

RECON

Evidence synthesis approaches for outbreak detection
and reconstruction

Thibaut Jombart

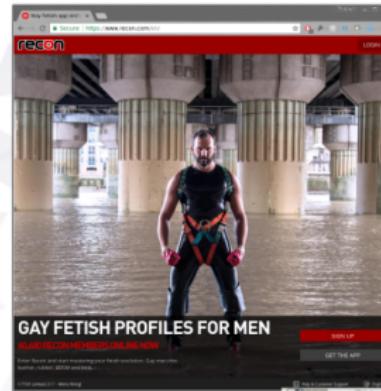
4th October 2017

Imperial College London
MRC Centre for Outbreak Analysis and Modelling

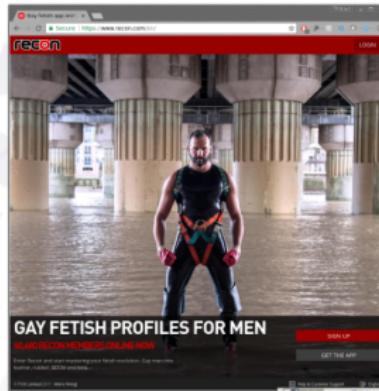
Topics of the day

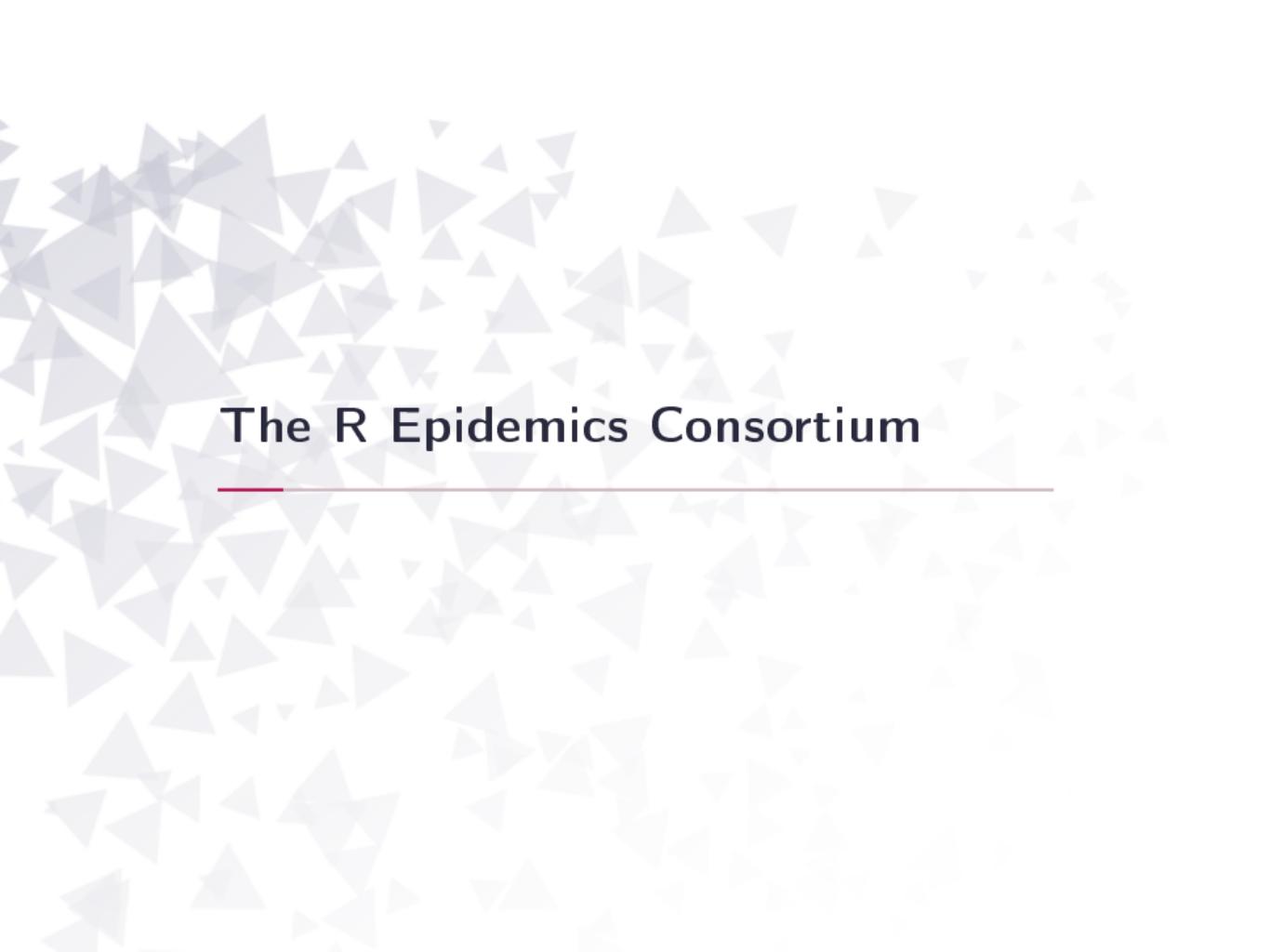


Topics of the day



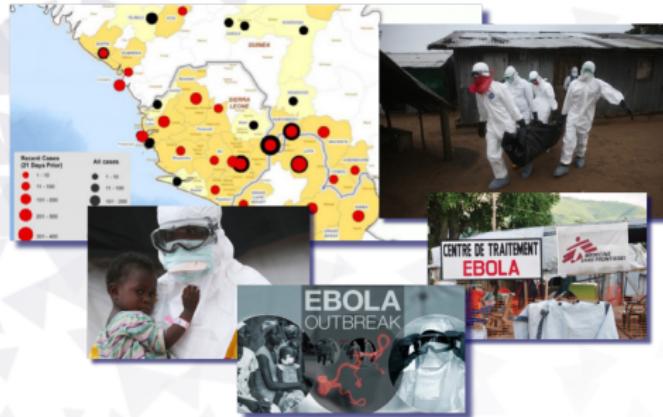
Topics of the day



The background of the slide features a large number of small, light-gray triangles of various sizes scattered across the entire area, creating a subtle geometric pattern.

The R Epidemics Consortium

Lessons learnt from the Ebola response



Lessons learnt from the Ebola response

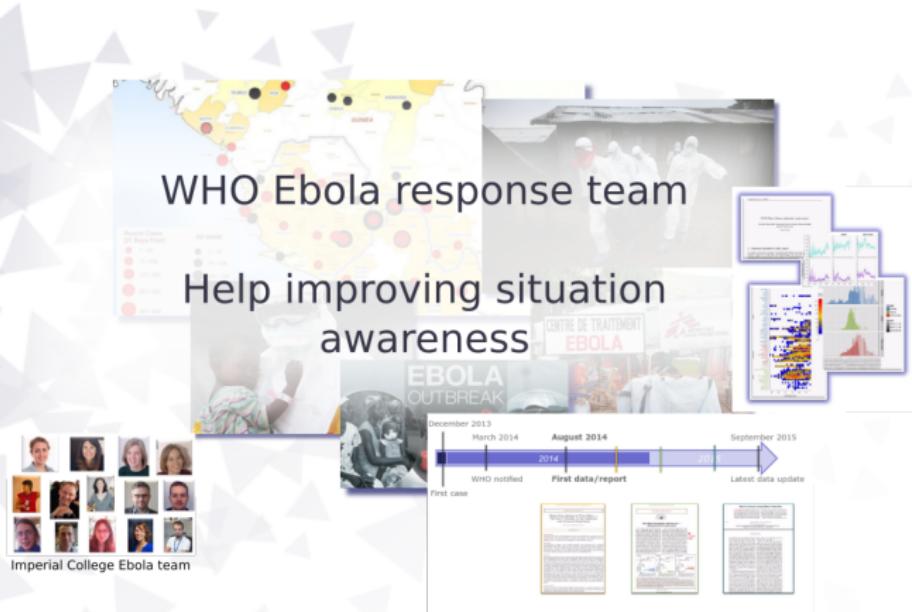


Lessons learnt from the Ebola response

The collage includes:

- A map of West Africa showing the locations of Ebola cases.
- A photograph of the WHO Ebola response team in protective suits.
- A graph titled "WHO Ebola response team" showing data over time.
- A photograph of a person in a protective suit.
- A photograph of a "CENTRE DE TRAITEMENT EBOLA" (Treatment Center).
- A graph titled "EBOLA OUTBREAK" showing trends.
- A timeline from December 2013 to September 2015 with markers for "First case", "WHO notified", "First data/report", and "Latest data update".
- A grid of 16 small portraits labeled "Imperial College Ebola team".
- A section titled "Help improving situation awareness" with a purple border containing various data visualizations.

