variables: (NA, "N/A", "", " ")
  - Typos/misspellings in strings
  - Unreasonable estimates (e.g., negative age)
- Check sample size before/after merges or filtering
  - Accidental drops
  - Accidental duplications



```
## CHECKS ##
# check on lags
mean(d_out_pre_cty$chk2 == d_out_pre_cty$deaths_21_lag_100k, na.rm = T)

# check on missing data
d_out_pre_cty %>% gather(var, value, deaths_weekly, admits_weekly, cases_w
  filter(date<="2022-10-01") %>%
  group_by(var) %>% summarize(sum(is.na(value)))

# check on joins
# 4 counties missing some hosp data -- after 9/1 (not an issue as only dea
# View(d_out_pre_cty %>% filter(is.na(health_service_area_population.x)) %
View(d_out_pre_cty %>% group_by(ymd) %>% summarize(sum(POPESTIMATE2019)))
```

Testing

I learned to rely on two indicators of code quality:

1) Did it run?
2) Do my results look weird?

These are important! But they are pretty **ad hoc**.

Testing

Software engineers formally **test** each **unit**, or chunk of code as well as how they fit together.

1. **Define tasks that each function (or set of functions) should complete.**
2. **Design tests to ensure that you receive expected outputs given a set of inputs.**
3. **Run tests over different sets of inputs.**

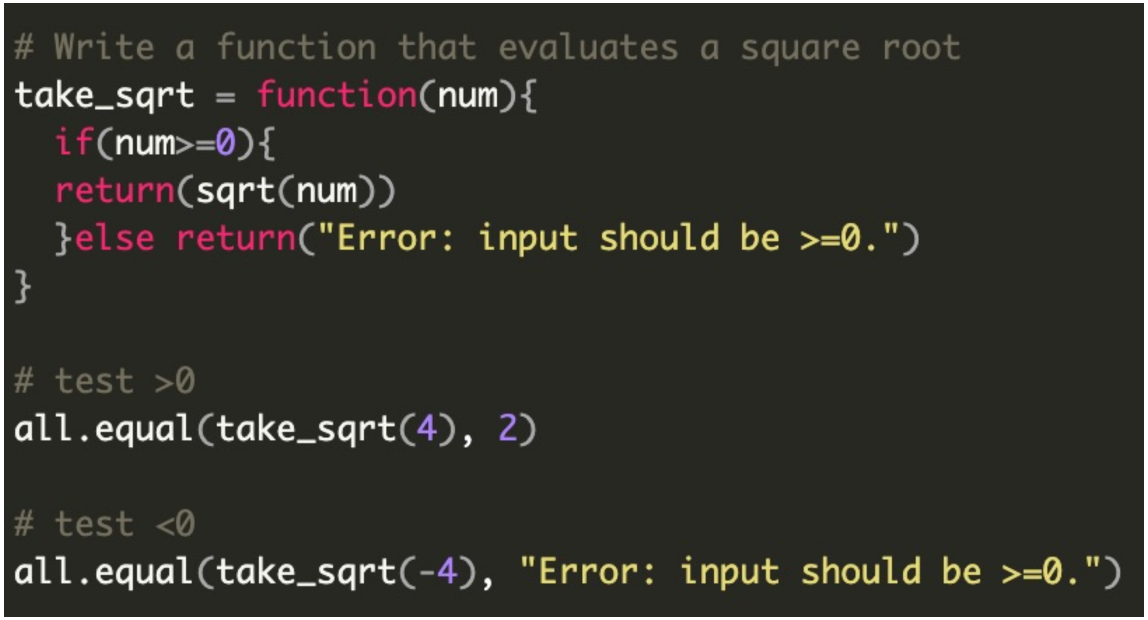
Testing

Testing is **rarely straightforward**.

Writing a sufficient set of tests is a **skill developed over time**.

Testing

Case #0 (toy): Square roots.



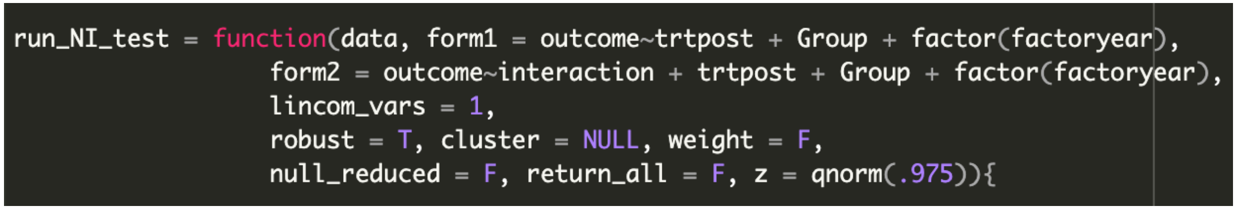
```r
# Write a function that evaluates a square root
take_sqrt = function(num){
  if(num>=0){
  return(sqrt(num))
  }else return("Error: input should be >=0.")
}

# test >0
all.equal(take_sqrt(4), 2)

# test <0
all.equal(take_sqrt(-4), "Error: input should be >=0.")
```

Testing

Case #1 (easy): Fast covariance matrices.

We developed a function to:

1) Adapt normal-based covariance matrices for a specific context
2) Speed matrix multiplication compared to standard estimates



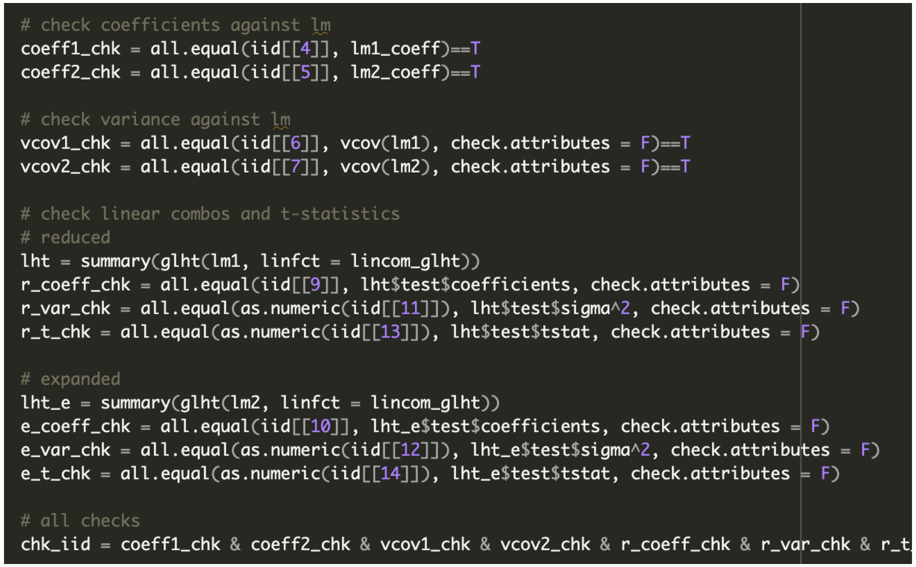
```
run_NI_test = function(data, form1 = outcome~trtpost + Group + factor(factoryear),
                       form2 = outcome~interaction + trtpost + Group + factor(factoryear),
                       lincom_vars = 1,
                       robust = T, cluster = NULL, weight = F,
                       null_reduced = F, return_all = F, z = qnorm(.975)){
```

Testing

Case #1 (easy): Fast covariance matrices.

