# workshop vignette

## 2023-01-12

This vignette summarises the findings from the 100 days and 100 lines of code workshop, hosted in December 2022 by Epiverse-TRACE.

This document is a draft, the final version will be published on Epiverse's blog after it has been reviewed by other Epiverse members and workshop participants

# What should the first 100 lines of code written during an epidemic look like?

To answer this question, we invited 40 experts, including academics, field epidemiologists, and software engineers, to take part in a 3-day workshop, where they discussed the current challenges, and potential solutions, in data analytic pipelines used to analyse epidemic data.

# What R packages and tools are available to use during an epidemic?

To investigate this in a similar setting to what an outbreak response team would experience, workshop participants were divided into groups, and asked to develop a plausible epidemic scenario, that included:

- A situation report, describing the characteristics of the epidemic
- A linelist of cases and contact tracing data, by modifying provided datasets containing simulated data
- A set of questions to address during the analytic process

Groups then exchanged epidemic scenarios and analysed the provided data to answer the questions indicated the previous group, as if they were a response team working to solve an outbreak. Details about each of these outbreak scenarios and the analytic pipelines developed by the groups are summarised in this vignette.

# Simulating epidemic data

Before the workshop, a fictitious dataset was created, which consisted of a linelist and contact tracing information.

To generate linelist data, the package bpmodels was used to generate a branching process network. Cases were then transformed from the model output to a linelist format. To add plausible hospitalisations and deaths, delay distributions for SARS-CoV were extracted from epiparameter.

To create the contact tracing database, a random number of contacts was generated for each of the cases included in the linelist. These contacts were then assigned a category of became case, under follow up or lost to follow up, at random.

• Through this workshop, we identified the need for a tool to simulate outbreak data in a linelist format, to test analysis methods and other packages while having control over the characteristics of the test data. For this purpose, an R package is currently in progress, see simulist.

# Scenario 1: Novel respiratory disease in The Gambia

#### Situation report

Location: Banjul, The Gambia

Pathogen: novel respiratory

Disease

**Background**: index case travelled to London in Dec 2022. 85 symptomatic individuals identified in February 2023 are under investigation

#### Modified data

#### Linelist:

- + travel history column
- + imported case y/n column
- + id column
- > Added missing cases
- > Introduced errors

#### Contact database:

- > Added missing cases
- > Introduced errors

#### Questions

- 1. What are the most affected groups?
- 2. What are the epidemiological characteristics of infection?
- 3. What is the level of individual variation of transmission?
- 4. How is the outbreak likely to develop over the next weeks?
- 5. What public health measures are required to contain it?

# Analytic pipeline for scenario 1 (analysed by group 2)

- · Data cleaning
  - linelist to standardise date format
  - cleanr from previous Hackathon
- Delay distributions
  - fitdisrplus to fit parameteric distributions to scenario data
  - epiparameter to extract delay distributions from respiratory pathogens
  - EpiNow2 to fit reporting delays
  - EpiEstim / coarseDataTools to estimate incubation period of disease
  - epicontacts
  - mixdiff
- Population demographics
  - Would like to have had access to an R package similar to ColOpenData
- Risk factors of infection
  - Used R4epis as a guide on how to create two-way tables and perform Chi-squared tests
- Severity of disease
  - datadelay for CFR calculation
  - Implementation of method developed by AC Ghani, 2005 to estimate CFR
- Contact matching
  - diyar to match and link records
  - fuzzyjoin to join contact and case data despite misspellings or missing cell contents
- Epi curve and maps
  - Used incidence and incidence for incidence calculation and visualisation
  - raster to extract spatial information from library of shapefiles
- Reproduction number

- APEestim
- bayEStim
- earlyR
- epicontacts
- epidemia
- epiFilter
- EpiNow2
- EpiEstim
- R0
- outbreaker2
- Used this comparison table to choose the most appropriate package.
- Superspreading, by using these resources:
  - fitdistrplus
  - epicontacts
- Epidemic projections
  - incidence R estimation using a loglinear model
  - projections using Rt estimates, SI distributions and overdispersion estimates
- Transmission chains and strain characterisation
  - IQtree and nextclade to build a maximum likelihood tree and mannually inspect it
  - Advanced modelling through phylodynamic methods, using tools like BEAST

