Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *doi: 10.18637/jss.v000.i00*

Time invariant analysis of epidemics with EpiCompare

Shannon K. Gallagher
Biostatistics Research Branch
National Institute of Allergy
and Infectious Diseases

Benjamin LeRoy
Dept. of Statistics & Data Science
Carnegie Mellon University

Abstract

We present **EpiCompare**, an R package that suppliments and enhances current infectious disease analysis pipelines and encourages comparisons across models and epidemics. A major contribution of this work is the set of novel *time-invariate* tools for model and epidemic comparisons - including time-invariate prediction bands. **EpiCompare** embraces R's *tidy* coding style to make adoption of the package easier and analysis faster. This paper provides an overview of both the tools in and intuition behind **EpiCompare** and a thorough demonstrating of the tools through a detailed example of a full data analysis pipeline.

Keywords: keywords, not capitalized, Java.

1. Introduction

The recent (and on-going) COVID-19 global pandemic has galvanized public interest in understanding more about infectious disease modeling and has highlighted the usefulness of research in the area of infectious disease epidemiology. Infectious diseases inflict enormous burdens on the world: millions of lives lost and trillions of dollars spent yearly. Infectious disease models typically attempt to do one or more of the following: 1) predict the spread of current and future epidemics (e.g. flu prediction Biggerstaff *et al.* 2016), 2) analyze past and current epidemics to increase scientific knowledge (e.g. historical measle outbreaks Neal and Roberts 2004), and 3) forecast or project epidemic scenarios under pre-specified parameters (e.g. Ferguson *et al.* 2020). At the same time, descriptive statistics and visualizations from universities, many branches and levels of government, and news organizations are an important first step of the process as has been seen in the current COVID-19 epidemic(Dong *et al.*

2020; CDC 2021; The Washington Post 2021). [1]

With the many visualization and exploratory tools, models and modeling paradigms, and reviews and comparisons in the literature and through the MIDAS (Models of Infectious Disease Agent Study) network (MIDAS Network 2021), this field has a lot of devices to aid an individual practitioner decide the correct approach. For example, R packages such as **surveillance**, **EpiModel**, and **pomp** have all made significant steps in standardizing the flow of the data analysis pipeline for epidemic modeling through digitizing data sets, making accessible statistical models, and providing a plethora of educational material for both coding novices and experts alike (Meyer *et al.* 2017; Jenness *et al.* 2018; King *et al.* 2016).

At the same time, analysis packages often only address a specific portion of the analysis pipeline~~, for instance focusing on certain types of models.~~ These modeling tools~~, which~~ usually require learning package-specific syntax~~,~~ and often don't provide easy ways to compare and assess their models on new data. Moreover, exploring, ~~and~~ modeling and comparing epidemics require transforming and *tidying* data in different ways. To fill these gaps, we present our R package **EpiCompare**. Our package's primary focus is to aid and advance research in the area of comparison and assessment of epidemic and epidemiological models. In Figure 1, we illustrate the data analysis pipeline of infectious diseases as 1) data pre-processing, 2) exploratory data analysis (EDA), 3) modeling and simulating, 4) post-processing, and 5) comparison and assessment; where each previous part of the pipeline influences the next. **EpiCompare** provides tools to aids practitioners in all areas of this pipeline.

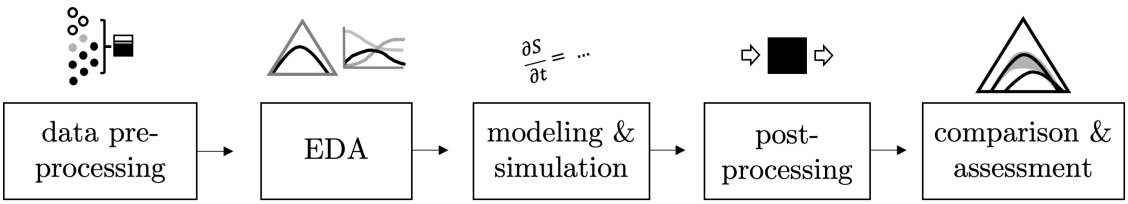

Figure 1: An idealized epidemiological data analysis pipeline.

EpiCompare also emphasizes the value of analyzing epidemics in a *time-invariant* way. Epidemics, despite by definition being a process that evolves over time, often need to be compared in a way not constrained to initial times or time scales to understand the processes at play. Time-invariant analysis can also make it easier to compare state-space models in a more global, holistic fashion. ~~Moreover, m~~ Many current time-dependent comparison tools for state-space models (e.g. SIR models) ~~highlight~~ examine the proportion of individuals in each state (at a given time) in a piece-wise / marginal fashion. These ~~This~~ approaches may reduce the amount of connections that can be seen, similar to projections of a multidimensional distribution onto a single axis at a time. Tools in **EpiCompare** give the user the ability to extend their toolkit to evaluate epidemics within a time-invariant lens. The goal of **EpiCompare** is not to supplant existing infectious disease modeling tools and software but, rather, is a concerted effort

[1] [Ben says: probably should have a conclusion sentence here - seems to end abruptly. *This is less so the case now.]

to create standard and fair comparisons among models developed for disease outbreaks and outbreak data.

This paper is broken up into the following sections; section 2 motivates and showcases tools of time-invariant analysis, section 3 presents an outline of how **EpiCompare** aids a practitioner in every step of the pipeline and section 4 provides a ~~thorough~~ demonstrating of the tools through a detailed example of a full data analysis pipeline.

2. Motivation and tools for time-invariant analysis

EpiCompare delivers *time-invariant* analysis by (1) taking a global, not marginal view of how epidemics move through populations and (2) by treating full epidemics as filamental trajectories. The following section aims to highlight the strengths of *time-invariant* analysis and define the mathematical foundations that **EpiCompare**'s tools stand upon.

Mathematically, epidemics are complex objects. They can be hard to assess and compare to one another due to the differences in the diseases, the location where the outbreak occurs, how the affected population reacts, and the time ~~related~~related features (including start of the epidemic, speed of infection and more). Time-invariant analysis makes different epidemics easier to compare by removing many time dependent aspects of an epidemic. ~~Instead,~~ Time-invariant analysis focuses on the global pattern of an epidemic, via filamental trajectories, and emphasizes the number of lives affected. [Ben wants to try this sentence again.]

2.1. Motivating time-invariant analysis through the reproduction number R_0

Time-invariant analysis, as it appears in **EpiCompare**, bypasses many difficulties in comparing different epidemics. With time-invariant analysis, comparing the decades-long outbreak of HIV in the US to a 10 day outbreak of norovirus on a cruise ship is ~~still~~ possible. Time-dependent problems can arise when estimating epidemiological parameters, including the reproduction number R_0. ~~We will use R_0 to motivate the usefulness of time-invariant analysis in this section.~~[2]

R_0 is probably the most famous ~~time-invariant~~ numerical summary of an epidemic and is often associated with the Susceptible-Infectious-Recovered (SIR) model (Hethcote 2000). R_0 is ~~a one-number summary of a disease and is~~ defined as the expected number of infections caused by a single infector who is added to a completely susceptible population (Anderson and May 1992). This definition has no mention of time and hence means that R_0 is a time-invariant parameter. Yet R_0 is estimated with time-~~based~~dependent data, which can make it a difficult quantity to estimate. For example, Gallagher *et al.* (2020) demonstrate how R_0 can be sensitive to time-~~based~~dependent parameters such as the beginning and end of an epidemic, two quantities that generally arehard to define precisely.~~do not have precise definitions~~. To demonstrate the difficulty of discerning R_0 in ~~a~~other[3] time-dependent analysis, we first introduce Kermack and McKendrick (1927)'s SIR model. This model captures the transitions from one state to the next as a system of ordinary differential equations, where N is the total number of individuals, β is the rate of infection, and γ is the rate of recovery,

[2] I don't think this is a necessary sentence. I still think it adds value to the story and I'm not sure people really read section titles that are long.

[3] I change this so we don't confused readers that we're going show the impact in tools beyond just the estimation itself.

$$
\begin{aligned}
S'(t) &= -\frac{\beta S(t) I(t)}{N} \\
I'(t) &= \frac{\beta S(t) I(t)}{N} - \gamma I(t) \\
R'(t) &= \gamma I(t).
\end{aligned} \tag{1}
$$

From this model,the reproduction number is the ratio of the infection rate to the recovery rate, $R_0 = \beta/\gamma$, ~~aka the ratio of the infection rate compared to the recovery rate.~~ From this definition, given ~~Since~~ β and γ are both rates, it should be clear that the ratio of the two, R_0, is a time-invariant quantity.[4] ~~Once R_0 is estimated, practitioners can infer important epidemic quantities such as the total number of infections or the percent of a population needed to be vacccinated to stop the sustained spread of an epidemic. Moreover, R_0 allows us to compare different diseases and different instances of outbreaks on the same scale.~~[5]

[Ben says: It's unclear to me why we have a subtitle here - isn't it just more motivation of time-invariant anlysis with R_0? Also, I feel like the story is weak here. The point is to leverage R_0 to show the value of time-invariant analysis - this seems a bit more like just discussing properties of R_0. In the follow rewrite I use "[" and "]" to indicate that this is a section from your earlier draft.] [Shannon says: Tried to tie this better to the previous part since it's no longer a new section. also highlighted tie to time-invariant analysis and R_0. I also wanted to bring the punch line (overlapping epidemics = same r0) closer to the beginning so those who don't want to slog through mathematical details can get the takeaway.]Shannon tries again in blue

~~[Ben says: this paragraph needs to still be looked at. Also I'm not sure why this particular paragraph was c~~
~~Time-invariant analysis helps practitioners to more easily compare R_0 from different outbreaks.~~
~~For example, consider two epidemics generated from the Kermack and McKendrick SIR equations. The first epidemic has parameters $\beta_1, \gamma_1 = (0.8, 0.4)$ and the second has $\beta_2, \gamma_2 = (0.64, 0.32)$. Both epidemics have populations of 1000 people with 10 individuals initially infected. Additionally note that the two reproduction numbers are the same for each epidemic, $R_0 = 2 = 0.8/0.4 = 0.64/0.32$. We plot the epidemics with traditional *state* vs. *time* plots[6]. In Fig. 2 we show the time-based paths for the S, I, and R states for the first 15 days of observed data. In this time-variant view, we may believe that epidemic 1 has a larger R_0 than epidemic 2 because the peak of infection occurs more quickly than in Epidemic 2. On the other hand, we may believe epidemic 2 has a larger R_0 because its unclear if the number of infections in that epidemic has not yet peaked at time 15. In this time-variant view, we cannot determine if one epidemic has larger value of R_0[7].~~

Since R_0 is an important value, it would be helpful to have more intuitive ways of comparing one R_0 to another. Usually numerical summaries of R_0 are presented, which while overall very helpful, may be confusing when presented along side epidemic data that are visualized in a traditional, time-dependent manner.

For example, consider two epidemics generated from the Kermack and McKendrick SIR equations where both have the same value of R_0. The first epidemic has parameters $\beta_1, \gamma_1 =$

[4] I am trying to make it look like we are not repeating ourselves by saying R_0 is time-invariant.
[5] cool facts about r0, but not the central point
[6] This sentecne is out of place / doesn't connect with the other sentences.
[7] This sentnce doesn't connect wit previous examples.