~ 50 tests (**testthat** in R)

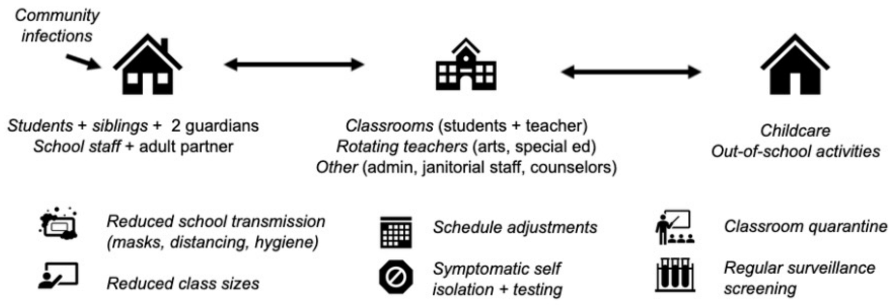
1) Simulate random data
2) For each simulated dataset and covariance option:
    1) Check that coefficients, variances, linear combinations matched alternative commands when inputs were set to standard case
    2) For non-standard cases, re-do estimates manually



```r
# check coefficients against lm
coeff1_chk = all.equal(iid[[4]], lm1_coeff)==T
coeff2_chk = all.equal(iid[[5]], lm2_coeff)==T

# check variance against lm
vcov1_chk = all.equal(iid[[6]], vcov(lm1), check.attributes = F)==T
vcov2_chk = all.equal(iid[[7]], vcov(lm2), check.attributes = F)==T

# check linear combos and t-statistics
# reduced
lht = summary(glht(lm1, linfct = lincom_glht))
r_coeff_chk = all.equal(iid[[9]], lht$test$coefficients, check.attributes = F)
r_var_chk = all.equal(as.numeric(iid[[11]]), lht$test$sigma^2, check.attributes = F)
r_t_chk = all.equal(as.numeric(iid[[13]]), lht$test$tstat, check.attributes = F)

# expanded
lht_e = summary(glht(lm2, linfct = lincom_glht))
e_coeff_chk = all.equal(iid[[10]], lht_e$test$coefficients, check.attributes = F)
e_var_chk = all.equal(as.numeric(iid[[12]]), lht_e$test$sigma^2, check.attributes = F)
e_t_chk = all.equal(as.numeric(iid[[14]]), lht_e$test$tstat, check.attributes = F)

# all checks
chk_iid = coeff1_chk & coeff2_chk & vcov1_chk & vcov2_chk & r_coeff_chk & r_var_chk & r_t
```

Testing

Case #2 (af3g%^&*): Complex simulation models

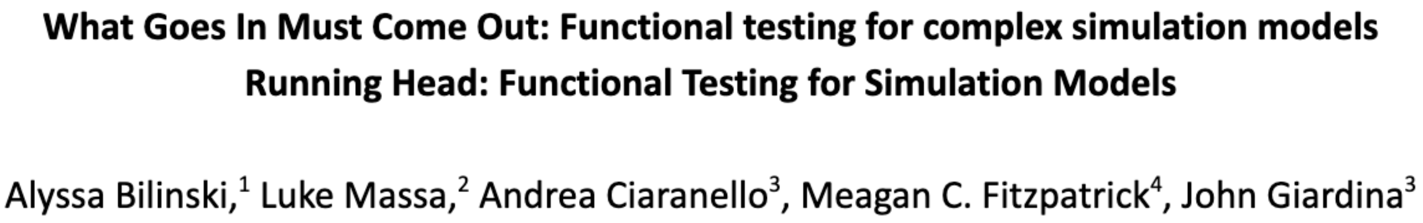

Community infections
Students + siblings + 2 guardians
School staff + adult partner
Classrooms (students + teacher)
Rotating teachers (arts, special ed)
Other (admin, janitorial staff, counselors)
Childcare
Out-of-school activities
Reduced school transmission (masks, distancing, hygiene)
Reduced class sizes
Schedule adjustments
Symptomatic self isolation + testing
Classroom quarantine
Regular surveillance screening

Goal: Write a sufficient set of tests to trust my model.
Roadblocks:

1) I'm writing a simulation model because I don't know expected outputs for a set of inputs.
2) I'm stringing together a lot of functions that may behave oddly even if each unit test passes

Testing

Case #2 (af3g%^&*): Complex simulation models



What Goes In Must Come Out: Functional testing for complex simulation models

Running Head: Functional Testing for Simulation Models

Alyssa Bilinski,[1] Luke Massa,[2] Andrea Ciaranello[3], Meagan C. Fitzpatrick[4], John Giardina[3]

1) **Collate input parameters**
   - Table 1 + structural parameters

2) **Define and track intermediate outputs**
   - Add intermediate "napkin" outputs for each input that can reverse engineer input behavior

3) **Run and report test results over different input combinations**

Testing

Case #2 (af3g%^&*): Complex simulation models

Overdispersion parameter
Wildtype COVID-19 was "overdispersed" → heterogeneous individual infectiousness.

- Implemented as a multiplier on individual attack rate

To track attack rates:
1) Track number of infectious individuals in each day in each setting.
2) Track number of contacts per day in each setting.

Testing

Case #2 (af3g%^&*): Complex simulation models

Overdispersion parameter
Wildtype COVID-19 was "overdispersed" → heterogeneous individual infectiousness.

- Implemented as a multiplier on individual attack rate

To track attack rates:
1) Track number of infectious individuals in each day in each setting.
2) Track number of contacts per day in each setting.

We observed a slight underestimate in attack rate:
1) only in households
2) only with overdispersion turned on

In rare cases, overdispersion could push the attack rate > 1 → NA.

Testing

Case #2 (af3g%^&*): Complex simulation models

Tests improve transparency.