Data analysis step	Challenges
Data cleaning	Not knowing what packages are available for this purpose
Delay distributions	Dealing with right censoring Accounting for multiple infectors
Population demographics	Lacking tools that provide information about population by age, gender, etc.
Risk factors of infection	Distinguishing between risk factors vs detecting differences in reporting
	frequencies among groups
Severity of disease	Knowing the prevalence of disease (denominator) Right truncated data
	Varying severity of different strains
Contact matching	Missing data Misspellings
Epicurve and maps	NA dates entries not included Reporting levels varying over time
Offspring distribution	Right censoring Time varying reporting efforts Assumption of a single
-	homogeneous epidemic Importation of cases
Forecasting	Underlying assumption of a given R distribution, e.g., single trend,
	homogeneous mixing, no saturation

# Scenario 2: Outbreak of an unidentified disease in rural Colombia

#### Situation report

Location: Ibague, Colombia

Pathogen: unspecified

**Background:** suspected link to a zoonotic reservoir, since early cases were farmers. Envoy is colourblind and unable to deal with complex visual information

#### **Modified data**

#### Linelist:

- + travel to rural area column
- + case occupation column
- + notification date column
- + notification delay column
- > Added missing cases
- > Introduced errors

Contact database: no changes

#### Questions

- 1. Have there been exposures to a zoonotic reservoir?
- 2. How transmissible is the disease?
- 3. What are the hospital bed requirements for next weeks?
- 4. What is the attack rate and the no. of deaths we expect?
- 5. Can you reconstruct transmission chains?

# Analytic pipeline for scenario 2 (analysed by group 3)

- Data cleaning: manually, using R (no packages specified), to
  - Fix data entry issues in columns onset date and gender
  - Check for missing data
  - Check sequence of dates: symptom onset  $\rightarrow$  hospitalisation  $\rightarrow$  death
- Data anonymisation to share with partners
  - fastlink for probabilistic matching between cases contacts, based on names, dates, and ages
- Case demographics
  - apyramid to stratify data by age, gender, and health status
- Reproductive number calculation, by using two approaches:
  - Manually, by calculating the number of cases generated by each source case, data management through dplyr and data.table
  - Using serial interval of disease, through EpiEstim or EpiNow2
- Severity of disease
  - Manual calculation of CFR and hospitalisation ratio
- Projection of hospital bed requirements
  - EpiNow2 to calculate average hospitalisation duration and forecasting
- Zoonotic transmission of disease
  - Manual inspection of cases' occupation
  - Use of IQtree and ggtree to plot phylogenetic data
- Superspreading
  - epicontacts
- Calculation of attack rate
  - Unable to calculate, given the lack of seroprevalence data

Data analysis step	Challenges
Data anonymisation	Dealing with typos and missing data when generating random unique identifiers
Reproduction number	Right censoring Underestimation of cases due to reporting delays
Projection of hospital bed requirements	Incomplete data (missing discharge date) Undocumented functionality in R packages used
Zoonotic transmission	Poor documentation Unavailability of packages in R Differentiation between zoonotic transmission and risk factors- need for population data
Attack rate	Not enough information provided

# Scenario 3: Reston Ebolavirus in the Philippines

#### Situation report

**Location**: Calabarzon, Philippines

**Pathogen**: Reston Ebolavirus (RESTV)

**Background**: first ever recorded case of symptomatic RESTV, in January 2023. More potential cases found: patients with haemorragic fever

#### **Modified data**

#### Linelist:

- + geographic region column
- + case occupation column
- > Added missing cases
- > Introduced errors

#### Contact database:

- + risk category column
- Added missing casesIntroduced errors

- 1. What is the CFR?
- 2. Can you estimate the % of unobserved cases?
- 3. When did the outbreak start?

Questions

- 4. What are the forecasted no. of cases over next weeks?
- 5. How many spillover events?
- 6. Is contact tracing effective?
- 7. How good is the risk classification system?

# Analytic pipeline for scenario 3 (analysed by group 4)

- Data cleaning
  - Importing data with rio, readxl, readr, or openxlsx
  - Rename variables with janitor
  - Initial data checks with pointblank, assertr, compareDF, or skimr
  - Vertical data checks with matchmaker, lubridate, or parsedate
  - Horizontal data checks with hmatch, assertr, or queryR
  - Detect duplicates with janitor and tidyverse
  - Checking for consistency with dplyr, or powerjoin
  - Translation with matchmaker
- Delay distributions
  - fitdistrplus to fit parameteric distributions to epidemic data
- Case demographics
  - apyramid to stratify data by age, gender, and health status
  - ggplot2 to visualise data
- Outbreak description
  - sitrep to generate reports
- Visualisation of geographic data
  - sf for static maps
  - leaflet for interactive maps