$(0.8, 0.4)$ and the second has $\beta_2, \gamma_2 = (0.64, 0.32)$. Both epidemics have populations of 1000 people with 10 individuals initially infected. An analysis may present an estimate of $\hat{R}_0 = 2$ alongside state vs. time plots like those shown in Figure 2. The paths of the epidemics in the state vs. time view seem to differ from one another including having different infection peaks. From these traditional time-based plots, there is no intuitive way to conclude that these two epidemics have the same value of R_0.

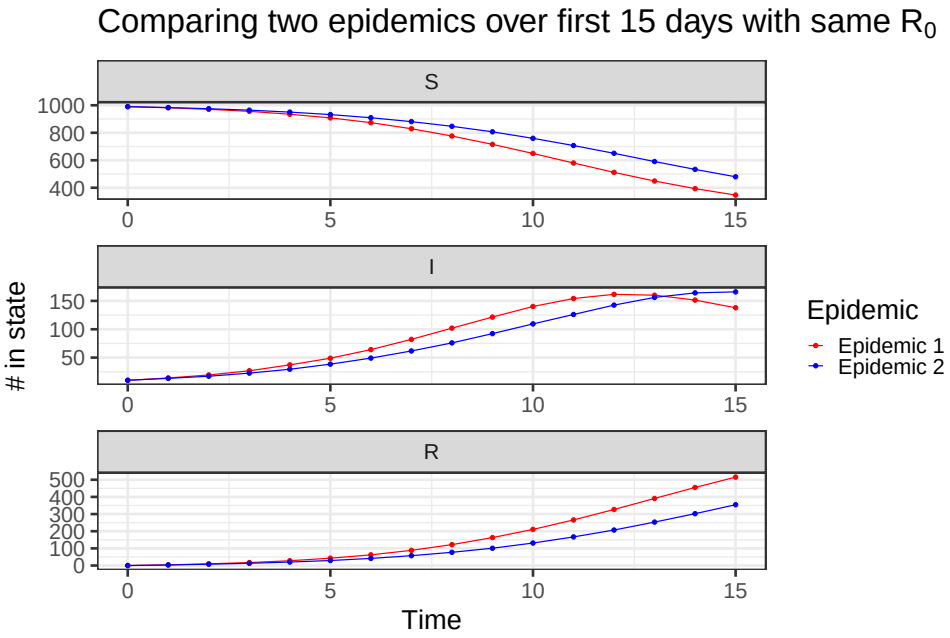

Figure 2: Example of two epidemics with different β and γ paremeters but the same initial reproduction number $R_0 = 2$. Both epidemics are generated from models with $N = 1000$ individuals with $S(0) = 990$ and $I(0) = 10$.

EpiCompare provides a time-invariant tool to visualize these epidemics in a more intuitive manner, at least in regards to comparing values of R_0. ~~A time-invariant approach to visualizing epidemics, in comparison, allows us to directly compare R_0 from a single plot.~~ For every time ~~point~~ t we have a point $(S(t), I(t), R(t))$~~, so we can treat epidemics as a trajectory in this three-dimensional space, as we visual in the left subplot of Figure ??.~~ so we can visualize the trajectory of the epidemic in three-dimensional space (see Fig. 3 (left)). For state space models like in our example, given the constraint that $S(t) + I(t) + R(t)$ is always equal to N (the total population size), we can visual these point in a a two-dimesional *ternary* plot, as seen in Figure 3 (right). ~~In Fig. 3 we plot the filamental trajectories of the two epidemics in a time-invariant view. We will explain how and why this works shortly. The important takeaway is that in this time-invariant view, i~~It is apparent that these epidemics are on "the same path." In this case, this indicates that two epidemics have the same value of R_0.

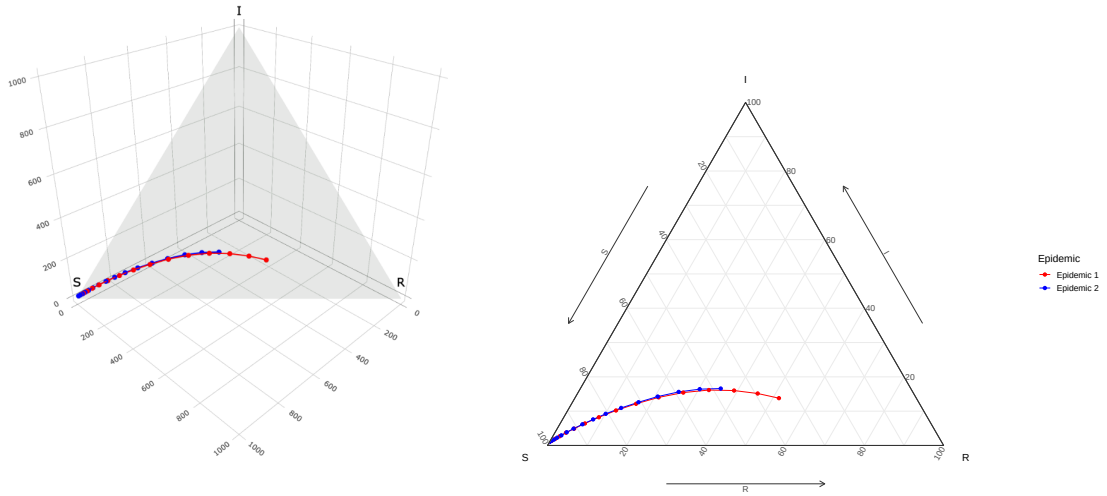

Figure 3: Left: trajectory of epidemic in three-dimensional space, plotting $(S(t), I(t), R(t))$. Right: the gray-shaded region and epidemic trajectory shown from (left) now shown in two-dimensional space. This is more commonly known as a ternary plot.

Underlying our time-invariant visualization that allowed for the comparison of R_0 in Fig. 3 is the treatment of the epidemic as a single filamental trajectory in the state space. ~~The reason why we can visually compare R_0 in Fig. 3 is because of the time-invariant nature of the filamental trajectory associated with an epidemic.~~ A filamental trajectory can be mathematically viewed as a set of points in space that have an ordering, and that all points on the line between these ordered points are also contained in the geometric object. For a SIR epidemic, we can represent the associated filamental trajectory ψ as

$$\psi = \{(S(t), I(t), R(t)) : S, I, R \geq 0, S + I + R = N\}_{t\in[0,T]}\,,$$

where a mapping $\xi : s \to \mathbb{R}$ that is strictly monotonically increasing would not change the definition of ψ, i.e. $\psi_\xi \equiv \psi$ where :

$$\psi_\xi = \{(S(\xi(s)), I(\xi(s)), R(\xi(s))) : S, I, R \geq 0, S + I + R = N\}_{s\in[0,T]}\,.$$

[Ben says: removed this paragraph now] Since the number of individuals in each state is non-negative and the sum over the three states for a given time point ~~sums to~~is N, then all points in ψ will lay in a \ben{two-dimensional triangular plane in three-dimensional space. ~~We can then which can be visualize the full filamental trajectory in a two-dimensional ternary plot~~ As a result, we can visualize the full filamental trajectory of an epidemic in a single ternary plot ~~2d plot and ultimately R_0~~. [Shannon says: Show pic here?]][8]

~~[This section could be a bit less wordy. But generally good.] We visualize the two epidemics in a global, ternary view in Figure 3. Without getting into too much detail of the intricacies of this plot, we immediately see the points of the two filaments ψ seem to form the same~~

[8]Which? 3d space one? But I'm leaning against it now. 3d never looks good in a paper.

~~trajectory. Now, it is much clearer that Model epidemic 2 is following the same trajectory as Model epidemic 1 but is not as far along in the infection process.~~

~~As suggested a few paragraphs back, t~~The filamental trajectories in Fig. 3 ~~seem to overlap~~, and we may suspect that something is fundamentally linking these two different epidemics together. Mathematically, we can show this fundamental link ~~turns~~ is R_0. Let our two epidemics be presented as $\{(S_1(t), I_1(t), R_1(t))\}_{t\geq 0}$, $\{(S_2(s), I_2(s), R_2(s))\}_{s\geq 0}$ respectively. As with the example, assume both models have the same initial values $(S(0), I(0), R(0))$, and let $R_0 = \frac{\beta_1}{\gamma_1} = \frac{\beta_2}{\gamma_2}$ where β_i and γ_i are the average infection rate and recovery rate, respectively, for SIR model $i = 1, 2$. And define $a > 0$ to be the relative scalar such that $\beta_2 = a\beta_1$ if and only if $\gamma_2 = a\gamma_1$.

Theorem 1. *Let there be two SIR models as described above. Then for all $t > 0$ there exists an $s > 0$ such that $(S_1(t), I_1(t), R_1(t)) = (S_2(s), I_2(s), R_2(s))$. Moreover, $s = \frac{1}{a}t$.*

The proof of Theorem 1 relies on a fairly recent result from Harko *et al.* (2014) and is shown in detail in Proof 4.7. The consequence of Theorem 1 is that for two SIR models that have the same initial percent of individuals in each state and R_0 then for every point on the epidemic path of the first SIR model ~~is also~~ can be mapped to a point on the epidemic path of the second SIR model. In other words, the two epidemics form the same filamental trajectory. For SIR models with similar initial state percentages, time-invariant analysis allows practitioners to compare values of R_0 at a glance.

2.2. Time-invariant analysis beyond R_0 and ~~Kermack's and McKendrick~~ SIR Models[9]

Through the R_0 example, we see that treating epidemics like filamental trajectories embedded in a lower dimensional space allows us to ~~better~~more fully compare the overall structure of the epidemic and see how the population is directly impacted. Time-invariant tools ~~that~~ can be useful even when the underlying generative model for the epidemic is unknown or have more than three epidemic states.

New paragraph Viewing epidemics as filamental trajectories provides ~~a lot~~ new ways to compare and examine epidemics in a time-invariant manner. \ben{For completed?epidemics that have ended, one way to examine their filamental trajectories is to ~~redefine~~represent the filamental trajectory as a finite sequence of equally spaced points. ~~finite sequence of points on the filamental trajectory that are equally spaced (i.e. equa-distance between pairs of ordered points). For epidemics that have "played" themselves out, one way to represent their filamental trajectories to avoid confusion stemming from~~impacts of temporal structure is ~~to define them as a sequence of points their trajectory with equi-distance between each point [Shannon says: are we missing a few words in this definition?]~~. This representation induces a natural distance between this type of representation between epidemics, specifically:

$$d_{\text{equi-distance}}(\psi_1, \psi_2) = \int_{s\in[0,1]} (\psi_1'(s) - \psi_2'(s))^2 ds$$

where $\psi_i'(s)$ the point along ψ_1 that is s fraction of $|\psi_1|$ distance away from the start of ψ_i.[10]

[9] Probably will need to change this title...

[10] I think you're trying to say something about a distance based on the equally space points. Some clarifying questions: is ψ' the derivative? Does proportion make more sense than fraction? or simply $\frac{|\psi|}{s}$? It's only naturally time-invariant if we have a well defined ending point, right?

This distance is naturally time-invariant, and can be plugged into multiple distance-based assessment tools to examine the overall "extremeness'' of points, including pseudo-density estimators and depth/local depth functions (for examples see Ciollaro *et al.* 2016; Geenens and Nieto-Reyes 2017). These extremeness estimators can be ~~very~~ useful when comparing ~~between a setof simulation~~ a set of simulated epidemics and the true epidemic. Moreover these extremness estimators, ~~and does~~ not constrain the number of states of the model~~, though we recommend projecting the points into the unit simplex~~[11] ~~(by making all values the proportion of the population in the given state)~~.