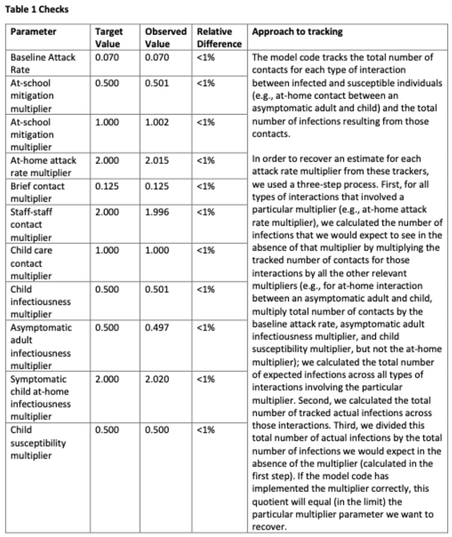
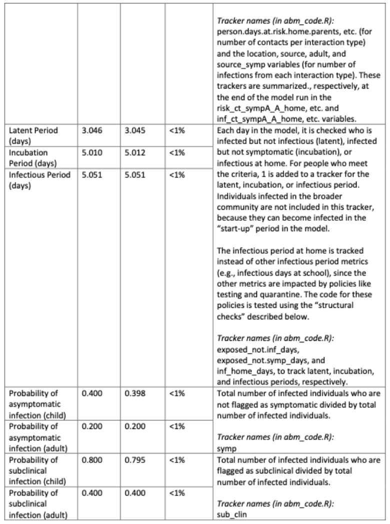
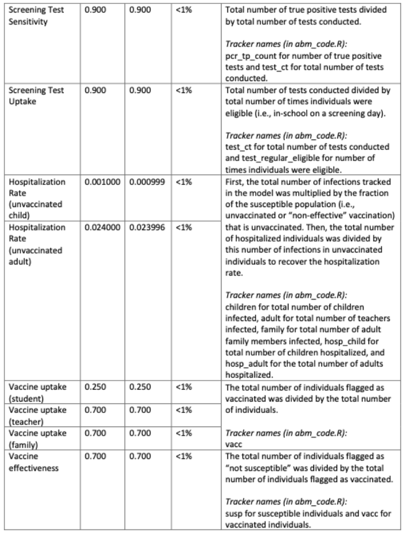


Table 1 Checks

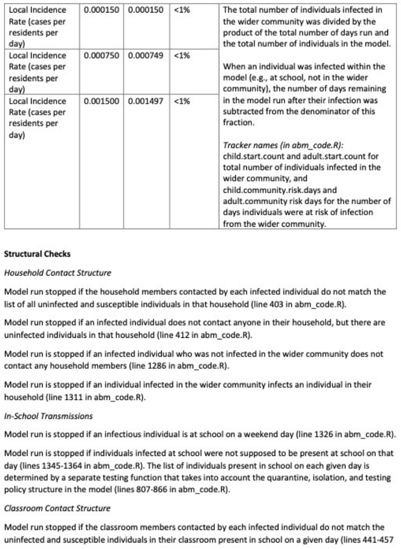
Parameter	Target Value	Observed Value	Relative Difference	Approach to tracking
Baseline Attack Rate	0.070	0.070	<1%	The model code tracks the total number of contacts for each type of interaction between infected and susceptible individuals (e.g., at-home contact between an asymptomatic adult and child) and the total number of infections resulting from those contacts. In order to recover an estimate for each attack rate multiplier from these trackers, we used a three-step process. First, for all types of interactions that involved a particular multiplier (e.g., at-home attack rate multiplier), we calculated the number of infections that we would expect to see in the absence of that multiplier by multiplying the tracked number of contacts for those interactions by all the other relevant multipliers (e.g., for at-home interaction between an asymptomatic adult and child, multiply total number of contacts by the baseline attack rate, asymptomatic adult infectiousness multiplier, and child susceptibility multiplier, but not the at-home multiplier); we calculated the total number of expected infections across all types of interactions involving the particular multiplier. Second, we calculated the total number of tracked actual infections across those interactions. Third, we divided this total number of actual infections by the total number of infections we would expect in the absence of the multiplier (calculated in the first step). If the model code has implemented the multiplier correctly, this quotient will equal (in the limit) the particular multiplier parameter we want to recover.
At-school mitigation multiplier	0.500	0.501	<1%	
At-school mitigation multiplier	1.000	1.002	<1%	
At-home attack rate multiplier	2.000	2.015	<1%	
Brief contact multiplier	0.125	0.125	<1%	
Staff-staff contact multiplier	2.000	1.996	<1%	
Child care contact multiplier	1.000	1.000	<1%	
Child infectiousness multiplier	0.500	0.501	<1%	
Asymptomatic adult infectiousness multiplier	0.500	0.497	<1%	
Symptomatic child at-home infectiousness multiplier	2.000	2.020	<1%	
Child susceptibility multiplier	0.500	0.500	<1%	



				Tracker names (in abm_code.R): person.days.at.risk.home.parents, etc. (for number of contacts per interaction type) and the location, source, adult, and source_symp variables (for number of infections from each interaction type). These trackers are summarized., respectively, at the end of the model run in the risk_ct_sympA_A_home, etc. and inf_ct_sympA_A_home, etc. variables.
Latent Period (days)	3.046	3.045	<1%	Each day in the model, it is checked who is infected but not infectious (latent), infected but not symptomatic (incubation), or infectious at home. For people who meet the criteria, 1 is added to a tracker for the latent, incubation, or infectious period. Individuals infected in the broader community are not included in this tracker, because they can become infected in the "start-up" period in the model. The infectious period at home is tracked instead of other infectious period metrics (e.g., infectious days at school), since the other metrics are impacted by policies like testing and quarantine. The code for these policies is tested using the "structural checks" described below. *Tracker names (in abm_code.R):* exposed_not.inf_days, exposed_not.symp_days, and inf_home_days, to track latent, incubation, and infectious periods, respectively.
Incubation Period (days)	5.010	5.012	<1%	
Infectious Period (days)	5.051	5.051	<1%	
Probability of asymptomatic infection (child)	0.400	0.398	<1%	Total number of infected individuals who are not flagged as symptomatic divided by total number of infected individuals. *Tracker names (in abm_code.R):* symp
Probability of asymptomatic infection (adult)	0.200	0.200	<1%	
Probability of subclinical infection (child)	0.800	0.795	<1%	Total number of infected individuals who are flagged as subclinical divided by total number of infected individuals. *Tracker names (in abm_code.R):* sub_clin
Probability of subclinical infection (adult)	0.400	0.400	<1%	



Screening Test Sensitivity	0.900	0.900	<1%	Total number of true positive tests divided by total number of tests conducted. *Tracker names (in abm_code.R):* pcr_tp_count for number of true positive tests and test_ct for total number of tests conducted.
Screening Test Uptake	0.900	0.900	<1%	Total number of tests conducted divided by total number of times individuals were eligible (i.e., in-school on a screening day). *Tracker names (in abm_code.R):* test_ct for total number of tests conducted and test_regular_eligible for number of times individuals were eligible.
Hospitalization Rate (unvaccinated child)	0.001000	0.000999	<1%	First, the total number of infections tracked in the model was multiplied by the fraction of the susceptible population (i.e., unvaccinated or "non-effective" vaccination) that is unvaccinated. Then, the total number of hospitalized individuals was divided by this number of infections in unvaccinated individuals to recover the hospitalization rate. *Tracker names (in abm_code.R):* children for total number of children infected, adult for total number of teachers infected, family for total number of adult family members infected, hosp_child for total number of children hospitalized, and hosp_adult for the total number of adults hospitalized.
Hospitalization Rate (unvaccinated adult)	0.024000	0.023996	<1%	
Vaccine uptake (student)	0.250	0.250	<1%	The total number of individuals flagged as vaccinated was divided by the total number of individuals. *Tracker names (in abm_code.R):* vacc
Vaccine uptake (teacher)	0.700	0.700	<1%	
Vaccine uptake (family)	0.700	0.700	<1%	
Vaccine effectiveness	0.700	0.700	<1%	The total number of individuals flagged as "not susceptible" was divided by the total number of individuals flagged as vaccinated. *Tracker names (in abm_code.R):* susp for susceptible individuals and vacc for vaccinated individuals.