- Generation of tables
  - gtsummary for static tables
  - janitor for interactive tables
- Severity of disease
  - EpiNow2 and survival to calculate CFR
- Attack rate
  - gadm function to get population data
  - epitabulate to describe data
  - sf and ggplot2 to plot data
- Forecasting
  - EpiEstim
  - EpiNow2
  - bpmodels
- Spillover events
  - By cross-referencing contact data with occupations
- Effectiveness of contact tracing
  - By calculating the proportion of case follow-ups and comparing the delay of disease exposure to the follow-up delay
- Transmission trees
  - epicontacts
  - ggplot2

Data analysis step	Challenges
Detection of outliers	No known tools to use
Severity of disease	Censoring
Spillover events	Missing data

# Scenario 4: Emerging avian influenza in Cambodia

# Situation report

**Location**: Phnom Penh, Cambodia

**Pathogen**: emerging avian influenza

**Background**: outbreak started in a small rural community of poultry farmers in Cambodia. Cases now also reported in the capital

### **Modified data**

### Linelist:

- + outcome column
- + origin column
- + exposure column
- > Added missing cases
- > Introduced errors

Contact database: no changes

### Questions

- 1. What are the % of human vs zoonotic transmission?
- 2. What is the epidemic potential?
- 3. Is the outbreak contained by the implemented measures?
- 4. What is the severity of disease?
- 5. Can underreporting be estimated?
- 6. Are more hospital beds needed?
- 7. Should they cull poultry farms?

# Analytic pipeline for scenario 4 (analysed by group 5)

- Data cleaning
  - readxl to import data
  - dplyr to remove names
  - Manually scanning through excel to check for errors
- Reproduction number
  - EpiEstim
- Severity of disease
  - Manually using R to detect missing cases
  - epiR to check for data censoring

Data analysis step	Challenges
Data cleaning Reproduction number Severity	No available R packages specific for epidemic data Difficulty finding parameter estimations in the literature Missing cases Need for an R package for systematic censoring analysis

# Scenario 5: Outbreak of respiratory disease in Canada

#### Situation report

Location: Saskatoon, Canada

**Pathogen**: unknown respiratory pathogen

Background: influenza-like illness (ILI), with severe outcomes, including deaths. Possible link to visits to local zoo, which has seen an increased number of bird deaths

# **Modified data**

## Linelist:

- + test date column
- + report date column
- > Added missing cases
- > Introduced errors

**Contact database:** no changes, as it was not provided to the next group

#### Questions

- 1. What is the demographic distribution of cases?
- 2. What data sources are needed to understand the outbreak?
- 3. What is the severity of disease?
- 4. Can you nowcast cases?
- 5. How many cases are being missed?
- 6. What % of infections are likely zoonoses?

# Analytic pipeline for scenario 5 (analysed by group 1)

- Define project structure
  - Defining the script's structure with cookiecutter, reportfactory, and orderly
  - Ensuring reproducibility of the analysis with iRODS and Git
  - Working in a group with GitHub
- Data cleaning
  - Importing data with readr or rio
  - Checking for errors with linelist, janitor, parsedate, matchmaker, or lubridate
  - janitor to eliminate duplicates
  - naniar to check for missing data
  - epitrix to anonymise data
- Delay distributions

- epitrix
- fitdistrplus to fit parameteric distributions to scenario data
- Case demographics
  - apyramid to stratify data by age, gender, and health status
- Nowcasting
  - incidence2 to visualise incidence from linelist data
  - epiparameter to extract infectious disease parameter data
  - EpiEstim or EpiNow2 for Rt calculation
- Severity of disease
  - Calculation of hospitalisation and mortality rates- no R package specified
- Zoonotic transmission
  - forecast
- Generation of reports
  - incidence for static reports
  - Quarto and R markdown for dashboards

Data analysis step	Challenges
Project structure	Working simultaneously on the same script and managing parallel tasks
	Anticipating future incoming data in early pipeline design
Data cleaning	Large amount of code lines used on (reasonably) predictable cleaning
	(e.g. data sense checks) Omitting too many data entries when simply
	removing NA rows Non standardised data formats Implementing rapid quality
	check reports before analysis
Delay distributions	Identifying the best method to calculate, or compare functionality of tools
v	Need to fit multiple parametric distributions and return best, and store as usable objects
Severity of disease	Censoring and truncation Underestimation of mild cases Need database of age/gender pyramids for comparisons
Forecasts	Need option for fitting with range of plausible pathogen serial intervals and comparing results Changing reporting delays over time Matching
	inputs/outputs between packages
Zoonotic transmisison	Need for specific packages with clear documentation How to compare simple
	trend-based forecasts