Lessons learnt from the Ebola response



Most statistical/modelling tools for situation awareness were missing.

Who do we need to develop these tools?



Who do we need to develop these tools?

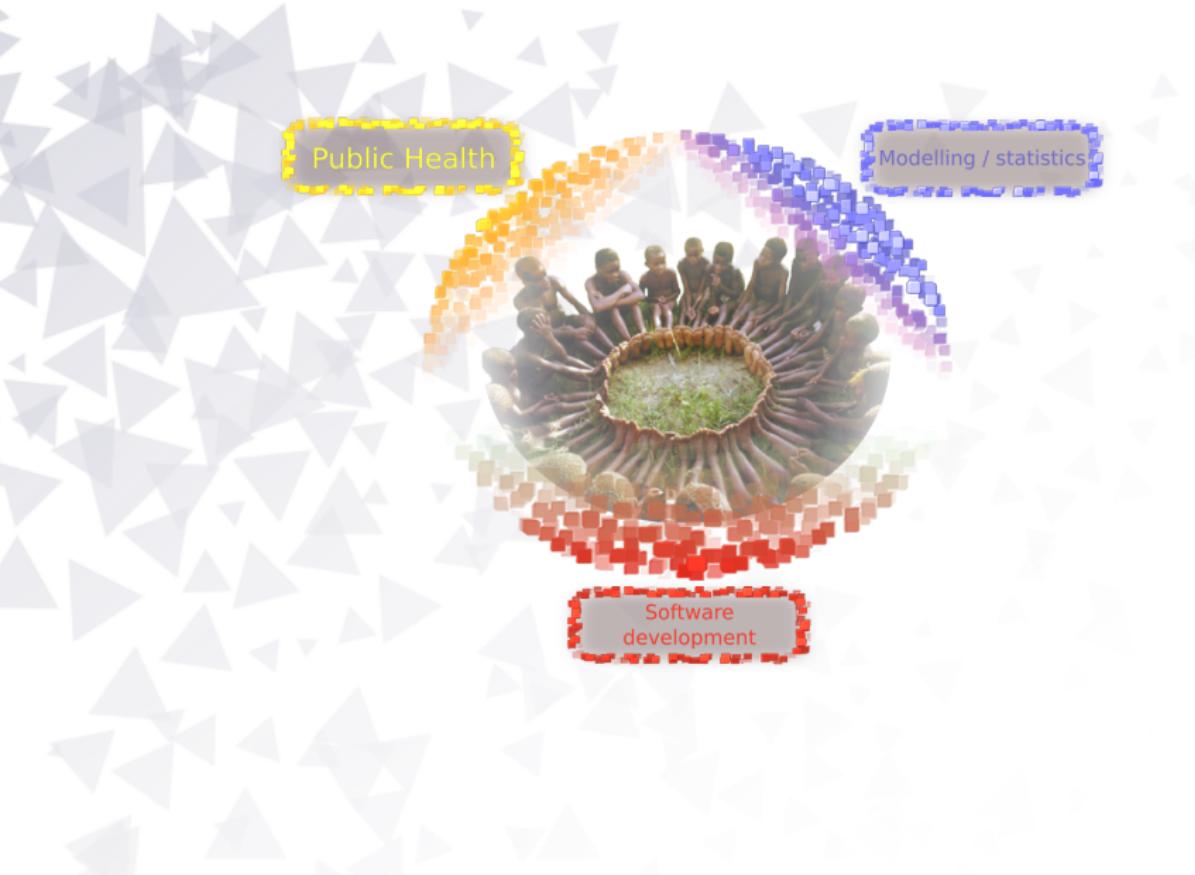
Public Health



Who do we need to develop these tools?



Who do we need to develop these tools?