Local Incidence Rate (cases per residents per day)	0.000150	0.000150	<1%	The total number of individuals infected in the wider community was divided by the product of the total number of days run and the total number of individuals in the model. When an individual was infected within the model (e.g., at school, not in the wider community), the number of days remaining in the model run after their infection was subtracted from the denominator of this fraction. *Tracker names (in abm_code.R):* child.start.count and adult.start.count for total number of individuals infected in the wider community, and child.community.risk.days and adult.community risk days for the number of days individuals were at risk of infection from the wider community.
Local Incidence Rate (cases per residents per day)	0.000750	0.000749	<1%	
Local Incidence Rate (cases per residents per day)	0.001500	0.001497	<1%	

Structural Checks

Household Contact Structure

Model run stopped if the household members contacted by each infected individual do not match the list of all uninfected and susceptible individuals in that household (line 403 in abm_code.R).

Model run stopped if an infected individual does not contact anyone in their household, but there are uninfected individuals in that household (line 412 in abm_code.R).

Model run is stopped if an infected individual who was not infected in the wider community does not contact any household members (line 1286 in abm_code.R).

Model run is stopped if an individual infected in the wider community infects an individual in their household (line 1311 in abm_code.R).

In-School Transmissions

Model run is stopped if an infectious individual is at school on a weekend day (line 1326 in abm_code.R).

Model run is stopped if individuals infected at school were not supposed to be present at school on that day (lines 1345-1364 in abm_code.R). The list of individuals present in school on each given day is determined by a separate testing function that takes into account the quarantine, isolation, and testing policy structure in the model (lines 807-866 in abm_code.R).

Classroom Contact Structure

Model run stopped if the classroom members contacted by each infected individual do not match the uninfected and susceptible individuals in their classroom present in school on a given day (lines 441-457

Testing

Case #2 (af3g%^&*): Complex simulation models

Not every test is worth doing.

But be explicit about what met the bar

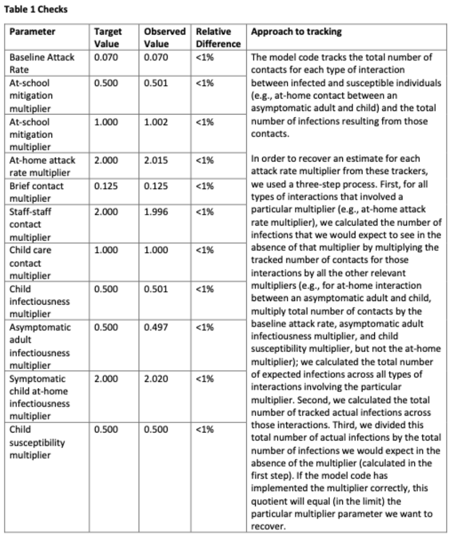
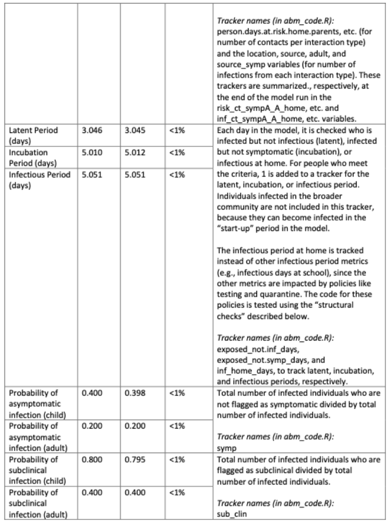
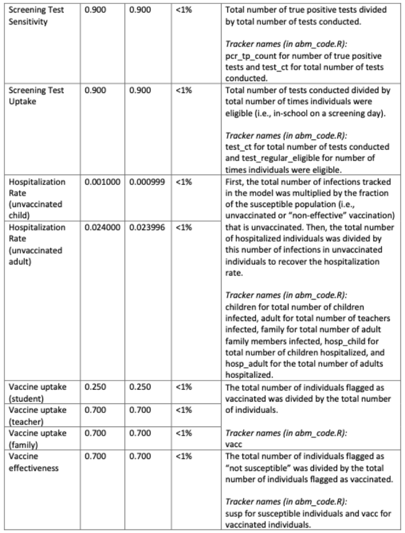


Table 1 Checks

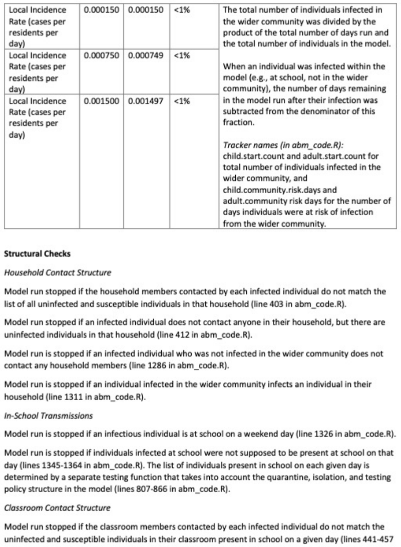
Parameter	Target Value	Observed Value	Relative Difference	Approach to tracking
Baseline Attack Rate	0.070	0.070	<1%	The model code tracks the total number of contacts for each type of interaction between infected and susceptible individuals (e.g., at-home contact between an asymptomatic adult and child) and the total number of infections resulting from those contacts. In order to recover an estimate for each attack rate multiplier from these trackers, we used a three-step process. First, for all types of interactions that involved a particular multiplier (e.g., at-home attack rate multiplier), we calculated the number of infections that we would expect to see in the absence of that multiplier by multiplying the tracked number of contacts for those interactions by all the other relevant multipliers (e.g., for at-home interaction between an asymptomatic adult and child, multiply total number of contacts by the baseline attack rate, asymptomatic adult infectiousness multiplier, and child susceptibility multiplier, but not the at-home multiplier); we calculated the total number of expected infections across all types of interactions involving the particular multiplier. Second, we calculated the total number of tracked actual infections across those interactions. Third, we divided this total number of actual infections by the total number of infections we would expect in the absence of the multiplier (calculated in the first step). If the model code has implemented the multiplier correctly, this quotient will equal (in the limit) the particular multiplier parameter we want to recover.
At-school mitigation multiplier	0.500	0.501	<1%	
At-school mitigation multiplier	1.000	1.002	<1%	
At-home attack rate multiplier	2.000	2.015	<1%	
Brief contact multiplier	0.125	0.125	<1%	
Staff-staff contact multiplier	2.000	1.996	<1%	
Child care contact multiplier	1.000	1.000	<1%	
Child infectiousness multiplier	0.500	0.501	<1%	
Asymptomatic adult infectiousness multiplier	0.500	0.497	<1%	
Symptomatic child at-home infectiousness multiplier	2.000	2.020	<1%	
Child susceptibility multiplier	0.500	0.500	<1%	