New paragraph: ~~If the set of epidemics that one is examining have only gone through a single cycle of the outbreak~~ If ~~one~~ a practicioner is interested in understanding an epidemics through a single ~~cycle~~realization of ~~their~~its outbreak (before the population of individuals have become susceptible again), then additional time-invariant tools, including prediction regions can be leveraged awk sentence[12]. In these settings, **EpiCompare** goes ~~we go~~ a step further and treats epidemics more like geometric filaments (i.e. filamental trajectories without an ordering of points) than filamental trajectories. In **EpiCompare**, we create prediction regions that contain a the top $(1-\alpha)$ proportion of simulated curves by defining geometric regions defined by the union of small geometric? filaments around the subset of simulations (~~subset~~ grouped by measures like the above pseudo-density estimates or depth estimates). These regions ~~look at~~show where in the state-space we expect the epidemic to traverse~~, and~~. Additionally, we can compare prediction regions defined by different models using many set difference distances ~~the Haussdorf why Haussdorf specifically? distance~~ as well as examining how well the truth epidemic matches the simulations by examining if the epidemic's trajectory lies within the prediction region. All these mentioned? geometric structures and distance notations apply to epidemics with any number of states, and at the end of Section 3 we ~~also~~ highlight how these prediction regions can aid in visual comparisons for epidemics with 3 states (like the SIR models).

3. Overview of EpiCompare

In this section, we present the tools implemented in **EpiCompare** and explain how they aid in the data analysis pipeline. In Figure 4, we show how our package's functions fit into the data analysis pipeline introduced in Figure 1. All front-facing functions in **EpiCompare** are aimed to be as user-friendly as possible. We also focus on providing the user "tidyverse" style functions, that encourage piping objects from one function to the next and follow clear "verb" naming schemes (Wickham *et al.* 2019). Although users can incorporate **EpiCompare** into any step in the data analysis pipeline, there are two primary points of entry. The first point of entry is the very beginning with pre-processing and visualizing raw data, and the second point of entry is after modeling and simulation. Figure 4 captures these different paths, and we highlight how to leverage **EpiCompare** functionalities in the subsections below.

Data pre-processing

The first step of most data analysis is "cleaning" the raw data so it can be explored. Before data can be explored, they must be collected. Sometimes individual records are collected,

[11] I'm not sure we've talked about this before... I don't think we have but am wondering if we're getting in the weeds

[12] {What are we predicting if the epidemic is done? Update 4/6: I'm satisfied.}

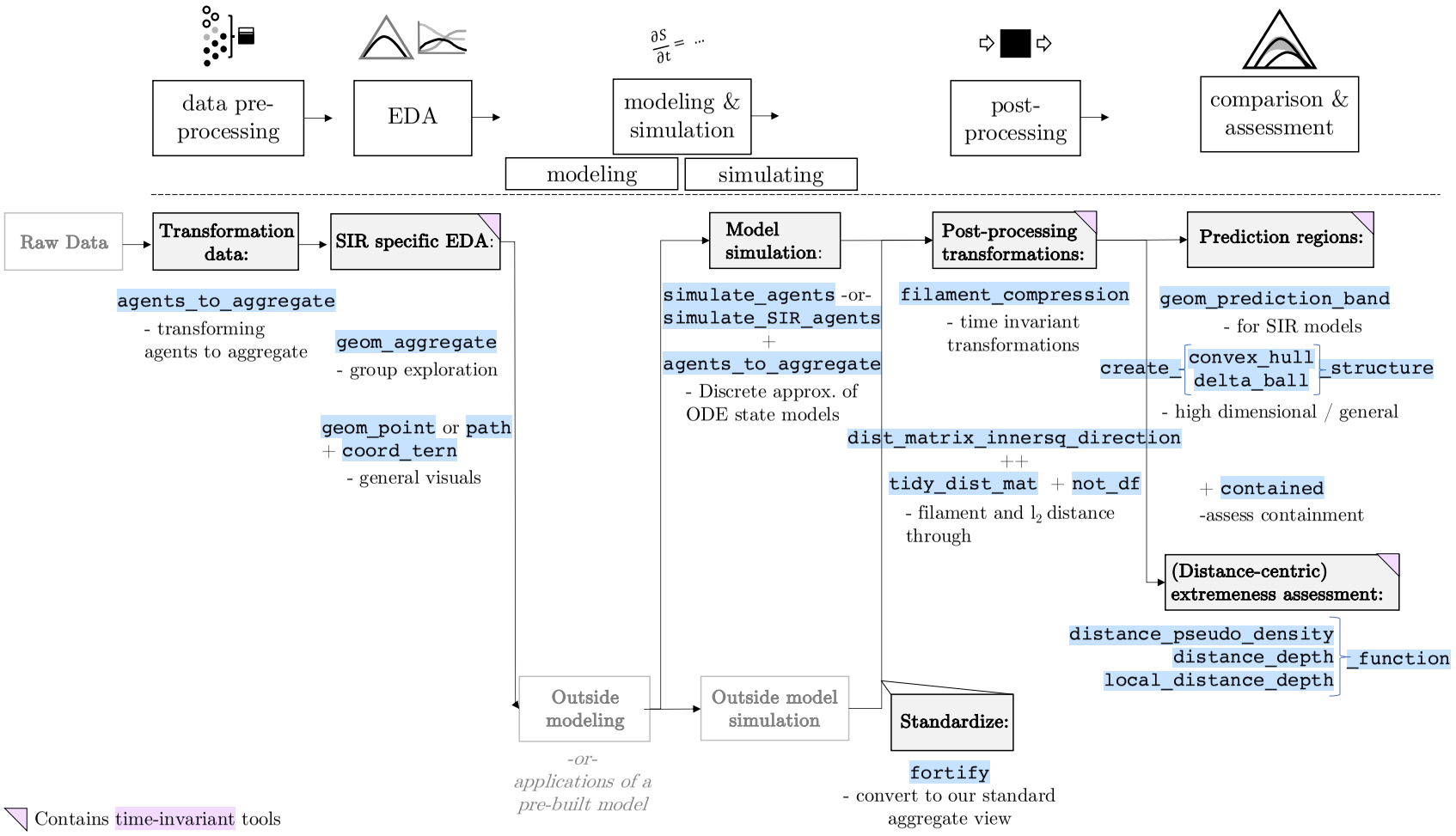

Figure 4: How **EpiCompare** supplements and aids in the epidemiological data analysis pipeline.

with times of different states of the epidemic (infection, recovery, etc.) as well as individual information like network structure, location, and sub-population information. Other data collections focus on aggregate counts of individuals in each epidemic state. In fact, many times only the number of new infections at each time step (e.g. weekly case counts) is observed. In this setting, compartment totals (amounts of individuals in each state) are then imputed from those case counts and using other information about the disease and the population of interest. In **EpiCompare**, we focus on understanding the overall impact of an outbreak at the aggregate/population level, which allows for streamlined examination of overall trends of an epidemic.

To help the practitioner examine epidemics from an aggregate/population lens, we provide a function called `agents_to_aggregate()`. This function transforms data about individual/agents' initial entry into each state (e.g. start of infection, start of recovery, etc.) to an aggregate view of how many individuals were in a state at a given time. Researchers, including Rvachev and Longini (1985); Anderson and May (1992); Worby *et al.* (2015), often are interesting in more granular trends that can be detected by aggregation, conditional on subpopulations (e.g. subpopulations defined by age or sex). By combining the function **dplyr**::`group_by()` and `agents_to_aggregate()`, **EpiCompare** provides group level aggregation.

Besides aiding subpopulation analysis, `agents_to_aggregate()` can accommodate a wide range of information about each individual. In fact, this function can account for infinitely many states. This functionality allows the practitioner to aggregate information relative to common states (e.g. "Susceptible", "Infectious", and "Recovered") as well as more complex states (e.g. "Exposed", "iMmune", "Hospitalized"). Additionally, `agents_to_aggregate()` permits indicators for death/exit and birth/entry dates. Overall, this function is a powerful tool for pre-processing data.

Exploratory data analysis (EDA)

In the early stages of a project, familiarizing oneself with the data usually means figuring out useful combinations of visualizations and numerical summaries of the data both at population and subpopulation level. An expert coder can start with `agents_to_aggregate()` to successfully accomplish exploratory data analysis (EDA) in many ways. **EpiCompare** also includes tools that allow a novice coder to rapidly explore data, provided there are three unique epidemiological states (like in the SIR model). Building on **ggplot2** and **ggtern** packages, **EpiCompare**'s `geom_aggregate()` provides a way to explore how different subpopulations experience of an epidemic (Wickham 2016; Hamilton and Ferry 2018). The function `geom_aggregate()` provides a visualization tool to holistically examine aggregate level information across different subpopulations by visualizing each subpopulation's epidemic trajectory in the three-dimensional state space. Visualization tools for three-state models were developed because SIR models are some of the most common and basic epidemic state-based models and our three-dimensional simplex representation of these epidemics emphasizes a time-invariant representation of the data (for a refresher see Section 2).

Model fitting and simulations

After getting a sense of what a past or current epidemic looks like with EDA, the next step in the data analysis pipeline is often model fitting and/or simulation. While **EpiCompare** does not focus on fitting models to data, we do provide some flexible functions for simulation of basic discrete-time epidemic-state models. These functions simulate individual-level informa-

tion based on practitioner estimated transition rates between states and can be combined with `agents_to_aggregate()` to view these simulations through an aggregate lens. The function `simulate_SIR_agents()` simulates a basic SIR epidemic with user inputs for the number of simulations, the initial number in each state, the infection and recovery parameters (β, γ), and the total number of discrete time steps. Beyond SIR models, the function `simulate_agents()` takes as input a user-specified state-transition matrix and other epidemic parameters to allow the user to create simulations for an outbreak with *any* number of states and any number of transitions among them. This flexibility in states can be used to also reflect group-based dynamics. Both of these functions allow users to explore the space of models in an intuitive way without getting bogged down by too much mathematical detail. For consistency, we have made output from `simulate_agents()` and `simulate_SIR_agents()` compatible with `agents_to_aggregate()` so aggregate information may easily be accessed.

Post-processing

If practitioners wish to compare models-to-observations or even models-to-models, they need to post-process their models and simulations to disseminate the results in an easily digestible format. In **EpiCompare**, we provide (1) functions to standardize simulation and model output from external packages and (2) a function to transform standardized simulation and model output into a format amenable to time-invariant analysis.

Modeling and simulation output can be very complex objects, and as a result, a number of epidemic modeling R packages return a special class. The special classes often contain a plethora of information about residuals, model diagnostics, input parameters, and more. While incredibly useful, these special classes can be difficult for novice coders to handle. To this end, **EpiCompare** provides a series of fortify-style methods, called `fortify_aggregate()` which transform output from infectious disease modeling and simulation packages like **pomp** and **EpiModel** into tidy-styled data frames which contain information about the total number of individuals in each state at a given time, for a given simulation. These fortify functions have output that is consistent with that of `agents_to_aggregate()`. These standardized outputs can then be piped to summaries, tables, and plots.