From a hack to a pack



Hackout 3, summer 2016, Berkeley

From a hack to a pack



Hackout 3, summer 2016, Berkeley



From a hack to a pack



Hackout 3, summer 2016, Berkeley

functional
incubation
userfriendly secure dictionary
systems testing automated continuous
collection series repository
rpp efficiency number fast
secured bias outbreaks
parsing code integration
reporting gui
unit data delay
epidemiology security peak
situation anonymised
opensource contact
epiinfo delay
clean compiled
outbreaker interface tree
symptoms interface
lineelist fellow
tracing shiny
automation cdc
epicontacts edic
ggplot cleaning
dashboard clusters rates
parallel reliable
parameters contacttracing
epidemics genomics
distribution

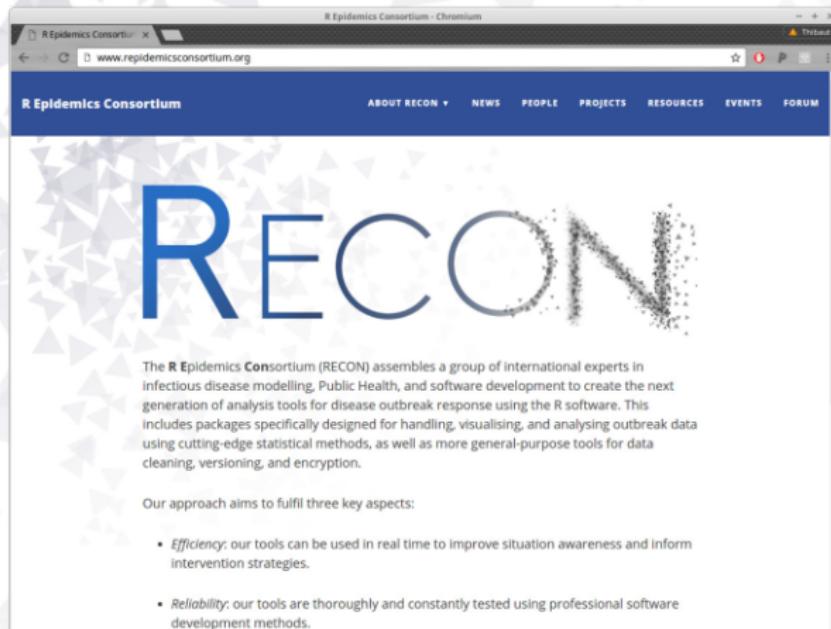
RECON
[The R Epidemics Consortium]

From a hack to a pack



RECON: the R Epidemics Consortium

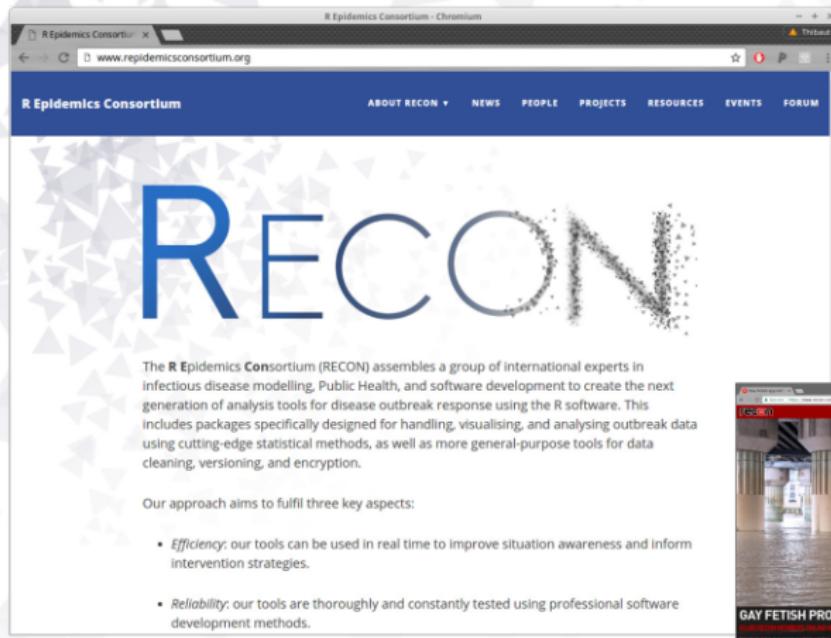
A taskforce to build a new generation of outbreak response tools in .



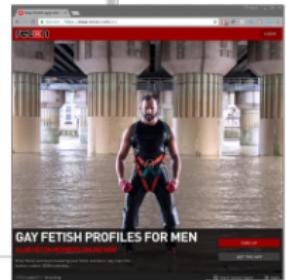
The screenshot shows a web browser window for the "R Epidemics Consortium" website. The address bar displays "www.repidemicsconsortium.org". The page features a large, stylized title "RECON" where the letter "O" is composed of numerous small dots. Below the title, a paragraph explains the consortium's purpose: "The R Epidemics Consortium (RECON) assembles a group of international experts in infectious disease modelling, Public Health, and software development to create the next generation of analysis tools for disease outbreak response using the R software. This includes packages specifically designed for handling, visualising, and analysing outbreak data using cutting-edge statistical methods, as well as more general-purpose tools for data cleaning, versioning, and encryption." A section titled "Our approach aims to fulfil three key aspects:" lists three bullet points: "*Efficiency*: our tools can be used in real time to improve situation awareness and inform intervention strategies.", "*Reliability*: our tools are thoroughly and constantly tested using professional software development methods.", and another bullet point that is partially cut off.