				Tracker names (in abm_code.R): person.days.at.risk.home.parents, etc. (for number of contacts per interaction type) and the location, source, adult, and source_symp variables (for number of infections from each interaction type). These trackers are summarized., respectively, at the end of the model run in the risk_ct_sympA_A_home, etc. and inf_ct_sympA_A_home, etc. variables.
Latent Period (days)	3.046	3.045	<1%	Each day in the model, it is checked who is infected but not infectious (latent), infected but not symptomatic (incubation), or infectious at home. For people who meet the criteria, 1 is added to a tracker for the latent, incubation, or infectious period. Individuals infected in the broader community are not included in this tracker, because they can become infected in the "start-up" period in the model. The infectious period at home is tracked instead of other infectious period metrics (e.g., infectious days at school), since the other metrics are impacted by policies like testing and quarantine. The code for these policies is tested using the "structural checks" described below. *Tracker names (in abm_code.R):* exposed_not.inf_days, exposed_not.symp_days, and inf_home_days, to track latent, incubation, and infectious periods, respectively.
Incubation Period (days)	5.010	5.012	<1%	
Infectious Period (days)	5.051	5.051	<1%	
Probability of asymptomatic infection (child)	0.400	0.398	<1%	Total number of infected individuals who are not flagged as symptomatic divided by total number of infected individuals. *Tracker names (in abm_code.R):* symp
Probability of asymptomatic infection (adult)	0.200	0.200	<1%	
Probability of subclinical infection (child)	0.800	0.795	<1%	Total number of infected individuals who are flagged as subclinical divided by total number of infected individuals. *Tracker names (in abm_code.R):* sub_clin
Probability of subclinical infection (adult)	0.400	0.400	<1%	



Screening Test Sensitivity	0.900	0.900	<1%	Total number of true positive tests divided by total number of tests conducted. *Tracker names (in abm_code.R):* pcr_tp_count for number of true positive tests and test_ct for total number of tests conducted.
Screening Test Uptake	0.900	0.900	<1%	Total number of tests conducted divided by total number of times individuals were eligible (i.e., in-school on a screening day). *Tracker names (in abm_code.R):* test_ct for total number of tests conducted and test_regular_eligible for number of times individuals were eligible.
Hospitalization Rate (unvaccinated child)	0.001000	0.000999	<1%	First, the total number of infections tracked in the model was multiplied by the fraction of the susceptible population (i.e., unvaccinated or "non-effective" vaccination) that is unvaccinated. Then, the total number of hospitalized individuals was divided by this number of infections in unvaccinated individuals to recover the hospitalization rate. *Tracker names (in abm_code.R):* children for total number of children infected, adult for total number of teachers infected, family for total number of adult family members infected, hosp_child for total number of children hospitalized, and hosp_adult for the total number of adults hospitalized.
Hospitalization Rate (unvaccinated adult)	0.024000	0.023996	<1%	
Vaccine uptake (student)	0.250	0.250	<1%	The total number of individuals flagged as vaccinated was divided by the total number of individuals. *Tracker names (in abm_code.R):* vacc
Vaccine uptake (teacher)	0.700	0.700	<1%	
Vaccine uptake (family)	0.700	0.700	<1%	
Vaccine effectiveness	0.700	0.700	<1%	The total number of individuals flagged as "not susceptible" was divided by the total number of individuals flagged as vaccinated. *Tracker names (in abm_code.R):* susp for susceptible individuals and vacc for vaccinated individuals.



Local Incidence Rate (cases per residents per day)	0.000150	0.000150	<1%	The total number of individuals infected in the wider community was divided by the product of the total number of days run and the total number of individuals in the model. When an individual was infected within the model (e.g., at school, not in the wider community), the number of days remaining in the model run after their infection was subtracted from the denominator of this fraction. *Tracker names (in abm_code.R):* child.start.count and adult.start.count for total number of individuals infected in the wider community, and child.community.risk.days and adult.community risk days for the number of days individuals were at risk of infection from the wider community.
Local Incidence Rate (cases per residents per day)	0.000750	0.000749	<1%	
Local Incidence Rate (cases per residents per day)	0.001500	0.001497	<1%	

Structural Checks

Household Contact Structure

Model run stopped if the household members contacted by each infected individual do not match the list of all uninfected and susceptible individuals in that household (line 403 in abm_code.R).

Model run stopped if an infected individual does not contact anyone in their household, but there are uninfected individuals in that household (line 412 in abm_code.R).

Model run is stopped if an infected individual who was not infected in the wider community does not contact any household members (line 1286 in abm_code.R).

Model run is stopped if an individual infected in the wider community infects an individual in their household (line 1311 in abm_code.R).

In-School Transmissions

Model run is stopped if an infectious individual is at school on a weekend day (line 1326 in abm_code.R).

Model run is stopped if individuals infected at school were not supposed to be present at school on that day (lines 1345-1364 in abm_code.R). The list of individuals present in school on each given day is determined by a separate testing function that takes into account the quarantine, isolation, and testing policy structure in the model (lines 807-866 in abm_code.R).

Classroom Contact Structure

Model run stopped if the classroom members contacted by each infected individual do not match the uninfected and susceptible individuals in their classroom present in school on a given day (lines 441-457

Collaboration

You'd be hard-pressed to find a business that has coders go solo.

Coding solo is more error-prone than collaborative coding, even with testing.

Code review and/or double-coding is often more efficient.