Because epidemic data is stored in a temporal way, we provide the function, `filament_compression()`, to transform temporally defined epidemics to their filamental representations. These filaments can then be fairly compared to one another or passed to further time-invariant analysis tools described below.

Comparisons and assessment

The last step of the data analysis pipeline often ends with plots, tables, and summary statistics that are used to assess model performance and compare across models or simulations. In **EpiCompare** we provide a set of comparison and assessment tools for model and simulation results that extend beyond the standard performance metrics (e.g. mean squared error or AIC) and into the lens of time-invariant analysis. We have found that these tools are specifically applicable for situations where only one season or cycle of an epidemic has occurred or is the object of interest.

The first set of of tools surround the creation of prediction regions. We can create a prediction regions from model simulations to examine if our model simulations capture the true epidemic trajectory. We do so in a time-invariant way and utilizing filamental representations of the model simulations and the true epidemic. For three-state epidemic models, we provide the `ggplot/ggtern` extension `geom_prediction_band()` which creates a prediction region around

the top $1 - \alpha$ proportion of the simulations. In this visual setting, comparing this prediction region to the true epidemic trajectory can be done by eye. In **EpiCompare**, we also provide these prediction regions for epidemic models with with more than three states. The functions `create_convex_hull_structure()` and `create_delta_ball_structure()` create different geometric representations of prediction regions for any dimensional state-based model. For both of these geometric structures, we provide functions to check if a path is contained (`contained()`)]. We can also use these prediction regions to visually or mathematically compare how similar two sets of simulations are. In **EpiCompare** we provide the `hausdorff_dist()` function to calculate the Hausdorff distance between multiple prediction regions, when visual comparison is not possible.

We also provide functions to calculate the "extremeness" of a true epidemic trajectory compared to simulated epidemics via the equi-distance filamental trajectory representation as mentioned in Section 2.2.] We provide implementations of a few distance-based score functions that capture how "reasonable" an epidemic is relative to other epidemics, and these scores can be turned into an extremeness measure with `mean(sim_scores > truth_score)`. [Specifically, functions like `distance_pseudo_density_function()` can calculate a pseudo-density estimate of the true epidemic relative to simulated ones. Functions `distance_depth_function()` and `local_distance_depth_function()` provide depth scores that suggest how geometrically central an epidemic is to simulations.

4. A tour of EpiCompare

I have substantially overhauled section 4. The older versions are kept after the entirety of this section but note that much of the text (but not the structure) of this section is different. I've tried to better motivate why we use the epicompare functions to answer our questions of interest.

[[NEWEST TEXT

To conclude our paper, we demonstrate how **EpiCompare** can streamline the data analysis process with a case study of a measles outbreak in 1861-1862 Germany. With the help of **EpiCompare**, we can answer questions such as is a SIR model a good fit for the data, does the outbreak spread differently within the different school classes of the children, and can incorporating an underlying network structure enhance model fit. We first begin with some context for the epidemic.

4.1. Background for 1861-1862 measles epidemic

We analyze an outbreak of measles in the town of Hagelloch, Germany from 1861-1862, a data set organized by Pfeilsticker (1863). The data was later made visible by Oesterle (1992) and made available in an R by Meyer *et al.* (2017). In this outbreak, 188 children were infected with measles over the course of three months. This data set includes a rich collection of features including day of measles rash, age, sex, school class, household and household location, and alleged infector of each child. We show a subset of the data in Table 1. We are particularly interested if and how group structure plays a role in the spread of measles.

4.2. Pre-processing and EDA

Table 1: Subset of Hagelloch infection data. Features include the person ID, household ID (HH ID), age, sex, class level (Pre-K/1st/2nd), date of first symptoms, date of the appearance of the measles rash, and the alleged infector ID of the individual.

ID	HH ID	Name	Age	Sex	Class	Symp. Start	Rash Date	Infector ID
1	61	Mueller	7	female	1st class	1861-11-21	1861-11-25	45
2	61	Mueller	6	female	1st class	1861-11-23	1861-11-27	45
3	61	Mueller	4	female	preschool	1861-11-28	1861-12-02	172
4	62	Seibold	13	male	2nd class	1861-11-27	1861-11-28	180
5	63	Motzer	8	female	1st class	1861-11-22	1861-11-27	45
45	51	Goehring	7	male	1st class	1861-11-11	1861-11-13	184

We begin our analysis by first examining and transforming the raw data (`hagelloch_raw`), which is individual-level data. With **EpiCompare**, we can quickly determine how the epidemic progresses through the population at an aggregate level. By specifying the time of infection (`tI`) and the time of recovery (`tR`), the function `agents_to_aggregate()` calculates the number susceptible, infectious, and recovered individuals at each time step. Once aggregated, we can plot the SIR values through a time-invariant lens using **ggplot2** and **ggtern** functions (as shown in Fig. 9) or with our custom `geom`, `geom_aggregate()`, which takes the raw agent data as input.

```
R> hagelloch_sir <- hagelloch_raw %>%
+   agents_to_aggregate(states = c(tI, tR),
+                       min_max_time = c(0, 55)) %>%
+   rename(time = t, S = X0, I = X1, R = X2)
R>
R>
R> ggplot(hagelloch_sir, aes(x = S, y = I, z = R))+
+   coord_tern() +
+   geom_path() +
+   labs(x = "S", y = "I", z = "R",
+        title = "Time invariant view of Hagelloch measles outbreak") +
+   theme_sir(base_size = 24)
```

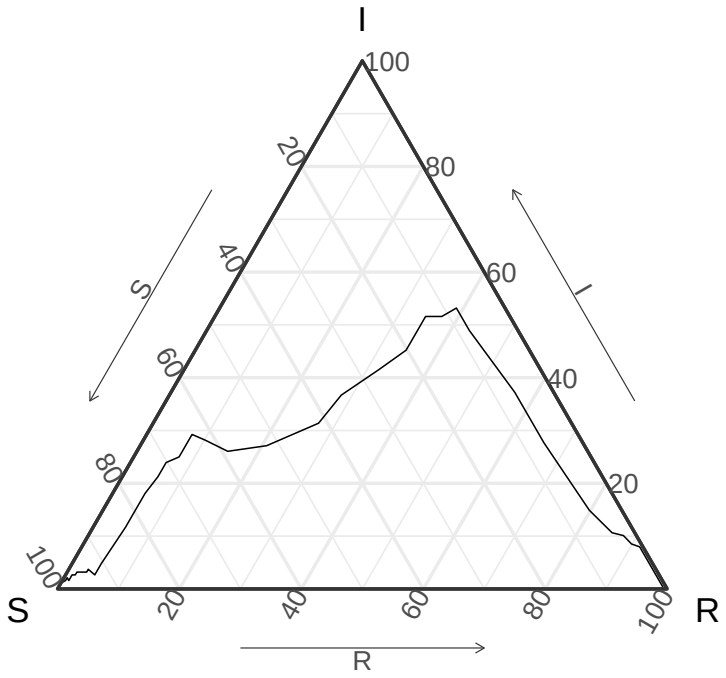

Figure 5: Time invariant view of the Hagelloch epidemic where we view the individuals in Susceptible, Infectious, or Recovered states. We see there are two peaks of infection (the vertical axis).

In Figure 9, we can focus on the infections over time by analyzing the vertical axis. Specifically, we see two peaks of infection. This is interesting because the SIR model, which is sometimes used to model the spread of measles, generally has one defined peak of infection. We may wonder if the two peaks in the observed data may be due to random noise or if a model more complex than the simple SIR is needed to adequately capture these two peaks.

Previous study tells us that measles outbreaks are often associated with children within the same grade level, and we examine if this is the case here. By combining the function `facet_wrap()` with `geom_aggregate()` we can easily analyze this scenario,

```
R> hagelloch_raw %>%
+   ggplot(aes(y = tI, z = tR, color = CL)) +
+   geom_aggregate(size = 2) + coord_tern() +
+   labs(x = "S", y = "I", z = "R",
+        color = "Class") +
+   scale_color_brewer(palette = "Dark2") +
+   facet_wrap(~CL)
```

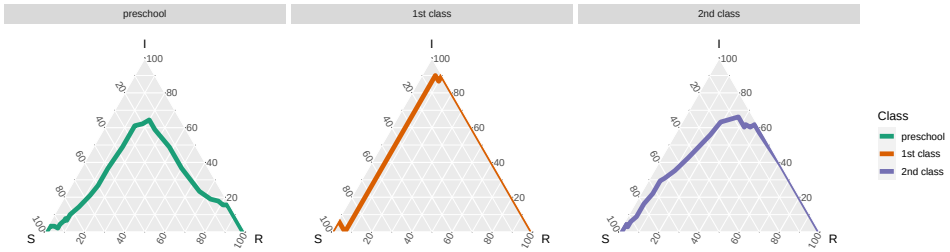

Figure 6: Time invariant outbreak curves for the three class groups. The pre-school class has a distinct peak of infection whereas the peak infection point for the other two classes are less well defined.

Immediately in Fig. 10, we see that time invariant infection curve is different for the pre-school class compared to the 1st class. In the 1st class, we see about 95% of the class become infected and less than 10% of them having recovered, which may be indicative of a super-spreading event. This suspicion is further supported in that 26 of the 30 1st class students have been reportedly infected by the same individual. We now have some evidence that class structure may play a role in the spread of infection. We can further analyze this claim with modeling and simulation.

4.3. Modeling and Simulation

We first try to model the Hagelloch data with a baseline stochastic SIR model, which we refer to as the 'simple SIR.' In our full vignette (available online), we show how to fit this simple SIR model via maximum likelihood, a common approach used to fit parameters, and simulate from the model with those best fit parameters. Our function `simulate_agents()` (or `simulate_SIR_agents()`) generates individual level data according to discrete-time multinomial draws, which depend on the number of individuals in each state at the previous time step and a matrix of transition probabilities. For example, the below code generates 100 simulations of an outbreak of a disease with one initial infector in a population of $n = 188$ individuals, a scenario analogous to the actual outbreak.

```
R> trans_mat <- matrix(c("X0 * (1 - X1 * par1 / N)", "X0 * X1  * par1 / N", "0",
+                        "0", "X1 * (1 - par2)", "par2 * X1",
+                        "0", "0", "X2"), byrow = TRUE, nrow = 3)

R> set.seed(2020)
R>
R> best_params <- c("beta" = .36, "gamma" = .13)
R> ## This is the SIR representation
R>
R> rownames(trans_mat) <- c("S", "I", "R")
R> init_vals <- c(187, 1, 0)
R> par_vals <- c(par1 = best_params[1], par2 = best_params[2])
R> max_T <- 55
R> n_sims <- 100
R>
```

```
R> agents <- simulate_agents(trans_mat,
+                         init_vals,
+                         par_vals,
+                         max_T,
+                         n_sims,
+                         verbose = FALSE)

R> agg_model <- agents %>% group_by(sim) %>%
+   agents_to_aggregate(states = c(I, R)) %>%
+   mutate(Type = "Simple SIR")
```

The result of our simulation is the object `agents` which is a 18800 $\times$ 5 data frame, which details the time of entry into the S, I, and R states for a given simulation. Ultimately, we would like to know if this simple SIR model is a good fit to the data, especially in comparison to a SIR model where we can incorporate the class structure.