RECON: the R Epidemics Consortium

A taskforce to build a new generation of outbreak response tools in .

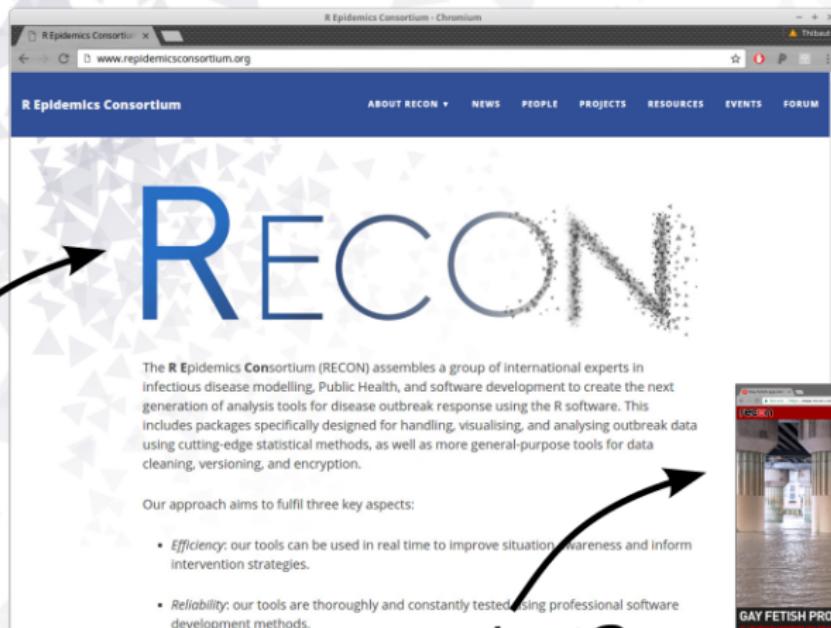


The screenshot shows the homepage of the R Epidemics Consortium (RECON) website. The header features the title "R Epidemics Consortium" and a navigation menu with links for "ABOUT RECON", "NEWS", "PEOPLE", "PROJECTS", "RESOURCES", "EVENTS", and "FORUM". The main visual is a large, stylized word "RECON" where the letters are composed of small dots or particles. Below this, a text block describes the consortium's mission: "The R Epidemics Consortium (RECON) assembles a group of international experts in infectious disease modelling, Public Health, and software development to create the next generation of analysis tools for disease outbreak response using the R software. This includes packages specifically designed for handling, visualising, and analysing outbreak data using cutting-edge statistical methods, as well as more general-purpose tools for data cleaning, versioning, and encryption." A section titled "Our approach aims to fulfil three key aspects:" lists three bullet points: "Efficiency: our tools can be used in real time to improve situation awareness and inform intervention strategies.", "Reliability: our tools are thoroughly and constantly tested using professional software development methods.", and "Scalability: our tools can handle large amounts of data and be scaled up as needed." At the bottom right of the page, there is a small image of a man in athletic gear standing in what appears to be an industrial or construction setting.



RECON: the R Epidemics Consortium

A taskforce to build a new generation of outbreak response tools in  .

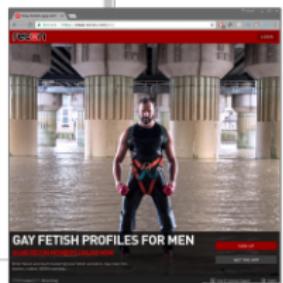


The R Epidemics Consortium (RECON) assembles a group of international experts in infectious disease modelling, Public Health, and software development to create the next generation of analysis tools for disease outbreak response using the R software. This includes packages specifically designed for handling, visualising, and analysing outbreak data using cutting-edge statistical methods, as well as more general-purpose tools for data cleaning, versioning, and encryption.