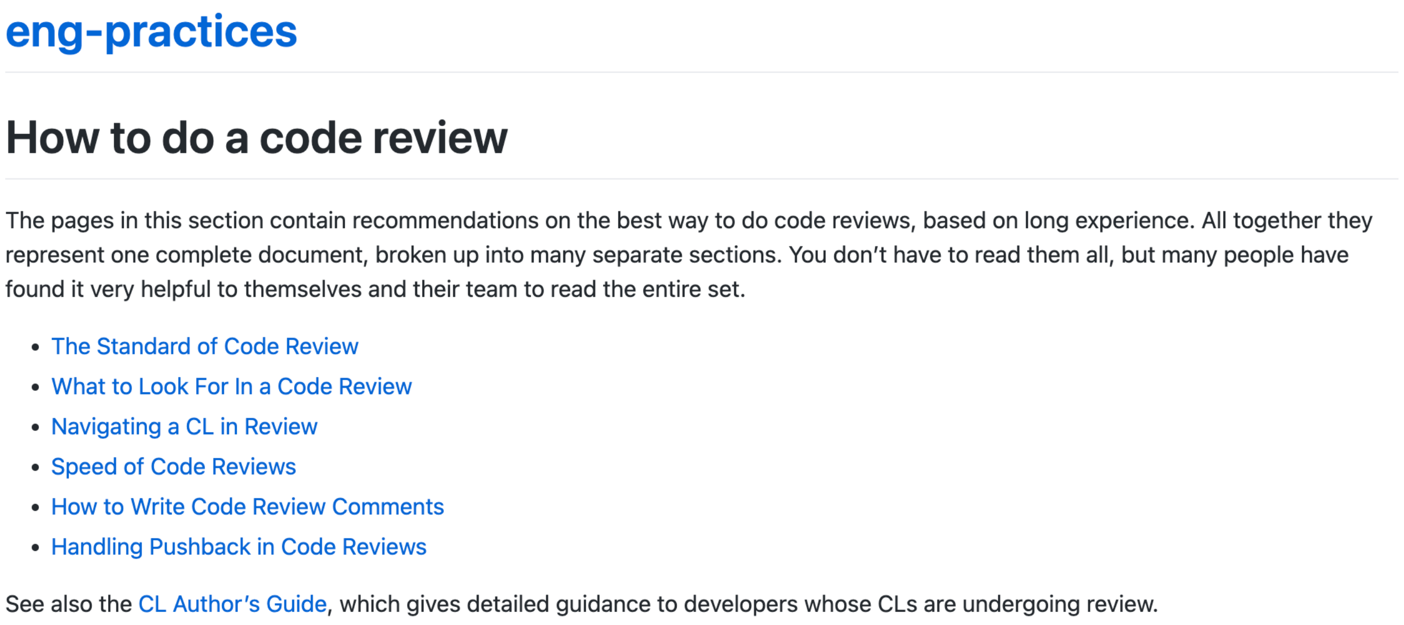
Collaboration

1 Code review

HOW TO MAKE A GOOD CODE REVIEW
geek & poke
AT LEAST WE DON'T NEED TO OBFUSCATE IT BEFORE SHIPPING
RULE 1: TRY TO FIND AT LEAST SOMETHING POSITIVE
KEEP IN MIND THAT I'M SELF-TAUGHT, SO MY CODE MAY BE A LITTLE MESSY.
LEMME SEE– I'M SURE IT'S FINE.
...WOW.
THIS IS LIKE BEING IN A HOUSE BUILT BY A CHILD USING NOTHING BUT A HATCHET AND A PICTURE OF A HOUSE.
IT'S LIKE A SALAD RECIPE WRITTEN BY A CORPORATE LAWYER USING A PHONE AUTOCORRECT THAT ONLY KNEW EXCEL FORMULAS.
IT'S LIKE SOMEONE TOOK A TRANSCRIPT OF A COUPLE ARGUING AT IKEA AND MADE RANDOM EDITS UNTIL IT COMPILED WITHOUT ERRORS.
OKAY, I'LL READ A STYLE GUIDE.

Collaboration

1 Code review



eng-practices

How to do a code review

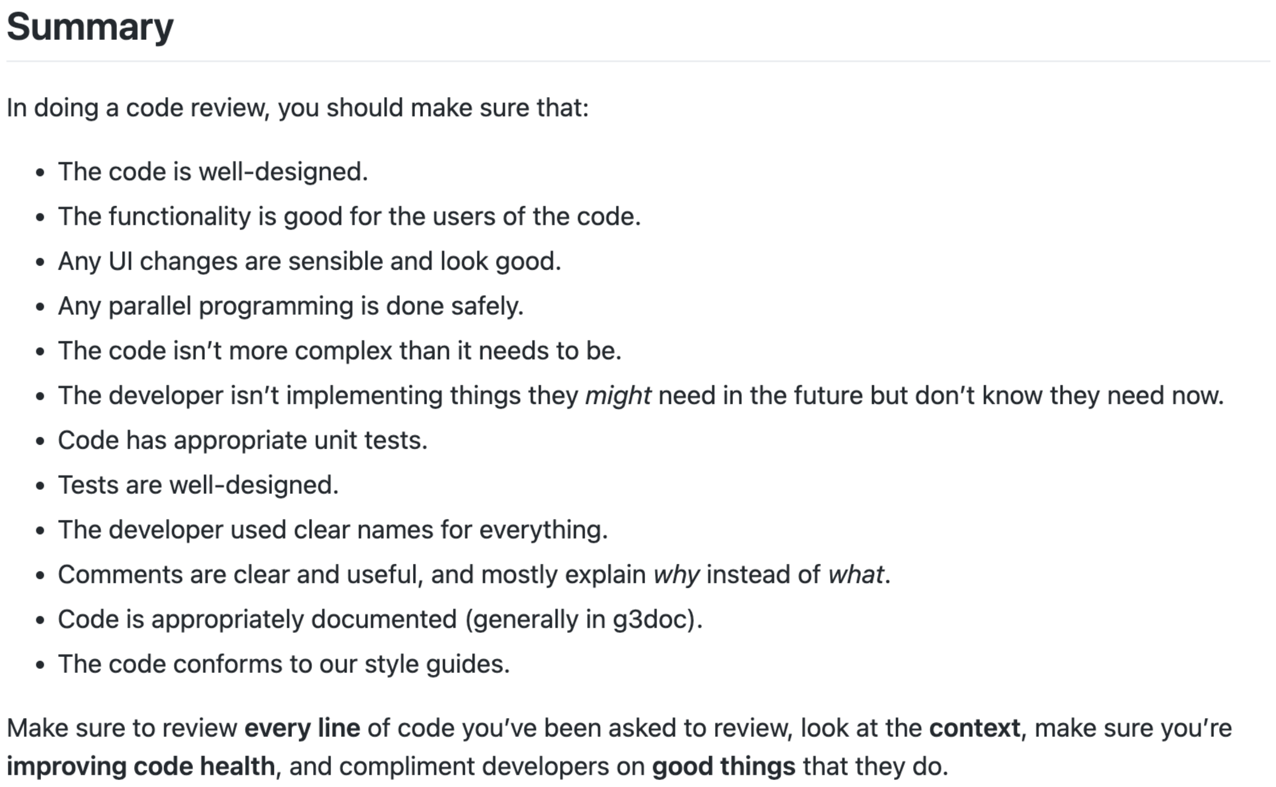
The pages in this section contain recommendations on the best way to do code reviews, based on long experience. All together they represent one complete document, broken up into many separate sections. You don't have to read them all, but many people have found it very helpful to themselves and their team to read the entire set.

- The Standard of Code Review
- What to Look For In a Code Review
- Navigating a CL in Review
- Speed of Code Reviews
- How to Write Code Review Comments
- Handling Pushback in Code Reviews

See also the CL Author's Guide, which gives detailed guidance to developers whose CLs are undergoing review.