To fit a more complex SIR model with a network structure, we use package **EpiModel** (Jenness *et al.* 2018). The below code sets up a network of individuals (which includes class as a variable) and then simulates infections over this network.

```
R> library(EpiModel)
R> ## WARNING:  Will take a minute or two
R>
R> set.seed(42)
R> nw <- network.initialize(n = 188, directed = FALSE)
R> nw <- set.vertex.attribute(nw, "group", rep(0:2, each = 90, 30, 68))
R> formation <- ~edges + nodematch("group") + concurrent
R> target.stats <- c(200, 300, 200)
R> coef.diss <- dissolution_coefs(dissolution = ~offset(edges),  duration = 5)
R> est1 <- netest(nw, formation, target.stats, coef.diss, edapprox = TRUE)
R>
R> param <- param.net(inf.prob = 0.1, act.rate = 5, rec.rate = 0.1)
R> status.vector <- c(rep(0, 90), rep(0, 30), rep(0, 67), 1)
R> status.vector <- ifelse(status.vector == 1, "i", "s")
R> init <- init.net(status.vector = status.vector)
R> control <- control.net(type = "SIR", nsteps = 55,
+                         nsims = 100, epi.by = "group")
R> epimodel_sir <- netsim(est1, param, init, control)
```

The output of this network model is `epimodel_sir`, an object of class `netsim`, which contains a plethora of modeling information. In the following section, we will use the capabilities of **EpiCompare** to streamline the process of comparing the two models contained in the objects `agg_model` (the simple SIR) and `epimodel_sir` (the network SIR model).

4.4. Post-processing and comparison

The next step is to compare the simple SIR model to the EpiModel SIR model. The **EpiCompare** function `fortify_aggregate()`, takes in an object from specialized classes of modeling output (like those made by `netsim()`) and transforms it into a tidy-style data frame.

```
R> fortified_net <- fortify_aggregate(epimodel_sir,
+                                     states = c("s.num", "i.num", "r.num")) %>%
+   mutate(Type = "EpiModel SIR",
+          sim = as.numeric(gsub("sim", "", sim)))
```

With the two modeling objects both in the same format, we can then compare the models side-by-side. The results are shown in Figure 11, where a 90% prediction region is estimated for the two models. For the Simple SIR model, we see that while the prediciton region covers the data fairly well, the prediction region clearly misses the second peak of infection. This indicates that the simple SIR is not a good fit to our data. We also see that the prediction region is very large, covering up a large area of the ternary plot. Together, this indicates that the simple SIR model produces a biased model with a large amount of variance. On the other hand, for the EpiModel network model, we see that the prediction region covers the data quite well and takes up less area. With **EpiCompare**, we can see that the model using the class structure is a better fit to the outbreak than the simple SIR model.

```
R> both_models <- bind_rows(agg_model, fortified_net)
R>
R>
R> g <- ggplot() + geom_prediction_band(data = both_models %>% filter(t != 0) %>%
+                   mutate(Type = factor(Type, levels = c("Simple SIR",
+                                                         "EpiModel SIR"))),
+          aes(x = X0, y = X1, z = X2,
+               sim_group = sim, fill = Type),
+          alpha = .5,
+          conf_level = .90)
```

```
R> g +   geom_path(data = both_models %>% filter(t != 0) %>%
+                   mutate(Type = factor(Type, levels = c("Simple SIR",
+                                                         "EpiModel SIR"))),
+             aes(x = X0, y = X1, z = X2, group = paste(Type, sim)),
+             alpha = .3, col = "gray40") +
+     coord_tern() + theme_sir(base_size = 24) +
+   geom_point(data = hagelloch_sir,
+              aes(x = S, y = I, z =R), col = "black") +
+   labs(title = "Simple SIR model",
+        subtitle = "90% Prediction band and original data",
+        x = "S", y = "I", z = "R") +
+   scale_fill_manual(values = c("#006677", "#AA6600")) +
+   facet_wrap(~Type) +
+   theme(legend.position = "bottom")
```

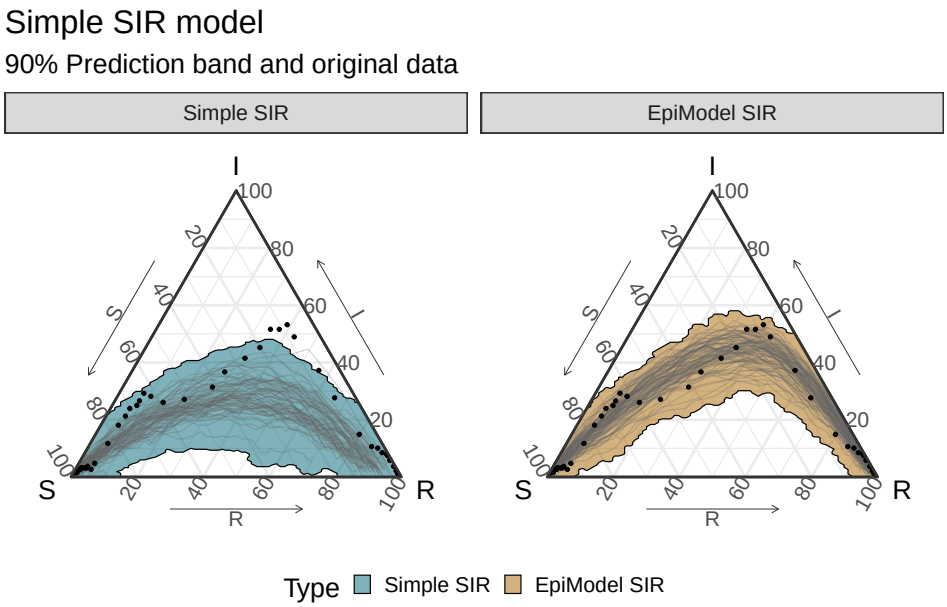

Figure 7: Original Hagelloch SIR data (black) along with 90% prediction band and actual simulation paths from the Simple SIR and the EpiModel SIR models.

The prediction region generated by the network model covers the observed data well, but that does not mean that individual filaments generated from the network model are good fits to the observed data. We can further examine the model fits by incorporating these filaments into our visualization. In Fig. 11 we show the indivdual filaments generated from the two sets of models as gray lines. Examining these, we see that the individual lines typically only have one defined peak, whereas the data certainly looks like it has two distinct peaks, likely caused by our assumed super-spreader event.

We can also examine these filaments quantitatively by using functions that take the distance between the filaments with the observed data. In the below code, we transform the simulations to a more computationally-friendly format with the function `filament_compression()`. Following that, we calculate the distance between the simulated filaments and the observed filament with the function `dist_matrix_innersq_direction()` and calculate the probability of the truth with respect to those simulations with the function `compare_new_to_rest_via_distance()`. The estimated pseudo-density of the observed epidemic (relative to the simulations from either model) is much less likely then **any** of the simulations (reported in Table 6). This indicates that neither of two SIR models are good fits to the data at the filament level.

```
R> simple_sir <- both_models %>% filter(Type == "Simple SIR") %>%
+   rename(S = "X0", I = "X1", R = "X2") %>%
+   select(Type, sim, t, S, I, R)
R>
R> hagelloch_sir2 <- hagelloch_sir %>%
+   rename(t = "time") %>%
+   mutate(Type = "true observation",
+          sim = 0) %>%
+   select(Type, sim, t, S, I, R)
```

```
R> #-- after cleaning up and combining --
R> all_together_df <- rbind(simple_sir,
+                           hagelloch_sir2)
```

Table 2: Top and bottom 2 rows of `all_together_df`, combining both simulated epidemics and the true epidemic.

Type	sim	t	S	I	R
Simple SIR	1	0	188	0	0
Simple SIR	1	1	187	1	0
true observation	0	54	1	0	187
true observation	0	55	1	0	187

```
R> compression_df <- all_together_df %>% group_by(Type, sim) %>%
+   filament_compression(data_columns = c("S","I","R"),
+                        number_points = 20)

R> tdmat <- compression_df %>%
+   dist_matrix_innersq_direction(
+     position = c(1:length(compression_df))[
+       names(compression_df) %in% c("S","I", "R")],
+     tdm_out = T)
R>
R> simple_sir_true_obs_info <- tdmat %>%
+   compare_new_to_rest_via_distance(
+     new_name_id = data.frame(Type = "true observation", sim = 0),
+     distance_func = distance_psuedo_density_function,
+     sigma = "20%")
```

Table 3: The extremeness of the true simulations based on comparing pseudo-density estimates between true vs simulated curves

Type	simulations-based estimated pseudo-density	proportion of simulations with lower estimated pseudo-density
Simple SIR	0.0036733	0.00
EpiModel SIR	0.0118283	0.03

In conclusion, **EpiCompare** allows us to fully examine this outbreak at every step in the data analysis pipeline (see Fig. 4) in a streamlined fashion. With EDA, we saw evidence that class structure may be important in the spread of of measles. We then compared a baseline simple SIR model to a more complicated SIR model which incorporated a network structure which included the class structure. Based on the prediction regions generated from

these models, we saw that the network model fit the data much better than the simple SIR model. However, when we examined the individual filaments generated by the network model, we found that the data is highly unlikely to be generated from such a model. For further analysis, we would recommend looking into models that can more accurately capture super-spreading events based on the observation that one child was allegedly responsible for nearly all of his classmates' infections. Overall, this analysis demonstrates how **EpiCompare** aids in the data analysis pipeline for both novice and expert practitioners and coders alike.

]]

[[NEWER TEXT

[13] To conclude our paper, we demonstrate the capabilities of **EpiCompare** with a complete data analysis of a measles outbreak in 1861-1862 Germany. Specifically, we demonstrate how tools in **EpiCompare** can be used in each step of the data analysis pipeline [14] ~~(see Figure 1)~~[15]. Additionally, we highlight how time-invariant analysis (see Section 2) can be used to enhance understanding of an outbreak[16].

[17] Before demonstrating **EpiCompare**, we provide some context for the measles outbreak presented here.[18] The data was originally organized by Pfeilsticker (1863), later made visible by Oesterle (1992), and made available in an R by Meyer *et al.* (2017). This data set includes a rich collection of features including household location, class level, and allged infector ID, and is an ideal testing ground for methodology in infectious disease epidemiology Neal and Roberts (2004); Britton *et al.* (2011); Groendyke *et al.* (2012); Becker *et al.* (2016)[19]. In this data set, there are 188 children who became infected with the measles over the course approximately 90 days.

]]

[[LESS NEW TEXT

Finally, in this section we show how tools from **EpiCompare** can be used in each step of the data analysis pipeline shown in Fig. 1. We analyze an outbreak of measles in the town of Hagelloch, Germany from 1861-1862, a data set organized by Pfeilsticker (1863). The data was later made visible by Oesterle (1992) and made available in an R by Meyer *et al.* (2017). This data set includes a rich collection of features and is an ideal testing ground for methodology in infectious disease epidemiology Neal and Roberts (2004); Britton *et al.* (2011); Groendyke *et al.* (2012); Becker *et al.* (2016).[20]

---

[13] [Ben says: Generally I found this whole section to be pretty passive and not well motivated on why we'd actually do the analysis.]

[14] [Ben says: based on an earlier read of this section, I might suggest something like "and streamline the analysis process" - is that a selling point you want to highlight?] check out new re-write. new goal is to motivate question of hagelloch and then show how epicompare helps

[15] [Ben says: It's unclear why you want to highlight the figure again - could you be clearere on that? / what does it add to the conv/ why should they reference it?]