Our approach aims to fulfil three key aspects:

- *Efficiency*: our tools can be used in real time to improve situation awareness and inform intervention strategies.
- *Reliability*: our tools are thoroughly and constantly tested using professional software development methods.

Not us



RECON

www.repidemicsconsortium.org

- started 6th September 2016
- ~70 members
- 17 countries, > 40 institutions
- ~ 3 packages released, 20 under development
- public forum, blog, online resources



Statistical software development

- **efficiency**: useful for improving situation awareness in real time
- **reliability**: outputs can be trusted
- **accessibility**: widely available, easy learning curve

RECON: what we do



[nicer
'strong and
stable']

Statistical software development

- **efficiency**: useful for improving situation awareness in real time
- **reliability**: outputs can be trusted
- **accessibility**: widely available, easy learning curve

RECON: what we do



[nicer
'strong and
stable']

Statistical software development

- **efficiency**: useful for improving situation awareness in real time
- **reliability**: outputs can be trusted
- **accessibility**: widely available, easy learning curve

Translation

- **disseminating knowledge**: free online training material, involvement with FETPs, workshops
- **outbreak response**: deployment to the field
- **RECON deployer**: portable data analysis environment

RECON: projects

The screenshot shows the 'R Epidemics Consortium' website at www.repidemicsconsortium.org/projects/. The page displays 14 projects, each represented by a circular icon containing gears and a brief description:

- epiflows**: Visualisation and analysis of passenger flows.
- epimaps**: Helpers and wrappers for mapping diseases.
- epimatch**: Finding matching patient records across tabular data sets.
- epitrix**: Small utility functions for epidemiology.
- gisfirstaid**: Tutorials and code gists for mapping infectious diseases.
- incidence.ul**: Graphical user interface for incidence.
- nomad**: Pack up R to take away.
- outbreaker2**: Inferring transmission chains by integrating epidemiological and genetic data.
- projections**: Projections of future incidence.
- recon.ul**: Template shiny GUI for RECON packages.
- recontools**: Tools to develop RECON packages.
- vimes**: Visualisation and Monitoring of Epidemics, including some outbreak detection algorithms.

A red circle highlights the **vimes** project icon.



An integrative approach for outbreak detection



Aims: develop a new method which..

- **detects outbreaks** i.e. groups of related cases (on the same transmission chain)



¹ well, really, I made that up because I was reading 'Snuff' at the time; at least this one is not a dodgy website (yet); incidentally, Terry Pratchett was a huge fan of using long footnotes, which were often quite entertaining to read; note that it does not apply here: if you are still reading this, you probably missed what I just said



Aims: develop a new method which..

- **detects outbreaks** i.e. groups of related cases (on the same transmission chain)
- **integrates different data:** temporal, spatial, genetic, etc.

¹ well, really, I made that up because I was reading 'Snuff' at the time; at least this one is not a dodgy website (yet); incidentally, Terry Pratchett was a huge fan of using long footnotes, which were often quite entertaining to read; note that it does not apply here: if you are still reading this, you probably missed what I just said