Collaboration

1 Code review



Summary

In doing a code review, you should make sure that:

- The code is well-designed.
- The functionality is good for the users of the code.
- Any UI changes are sensible and look good.
- Any parallel programming is done safely.
- The code isn’t more complex than it needs to be.
- The developer isn’t implementing things they *might* need in the future but don’t know they need now.
- Code has appropriate unit tests.
- Tests are well-designed.
- The developer used clear names for everything.
- Comments are clear and useful, and mostly explain *why* instead of *what*.
- Code is appropriately documented (generally in g3doc).
- The code conforms to our style guides.

Make sure to review **every line** of code you’ve been asked to review, look at the **context**, make sure you’re **improving code health**, and compliment developers on **good things** that they do.

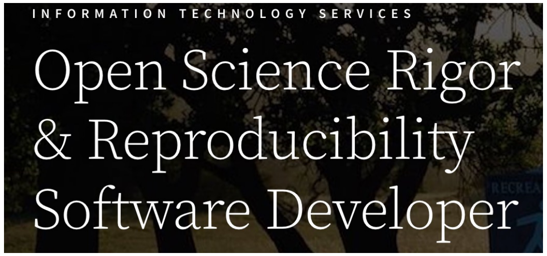
Collaboration

Let's be creative.

2 Double code when feasible.

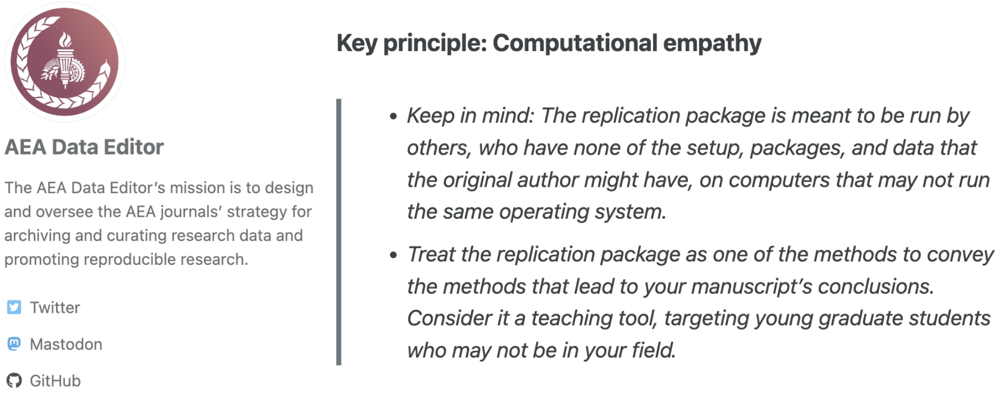
Even self-coding…

3 Formal structures for independent reproducibility.



INFORMATION TECHNOLOGY SERVICES

Open Science Rigor & Reproducibility Software Developer



AEA Data Editor

The AEA Data Editor's mission is to design and oversee the AEA journals' strategy for archiving and curating research data and promoting reproducible research.

Twitter

Mastodon

GitHub

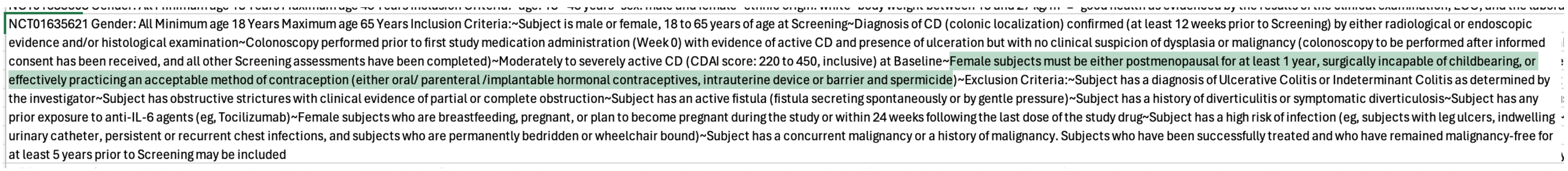
Key principle: Computational empathy

- *Keep in mind: The replication package is meant to be run by others, who have none of the setup, packages, and data that the original author might have, on computers that may not run the same operating system.*
- *Treat the replication package as one of the methods to convey the methods that lead to your manuscript's conclusions. Consider it a teaching tool, targeting young graduate students who may not be in your field.*

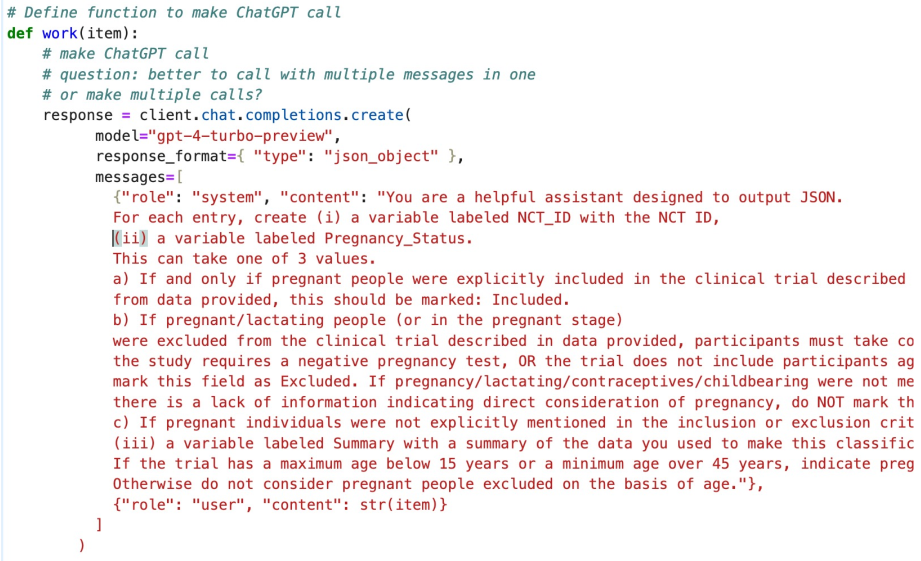
Collaboration

Let’s be creative.