[16] [Ben says: I'm not sure this is necessary to state. Additionally - is it really true do we highlight this?]

[17] [Ben says:The data introduction could be another subsection. Why does it fit better here?. Other comment: This paragraph is pretty rambly. It is unclear what you want someone to take away from it. I might suggest highlight the fact that the data is "an ideal testing group for methodology".]

[18] [Ben says: this first sentence is pretty sign-post-y yet it only relates to the next few sentences. Update so it's less sign-post-y.]

[19] [Ben says: This citation doesn't make sense - should it be "citep"?]

[20] The old first paragraph from data and exploratory analysis paragraph was combined with the intro as a better lead-in to what's going on.

]]
[[OLD TEXT
[21]In this section, we highlight many of the tools available in **EpiCompare**. As previously discussed, these tools include data cleaning; visualization; modeling and simulation; post-processing; and comparison and model assessment, in accordance with the data analysis pipeline (Fig. 1). We show a full data analysis from beginning to end that can be accomplished in a streamlined and standardized manner via **EpiCompare**.]]

4.5. Pre-processing and EDA

[22]The Hagelloch data include a rich set of features at the individual level, and the tools in **EpiCompare** help with pre-processsing and EDA. Recorded features include household members, school level, household locations, date of first symptoms (prodromes), date of measles rash, and even the alleged infector. A subset of the data is shown in Table 1. For example,[23] with **EpiCompare**, we can easily pre-process the data to obtain the empirical cumulative incidence function with respect to the measles rash appearance (variable `ERU`) with the following tidy-style function, `agents_to_aggregate()`. The function `agents_to_aggregate()` is a key component of **EpiCompare**, allowing the user to easily switch from an individual-level (i.e. an agent) ~~view~~ lens of a disease to an aggregate ~~level~~ lens. For example[24], the below code shows how we can convert the agent data to a cumulative incidence plot of the measles rash~~, in order to see how the disease spread through the population over time.~~ We can then compare the cumulative incidence of the rash to the cumulative incidence of the prodromes, i.e. the initial[25] symptoms[26]. We do this with the below code, and a part of the cumulative incidence data output is shown in Table 4. The argument `integer_time_expansion` indicates whether we should include all time points in the recorded range of the data or only when there is a change in the incidence.

```
R> cif_rash  <- hagelloch_raw %>%
+   mutate(time_of_rash = as.numeric(ERU - min(PRO, na.rm = TRUE))) %>%
+   agents_to_aggregate(states = time_of_rash,
+                       integer_time_expansion = FALSE) %>%
+   mutate(type = "Rash")
```

One possible question of interest is the duration between initial onset of prodromes and the appearance of the measles rash[27]. Since `agents_to_aggregate()` outputs a tidy-style data frame, it is a simple task to plot the two sets of incidence curves on the same graph (Fig. 8).

[21][Ben says: Shannon, would you mind reading this whole section over again once we've finished edits for section 2 and 3? This initial paragraph seems to be stating section 3's story.]
[22][Ben says: The first two sentences is very similar to the data paragraph in the section above. Given it's not really connecting the 2 sections I suggest a rewrite - could move some of this stuff above.]
[23][Ben says: I'm unclear of what this is actually an example of.]
[24][Ben says: Same comment as before.]
[25][Ben says: please be clearer on what these could be given you comparing them to start of the rash - which seems like an early symptom to me...]
[26][Ben says: This action in the analysis pipeline is unmotivated - which naturally makes me want to ask "why would I do this?"]
[27][Ben says: You don't give a good definition of prodromes above, and you only use the name twice. Is this a super common term in Epi? I find it a bit taxing on the reader to remember what this is referring to.]

Table 4: Turning the individual-level information from the Hagelloch data to an aggregate view of the cumulative incidence of the measles rash in the population over time.

Time	# Susceptible	# Total rash appearances
0	188	0
4	187	1
7	186	2
9	185	3
12	183	5

```
R> cif_prodromes <- hagelloch_raw %>%
+   mutate(time_of_PRO = as.numeric(PRO - min(PRO, na.rm = TRUE))) %>%
+   agents_to_aggregate(states = time_of_PRO,
+                       integer_time_expansion = FALSE) %>%
+   mutate(type = "Pro")
```

[28]

```
R> plot_df <- bind_rows(cif_rash, cif_prodromes)
R>
R> ggplot(data = plot_df,
+        aes(x = t, y = X1, col = type)) +
+   geom_step() +
+   labs(title = "Cumulative incidence of measles appearance",
+        x = "Time (days relative to first prodrome appearance)",
+        y = "Cumulative incidence of event") +
+   coord_cartesian(xlim = c(0, 55)) +
+   scale_color_manual(values = c("blue", "red"))
```

[28] I'm confused why Figure 5 is included. What is the conclusion you'd like to take away? / Why do people create plots like this?

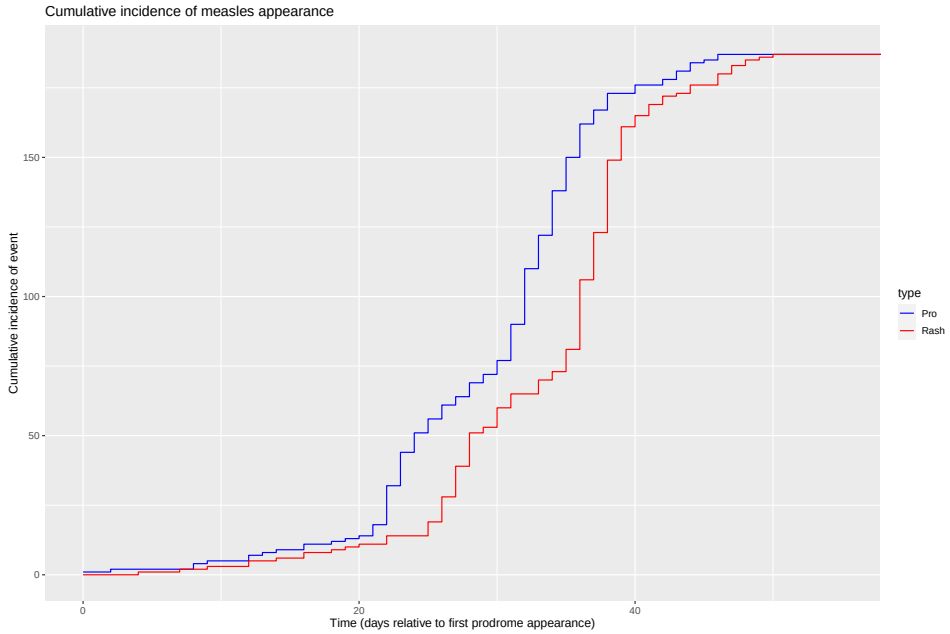

Figure 8: Empirical cumulative incidence functions of prodrome (symptom) onset and measles rash appearance. We see that there is approximately a a constant lag between the two curves.

The real power of `agents_to_aggregate()` lies in its ability to aggregate over any number of pre-specified states. For example, the Hagelloch data sets contains two columns, `tI` and `tR`, the time of infection and recovery, respectively of each individual. We can then[29] plot the SIR values through a time-invariant lens using **ggplot2** and **ggtern** functions (as shown in Fig. 9) or with our custom `geom`, `geom_aggregate`, which takes the raw agent data as input.

```
R> hagelloch_sir <- hagelloch_raw %>%
+   agents_to_aggregate(states = c(tI, tR),
+                       min_max_time = c(0, 55)) %>%
+   rename(time = t, S = X0, I = X1, R = X2)
R>
R>
R> ggplot(hagelloch_sir, aes(x = S, y = I, z = R))+
+   coord_tern() +
+   geom_path() +
+   labs(x = "S", y = "I", z = "R",
+        title = "Time invariant view of Hagelloch measles outbreak") +
+   theme_sir(base_size = 24)
```

[29] [Ben says: using "then" here captures a very progression step of analysis but I stopped here and asked "what is this following" - and the previous "step" occurred a paragraph back and wasn't described as a direct progression but just a possible thing to do.]

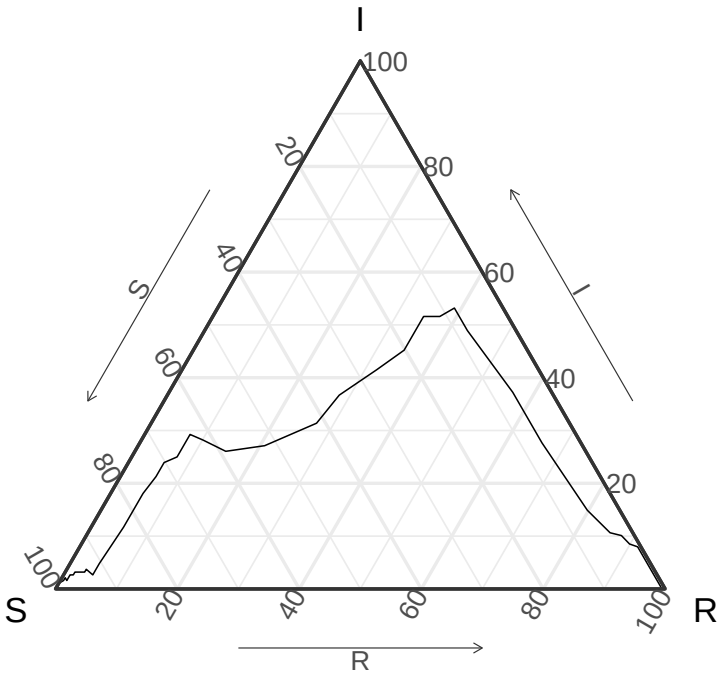

Figure 9: Time invariant view of the Hagelloch epidemic where we view the individuals in Susceptible, Infectious, or Recovered states. We see there are two peaks of infection (the vertical axis).

[30]Moreover, we can look at the outbreaks of the disease by group within `agent_to_aggregate()` or `geom_aggregate()`. This allows us to examine differences among the different groups of individuals. For example, we show the time invariant outbreak by class level in Figure 10. Immediately, we see that time invariant infection curve is different for the pre-school class compared to the 1st class. In the 1st class, we see about 95% of the class become infected and less than 10% of them having recovered, which may be indicative of a super-spreading event. This suspicion is further confirmed in that 26 of the 30 1st class students have been reportedly infected by the same individual.

```
R> hagelloch_raw %>%
+   ggplot(aes(y = tI, z = tR, color = CL)) +
+   geom_aggregate(size = 2) + coord_tern() +
+   labs(x = "S", y = "I", z = "R",
+        color = "Class") +
+   scale_color_brewer(palette = "Dark2") +
+   facet_wrap(~CL)
```

[30]I found this pargraph very unmotivated. I recommend first arguing why we might a care to look into the class subpopulation grouping. And maybe comment that is is a common desire for practioners.