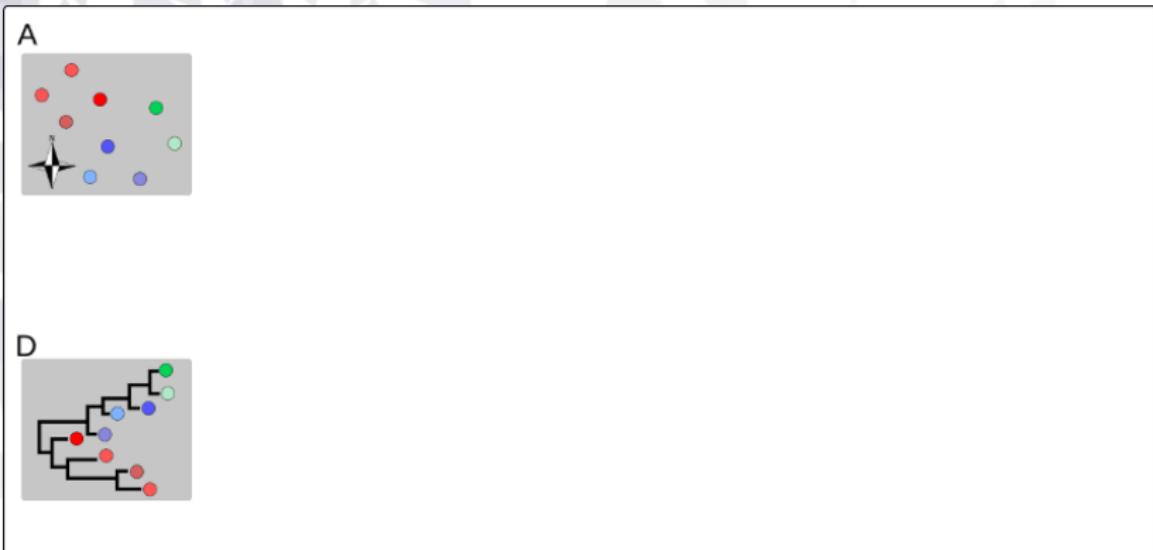


Aims: develop a new method which..

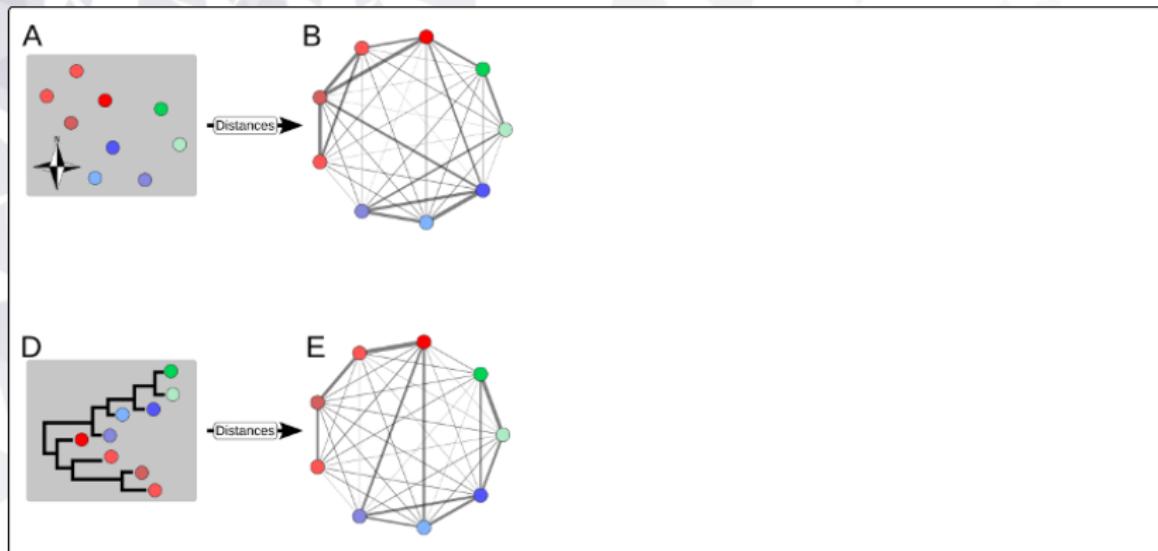
- **detects outbreaks** i.e. groups of related cases (on the same transmission chain)
- **integrates different data:** temporal, spatial, genetic, etc.
- **works fast, scales well:** so that it can be used for real-time outbreak detection

¹ well, really, I made that up because I was reading 'Snuff' at the time; at least this one is not a dodgy website (yet); incidentally, Terry Pratchett was a huge fan of using long footnotes, which were often quite entertaining to read; note that it does not apply here: if you are still reading this, you probably missed what I just said