4 AI solutions



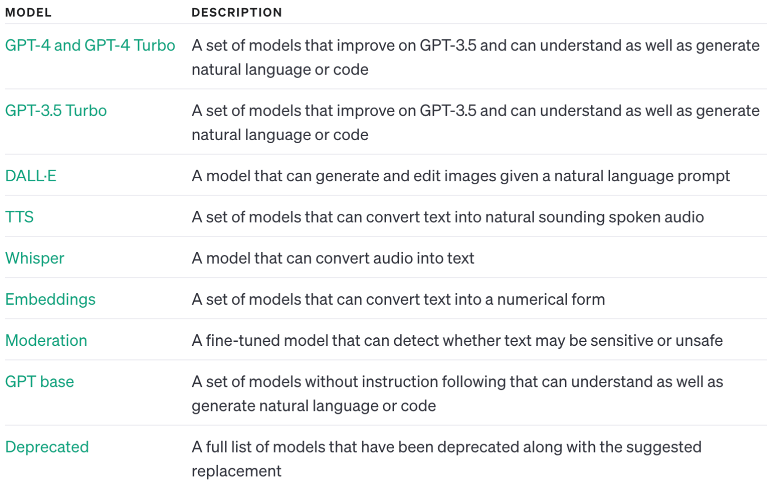
NCT01635621 Gender: All Minimum age 18 Years Maximum age 65 Years Inclusion Criteria:~Subject is male or female, 18 to 65 years of age at Screening~Diagnosis of CD (colonic localization) confirmed (at least 12 weeks prior to Screening) by either radiological or endoscopic evidence and/or histological examination~Colonoscopy performed prior to first study medication administration (Week 0) with evidence of active CD and presence of ulceration but with no clinical suspicion of dysplasia or malignancy (colonoscopy to be performed after informed consent has been received, and all other Screening assessments have been completed)~Moderately to severely active CD (CDAI score: 220 to 450, inclusive) at Baseline~Female subjects must be either postmenopausal for at least 1 year, surgically incapable of childbearing, or effectively practicing an acceptable method of contraception (either oral/ parenteral /implantable hormonal contraceptives, intrauterine device or barrier and spermicide)~Exclusion Criteria:~Subject has a diagnosis of Ulcerative Colitis or Indeterminant Colitis as determined by the investigator~Subject has obstructive strictures with clinical evidence of partial or complete obstruction~Subject has an active fistula (fistula secreting spontaneously or by gentle pressure)~Subject has a history of diverticulitis or symptomatic diverticulosis~Subject has any prior exposure to anti-IL-6 agents (eg, Tocilizumab)~Female subjects who are breastfeeding, pregnant, or plan to become pregnant during the study or within 24 weeks following the last dose of the study drug~Subject has a high risk of infection (eg, subjects with leg ulcers, indwelling urinary catheter, persistent or recurrent chest infections, and subjects who are permanently bedridden or wheelchair bound)~Subject has a concurrent malignancy or a history of malignancy. Subjects who have been successfully treated and who have remained malignancy-free for at least 5 years prior to Screening may be included



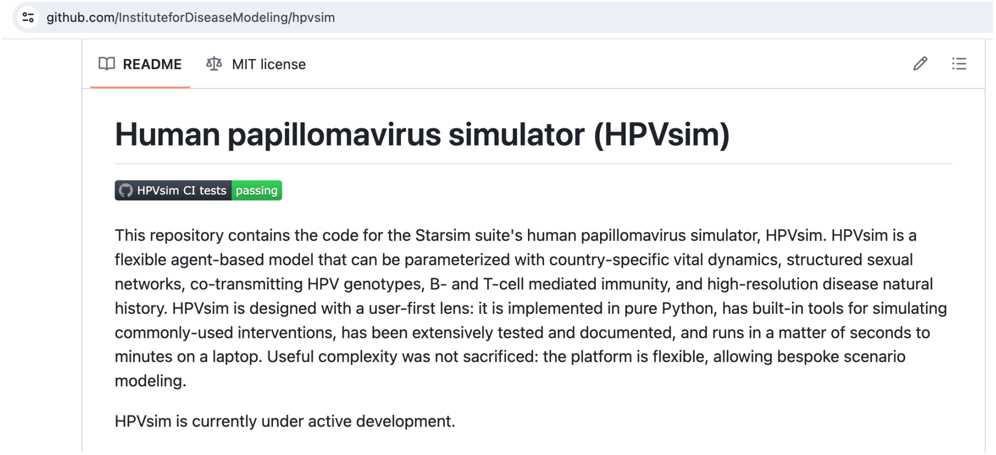
```
# Define function to make ChatGPT call
def work(item):
    # make ChatGPT call
    # question: better to call with multiple messages in one
    # or make multiple calls?
    response = client.chat.completions.create(
          model="gpt-4-turbo-preview",
          response_format={ "type": "json_object" },
          messages=[
            {"role": "system", "content": "You are a helpful assistant designed to output JSON.
            For each entry, create (i) a variable labeled NCT_ID with the NCT ID,
            (ii) a variable labeled Pregnancy_Status.
            This can take one of 3 values.
            a) If and only if pregnant people were explicitly included in the clinical trial described
            from data provided, this should be marked: Included.
            b) If pregnant/lactating people (or in the pregnant stage)
            were excluded from the clinical trial described in data provided, participants must take co
            the study requires a negative pregnancy test, OR the trial does not include participants ag
            mark this field as Excluded. If pregnancy/lactating/contraceptives/childbearing were not me
            there is a lack of information indicating direct consideration of pregnancy, do NOT mark th
            c) If pregnant individuals were not explicitly mentioned in the inclusion or exclusion crit
            (iii) a variable labeled Summary with a summary of the data you used to make this classific
            If the trial has a maximum age below 15 years or a minimum age over 45 years, indicate preg
            Otherwise do not consider pregnant people excluded on the basis of age."},
            {"role": "user", "content": str(item)}
          ]
        )
```

Iteration

Final_Version_2_FINAL



MODEL	DESCRIPTION
GPT-4 and GPT-4 Turbo	A set of models that improve on GPT-3.5 and can understand as well as generate natural language or code
GPT-3.5 Turbo	A set of models that improve on GPT-3.5 and can understand as well as generate natural language or code
DALL·E	A model that can generate and edit images given a natural language prompt
TTS	A set of models that can convert text into natural sounding spoken audio
Whisper	A model that can convert audio into text
Embeddings	A set of models that can convert text into a numerical form
Moderation	A fine-tuned model that can detect whether text may be sensitive or unsafe
GPT base	A set of models without instruction following that can understand as well as generate natural language or code
Deprecated	A full list of models that have been deprecated along with the suggested replacement



Human papillomavirus simulator (HPVsim)

This repository contains the code for the Starsim suite's human papillomavirus simulator, HPVsim. HPVsim is a flexible agent-based model that can be parameterized with country-specific vital dynamics, structured sexual networks, co-transmitting HPV genotypes, B- and T-cell mediated immunity, and high-resolution disease natural history. HPVsim is designed with a user-first lens: it is implemented in pure Python, has built-in tools for simulating commonly-used interventions, has been extensively tested and documented, and runs in a matter of seconds to minutes on a laptop. Useful complexity was not sacrificed: the platform is flexible, allowing bespoke scenario modeling.

HPVsim is currently under active development.

Opportunities and challenges for health policy

Software engineers have both:

More intensive procedures
Much lower standards

This invites both humility and effort.

Opportunities and challenges for health policy

Develop **framework** and **software** to support reproducibility in health policy.

1 **Proprietary data**

Code release

Replication package on public/aggregated data

2 **Limited seats**

3 **Ethical concerns**

Opportunities and challenges for health policy

Develop **framework** and **software** to support reproducibility in health policy.

How you thought about your code should be as clear as how you thought about your statistical **methods**

Questions?

We would love any of your thoughts on these ideas.

alyssa_bilinski@brown.edu