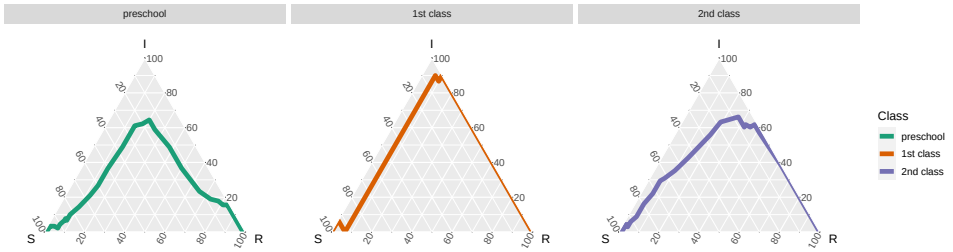

Figure 10: Time invariant outbreak curves for the three class groups. The pre-school class has a distinct peak of infection whereas the peak infection point for the other two classes are less well defined.

~~Along with multiple epidemic states, the function~~ `agents_to_aggregate()` ~~can also be extended to populations with vital dynamics (e.g. birth and death) and examples of this are shown in the package vignette. In summary,~~ `agents_to_aggregate()` ~~is a multi-purpose workhorse that may be leveraged to convert individual level records into aggregate information that may be more useful for some forms of epidemic modeling such as compartment modeling.~~[31]

4.6. Modeling and simulation

[32]

Up to this point, we have used **EpiCompare** in the context of observed data. [33]We also want to compare statistical models, and **EpiCompare** aids in that process via a simple yet flexible individual-level simulator~~, conversion tools for popular epidemic model packages, and model assessments.~~ We demonstrate an example[34] here.

We first try to model the Hagelloch data with a stochastic SIR model, which we refer to as the 'simple SIR.'[35] In our full vignette (available online), we show how to fit this simple SIR model via maximum likelihood and simulate from the model with those best fit parameters[36]. Our function `simulate_agents()`\footnote{[Ben Should somethings be said about the `simulation_SIR_agents()` function?]} generates individual level data according to discrete time multinomial draws, which depend on the number of individuals in each state at the previous time step and a matrix of transition probabilities. For example, the below code generates 100 simulations of an outbreak of a disease with one initial infector in a population of $n = 188$ individuals.

```
R> trans_mat <- matrix(c("X0 * (1 - X1 * par1 / N)", "X0 * X1  * par1 / N", "0",
+                        "0", "X1 * (1 - par2)", "par2 * X1",
+                        "0", "0", "X2"), byrow = TRUE, nrow = 3)

R> set.seed(2020)
R>
```

[31] Is this not just a repeat of section 3?
[32] section headings to align with our pipeline
[33] [Ben says: Why?]
[34] [Ben says: to me this whole section is 1 example - as such this wording is confusing to me.]
[35] [Ben says: could / should this be thought of as a "base" model?]
[36] [Ben says: should you highlight that this is a common approach?]

```
R> best_params <- c("beta" = .36, "gamma" = .13)
R> ## This is the SIR representation
R>
R> rownames(trans_mat) <- c("S", "I", "R")
R> init_vals <- c(187, 1, 0)
R> par_vals <- c(par1 = best_params[1], par2 = best_params[2])
R> max_T <- 55
R> n_sims <- 100
R>
R> agents <- simulate_agents(trans_mat,
+                         init_vals,
+                         par_vals,
+                         max_T,
+                         n_sims,
+                         verbose = FALSE)

R> agg_model <- agents %>% group_by(sim) %>%
+   agents_to_aggregate(states = c(I, R)) %>%
+   mutate(Type = "Simple SIR")
```

The result of our simulation is the object `agents` which is a 18800 $\times$ 5 data frame, which details the time of entry into the S, I, and R states for a given simulation. [37]Before we examine the results of this simple SIR model, we will also examine another, more sophisticated SIR model, this time from the package **EpiModel** (Jenness *et al.* 2018). Briefly, this model first fits a contact network to the set of individuals, where the class of the child is a covariate[38]. The model then simulates a SIR-epidemic on that network.

```
R> library(EpiModel)
R> ## WARNING:  Will take a minute or two
R>
R> set.seed(42)
R> nw <- network.initialize(n = 188, directed = FALSE)
R> nw <- set.vertex.attribute(nw, "group", rep(0:2, each = 90, 30, 68))
R> formation <- ~edges + nodematch("group") + concurrent
R> target.stats <- c(200, 300, 200)
R> coef.diss <- dissolution_coefs(dissolution = ~offset(edges),  duration = 5)
R> est1 <- netest(nw, formation, target.stats, coef.diss, edapprox = TRUE)
R>
R> param <- param.net(inf.prob = 0.1, act.rate = 5, rec.rate = 0.1)
R> status.vector <- c(rep(0, 90), rep(0, 30), rep(0, 67), 1)
R> status.vector <- ifelse(status.vector == 1, "i", "s")
R> init <- init.net(status.vector = status.vector)
R> control <- control.net(type = "SIR", nsteps = 55,
```

[37] [Ben says: please motivate - through model comparison?]

[38] [Ben says: Make this connect better to the problem at hand - why do you think you should build this bigger model?]

```
+                          nsims = 100, epi.by = "group")
R> epimodel_sir <- netsim(est1, param, init, control)
```

The output of this model is `epimodel_sir`, an object of class `netsim`, which contains a plethora of modeling information. [39]

4.7. Post-processing and comparison

The next step is to compare the simple SIR model to the EpiModel SIR model. We provide[40] the function `fortify_aggregate()`, which can take objects from specialized classes of modeling output (like those made by `netsim()`) and transform it into a tidy-style data frame.

```
R> fortified_net <- fortify_aggregate(epimodel_sir,
+                                     states = c("s.num", "i.num", "r.num")) %>%
+   mutate(Type = "EpiModel SIR",
+          sim = as.numeric(gsub("sim", "", sim)))
```

We can then analyze the results of the two models side by side as time-invariant[41] epidemic curves. The results are shown in Figure 11, where a 90% prediction band is estimated from the delta ball[42] method for each of the two models. For the Simple SIR model, we see that the data generally covers the data fairly well but clearly misses the second peak of infection[43]. We also see that the prediction band is very large, covering up a large area of the ternary plot. On the other hand, for the EpiModel network model, we see that the prediction band covers the data quite well and takes up less area.

```
R> both_models <- bind_rows(agg_model, fortified_net)
R>
R>
R> g <- ggplot() + geom_prediction_band(data = both_models %>% filter(t != 0) %>%
+                   mutate(Type = factor(Type, levels = c("Simple SIR",
+                                                         "EpiModel SIR"))),
+          aes(x = X0, y = X1, z = X2,
+              sim_group = sim, fill = Type),
+          alpha = .5,
+          conf_level = .90)
```

[Ben says: In figure 8 I changed the order of the facets given we talk about the simple model first and its more like the "base" model. I think the title should be changed?]

```
R> g +   geom_path(data = both_models %>% filter(t != 0) %>%
+                   mutate(Type = factor(Type, levels = c("Simple SIR",
```

[39] [Ben says: what's the point for this sentence. It also doesn't flow/ connect to previous and later text.]
[40] [Ben says: the phrase "We provide" is very passive / distance from the current demonstration at hand. Moreover section 3 already phrases things this way.]more of sentence 2 of ben than 1.
[41] [Ben says: this isn't a clear phrrase here - what are you trying to say?]
[42] [Ben says: this has never been discussed anyway.]
[43] [Ben says: This could be better movitated with talk of model fit...]

```
+                                                          "EpiModel SIR"))),
+             aes(x = X0, y = X1, z = X2, group = paste(Type, sim)),
+             alpha = .3, col = "gray40") +
+     coord_tern() + theme_sir(base_size = 24) +
+   geom_point(data = hagelloch_sir,
+              aes(x = S, y = I, z =R), col = "black") +
+   labs(title = "Simple SIR model",
+        subtitle = "90% Prediction band and original data",
+        x = "S", y = "I", z = "R") +
+   scale_fill_manual(values = c("#006677", "#AA6600")) +
+   facet_wrap(~Type) +
+   theme(legend.position = "bottom")
```

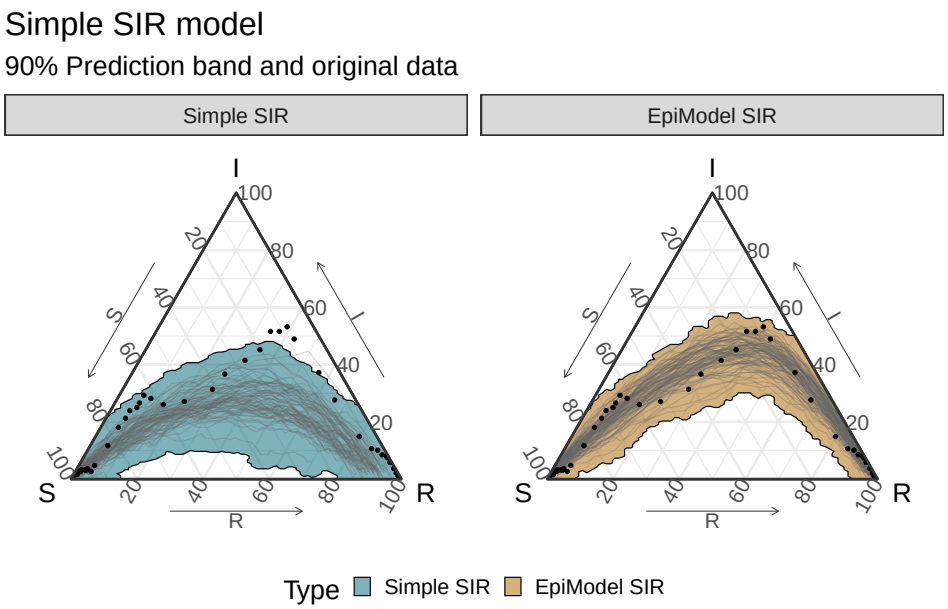

Figure 11: Original Hagelloch SIR data (black) along with 90% prediction band and actual simulation paths from the Simple SIR and the EpiModel SIR models.

However, both models are not a good fit to the filamental path as opposed to the individual points in (S, I, R)-space. This can be[44] captured with the set of simulations both models predict (gray lines), which all generally have a single defined peak of infection whereas the data certainly looks like it has two distinct peaks, likely caused by our assumed super-spreader event. This observation is backed up[45] by the below analysis that demonstrates that the estimated pseudo-density of the observed epidemic (relative to the simulations from either model) is much less likely then **any** of the simulations (reported in Table 6)[46] In conclusion, **EpiCompare** makes it clear that, at a glance, 1) the EpiModel network model is a better fit than the Simple SIR model, and 2) the fit is only good at the ~~geometric filamental level as~~

[44] [Ben says: this is a very passive way to say such things. Try being more direct.]
[45] [Ben says: describe this?]
[46] Ben, do we want to add another sentence or two explaining the two columns in the table? The second one I think makes sense to me but not the first.

~~opposed to the epidemic trajectory filamental level.~~ individual point level as opposed to the geometric filamental level.[47]

```
R> #-- after cleaning up and combining --
R> all_together_df <- rbind(simple_sir,
+                           hagelloch_sir2)
```

Table 5: Top and bottom 2 rows of `all_together_df`, combining both simulated epidemics and the true epidemic.

Type	sim	t	S	I	R
Simple SIR	1	0	188	0	0
Simple SIR	1	1	187	1	0
true observation	0	54	1	0	187
true observation	0	55	1	0	187

```
R> compression_df <- all_together_df %>% group_by(Type, sim) %>%
+   filament_compression(data_columns = c("S","I","R"),
+                        number_points = 20)

R> tdmat <- compression_df %>%
+   dist_matrix_innersq_direction(
+     position = c(1:length(compression_df))[
+       names(compression_df) %in% c("S","I", "R")],
+     tdm_out = T)
R>
R> simple_sir_true_obs_info <- tdmat %>%
+   compare_new_to_rest_via_distance(
+     new_name_id = data.frame(Type = "true observation", sim = 0),
+     distance_func = distance_psuedo_density_function,
+     sigma = "20%")
```

Table 6: The extremeness of the true simulations based on comparing pseudo-density estimates between true vs simulated curves

Type	simulations-based estimated pseudo-density	proportion of simulations with lower estimated pseudo-density
Simple SIR	0.0036733	0.00
EpiModel SIR	0.0149686	0.02

[47] [Ben says: how would this look with the time plots? Do we add value here?]

[48]Overall, **EpiCompare** aids in the data analysis pipeline for both novice and expert practitioners and coders alike. These tools encourage model and simulation exploration of many of the existing and well-supported packages that already exist, and side-by-side comparison thereof. Finally, we hope that practicioners will consider using time-invariant analysis when trying to assess and compare epidemics and epidemic models.

A. Appendix

A.1 Proof of Theorem 1

Proof. Harko *et al.* (2014) provide an analytical solution for the Kermack and McKendrick equations (Eq. (1)) by reparameterizing the ODEs so that $\mathcal{S}(u) = S(t)$, $\mathcal{I}(u) = S(t)$, and $\mathcal{R}(u) = R(t)$ for $0 < u_T < 1$ with

$$\begin{aligned} \mathcal{S}(u) &= S(0)u \\ \mathcal{I}(u) &= N - R(0) + NR_0^{-1}\log u - S(0)u \\ \mathcal{R}(u) &= R(0) - NR_0^{-1}\log u, \end{aligned} \tag{2}$$

and u and t are related by the following integral,

$$\begin{aligned} t &= \int_u^1 \frac{N}{\beta\tau(N - R(0) + R_0^{-1}\log\tau - S(0)\tau)}d\tau \\ &= \int_u^1 \frac{1}{\beta f(S(0), R(0), N, R_0, \tau)}d\tau \\ &= \int_u^1 \frac{1}{\beta f(\tau)}d\tau, \end{aligned}$$

where we have made the denominator of the integral a function of N, the initial values, R_0, and τ, which we further condense to $f(\tau)$ for brevity. Then for a given t we want to find s such that $(S_1(t), I_1(t), R_1(t)) = (S_2(s), I_2(s), R_2(s))$. Or equivalently, for a fixed u want to find v such that $\mathcal{S}_1(u) = \mathcal{S}_2(v)$ and then the corresponding t and s are given by

$$\begin{aligned} t &= \int_u^1 \frac{1}{\beta_1 f(\tau)}d\tau \\ s &= \int_v^1 \frac{1}{\beta_2 f(\tau)}d\tau. \end{aligned}$$

Note that since the equations in Eq. (2) are functions of the initial values and R_0, then $u = v$.

[48]I think this paragraph captures some good goals, but I don't think we've done some of this. For example - we don't really highlight novice/expert usage, and we don't highlight side-by-side comparisons of models.

We then can find a relation for s,

$$\begin{aligned} s &= \int_u^1 \frac{1}{\beta_2 f(\tau)} d\tau \\ &= \int_u^1 \frac{1}{a\beta_1 f(\tau)} d\tau \\ &= \frac{1}{a} \int_u^1 \frac{1}{\beta_1 f(\tau)} d\tau \\ &= \frac{1}{a} t. \end{aligned}$$

□

References

Anderson RM, May RM (1992). *Infectious diseases of humans: dynamics and control*. Oxford university press.

Becker AD, Birger RB, Teillant A, Gastanaduy PA, Wallace GS, Grenfell BT (2016). "Estimating enhanced prevaccination measles transmission hotspots in the context of cross-scale dynamics." *Proceedings of the National Academy of Sciences*, **113**(51), 14595–14600.

Biggerstaff M, Alper D, Dredze M, Fox S, Fung ICH, Hickmann KS, Lewis B, Rosenfeld R, Shaman J, Tsou MH, Velardi P, Vespignani A, Finelli L, Chandra P, Kaup H, Krishnan R, Madhavan S, Markar A, Pashley B, Paul M, Meyers LA, Eggo R, Henderson J, Ramakrishnan A, Scott J, Singh B, Srinivasan R, Bakach I, Hao Y, Schaible BJ, Sexton JK, Del Valle SY, Deshpande A, Fairchild G, Generous N, Priedhorsky R, Hickman KS, Hyman JM, Brooks L, Farrow D, Hyun S, Tibshirani RJ, Yang W, Allen C, Aslam A, Nagel A, Stilo G, Basagni S, Zhang Q, Perra N, Chakraborty P, Butler P, Khadivi P, Ramakrishnan N, Chen J, Barrett C, Bisset K, Eubank S, Anil Kumar VS, Laskowski K, Lum K, Marathe M, Aman S, Brownstein JS, Goldstein E, Lipsitch M, Mekaru SR, Nsoesie EO, Gesualdo F, Tozzi AE, Broniatowski D, Karspeck A, Tse ZTH, Ying Y, Gambhir M, Scarpino S (2016). "Results from the centers for disease control and prevention's predict the 2013-2014 Influenza Season Challenge." *BMC Infectious Diseases*, **16**(1), 1–10. ISSN 14712334. doi:10.1186/s12879-016-1669-x. URL http://dx.doi.org/10.1186/s12879-016-1669-x.

Britton T, Kypraios T, O'Neill PD (2011). "Inference for epidemics with three levels of mixing: methodology and application to a measles outbreak." *Scandinavian Journal of Statistics*, **38**(3), 578–599.

CDC (2021). "CDC COVID Data Tracker." URL https://covid.cdc.gov/covid-data-tracker/#cases_casesper100klast7days.

Ciollaro M, Genovese CR, Wang D (2016). "Nonparametric clustering of functional data using pseudo-densities." *Electronic Journal of Statistics*, **10**(2), 2922–2972. ISSN 19357524. doi:10.1214/16-EJS1198.

Dong E, Du H, Gardner L (2020). "An interactive web-based dashboard to track COVID-19 in real time." *The Lancet infectious diseases*, **20**(5), 533–534.

Ferguson N, Laydon D, Nedjati Gilani G, Imai N, Ainslie K, Baguelin M, Bhatia S, Boonyasiri A, Cucunuba Perez Z, Cuomo-Dannenburg G, *et al.* (2020). "Report 9: Impact of non-pharmaceutical interventions (NPIs) to reduce COVID19 mortality and healthcare demand."

Gallagher S, Chang A, Eddy WF (2020). "Exploring the nuances of R0: Eight estimates and application to 2009 pandemic influenza." *arXiv preprint arXiv:2003.10442*.

Geenens G, Nieto-Reyes A (2017). "On the functional distance-based depth."

Groendyke C, Welch D, Hunter DR (2012). "A network-based analysis of the 1861 Hagelloch measles data." *Biometrics*, **68**(3), 755–765.

Hamilton NE, Ferry M (2018). "ggtern: Ternary Diagrams Using ggplot2." *Journal of Statistical Software, Code Snippets*, **87**(3), 1–17. doi:10.18637/jss.v087.c03.

Harko T, Lobo FS, Mak MK (2014). "Exact analytical solutions of the Susceptible-Infected-Recovered (SIR) epidemic model and of the SIR model with equal death and birth rates." *Applied Mathematics and Computation*, **236**, 184–194. ISSN 00963003. doi:10.1016/j.amc.2014.03.030. 1403.2160, URL http://dx.doi.org/10.1016/j.amc.2014.03.030.

Hethcote HW (2000). "The Mathematics of Infectious Diseases." *SIAM Review*, **42**(4), 599–653. ISSN 00361445. URL http://www.jstor.org/stable/2653135.

Jenness SM, Goodreau SM, Morris M (2018). "EpiModel: An R Package for Mathematical Modeling of Infectious Disease over Networks." *Journal of Statistical Software*. doi:10.18637/jss.v084.i08.EpiModel.

Kermack WO, McKendrick AG (1927). "A contribution to the mathematical theory of epidemics." *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character*, **115**(772), 700–721.

King AA, Nguyen D, Ionides EL (2016). "Statistical inference for partially observed markov processes via the R package pomp." *Journal of Statistical Software*, **69**(12), 1–43. ISSN 15487660. doi:10.18637/jss.v069.i12. 1509.00503.

Meyer S, Held L, Höhle M (2017). "Spatio-Temporal Analysis of Epidemic Phenomena Using the R Package surveillance." *Journal of Statistical Software*, **77**(11), 1–55. doi:10.18637/jss.v077.i11.

MIDAS Network (2021). "Online Portal for COVID-19 Modeling and Research." URL https://midasnetwork.us/covid-19/.

Neal PJ, Roberts GO (2004). "Statistical inference and model selection for the 1861 Hagelloch measles epidemic." *Biostatistics*, **5**(2), 249–261. ISSN 14654644. doi:10.1093/biostatistics/5.2.249.

Oesterle H (1992). "Statistische Reanalyse einer Masernepidemie 1861 in Hagelloch."

Pfeilsticker A (1863). "Beiträge zur Pathologie der Masern mit besonderer Berücksichtigung der statistischen Verhältnisse." URL http://www.archive.org/details/beitrgezurpatho00pfeigoog.

Rvachev LA, Longini IM (1985). "A mathematical model for the global spread of influenza." *Mathematical Biosciences*, **75**(1), 3 – 22. ISSN 0025-5564. doi:http://dx.doi.org/10.1016/0025-5564(85)90064-1. URL http://www.sciencedirect.com/science/article/pii/0025556485900641.

The Washington Post (2021). "Coronavirus US Cases and." URL https://washingtonpost.com/graphics/2020/national/coronavirus-us-cases-deaths/.

Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN 978-3-319-24277-4. URL https://ggplot2.tidyverse.org.

Wickham H, Averick M, Bryan J, Chang W, McGowan LD, François R, Grolemund G, Hayes A, Henry L, Hester J, Kuhn M, Pedersen TL, Miller E, Bache SM, Müller K, Ooms J, Robinson D, Seidel DP, Spinu V, Takahashi K, Vaughan D, Wilke C, Woo K, Yutani H (2019). "Welcome to the tidyverse." *Journal of Open Source Software*, **4**(43), 1686. doi:10.21105/joss.01686.

Worby CJ, Chaves SS, Wallinga J, Lipsitch M, Finelli L, Goldstein E (2015). "On the relative role of different age groups in influenza epidemics." *Epidemics*, **13**, 10–16.

Affiliation:

Shannon K. Gallagher
Biostatistics Research Branch
National Institute of Allergy
and Infectious Diseases
5603 Fishers Lane
Rockville, MD 20852
E-mail: shannon.gallagher@nih.gov
URL: http://skgallagher.github.io

Benjamin LeRoy
Dept. of Statistics & Data Science
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
E-mail: bpleroy@andrew.cmu.edu
URL: https://benjaminleroy.github.io/

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
MMMMMM YYYY, Volume VV, Issue II *Submitted:* yyyy-mm-dd
doi:10.18637/jss.v000.i00 *Accepted:* yyyy-mm-dd