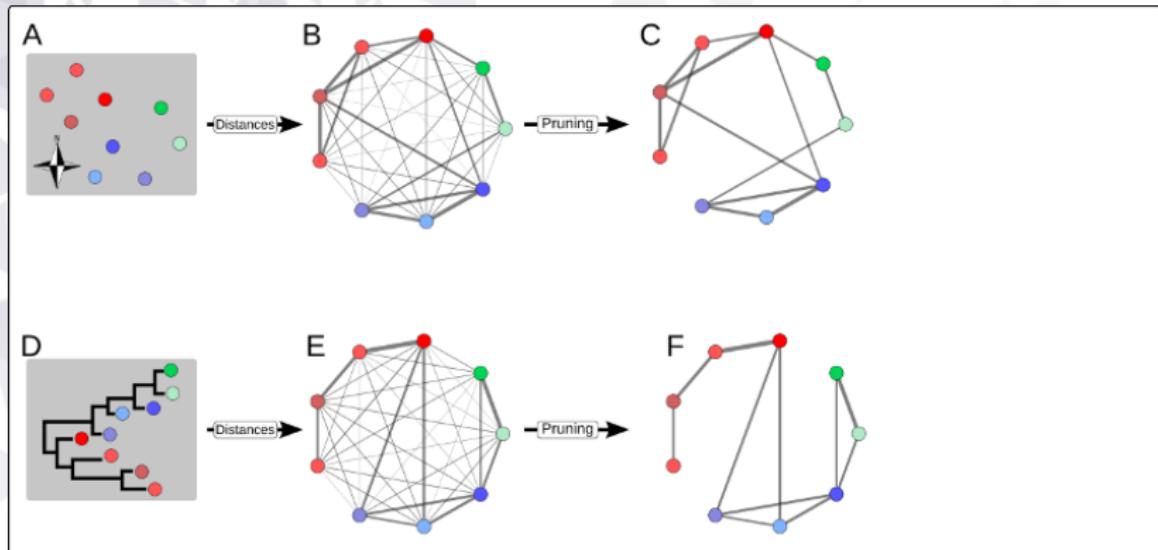
A graph-based evidence synthesis approach



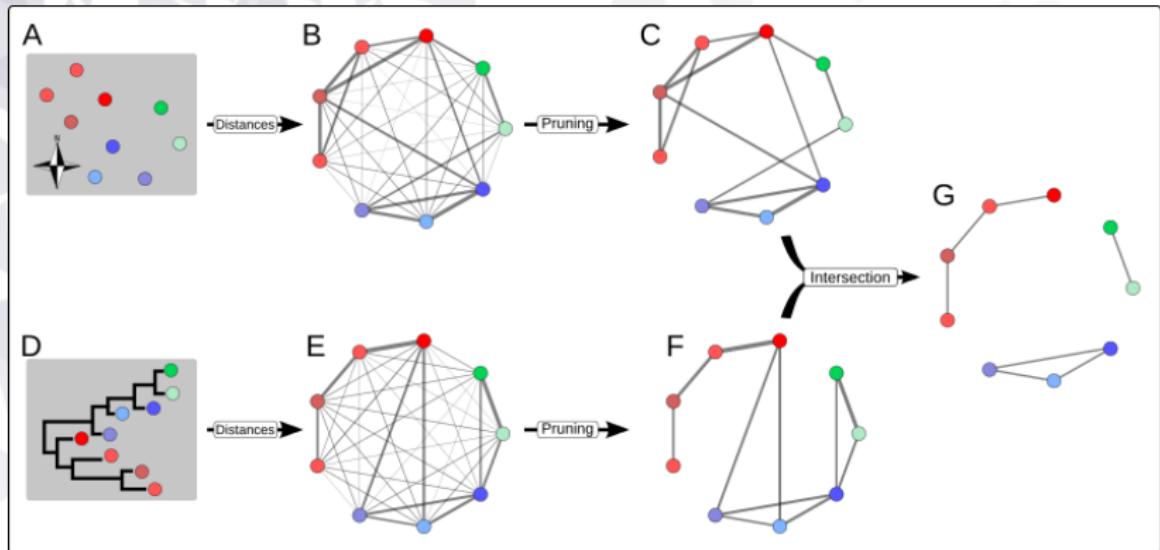
A graph-based evidence synthesis approach



A graph-based evidence synthesis approach

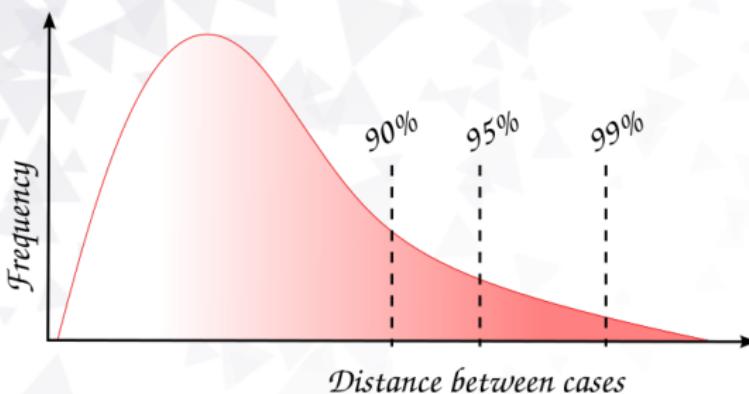


A graph-based evidence synthesis approach

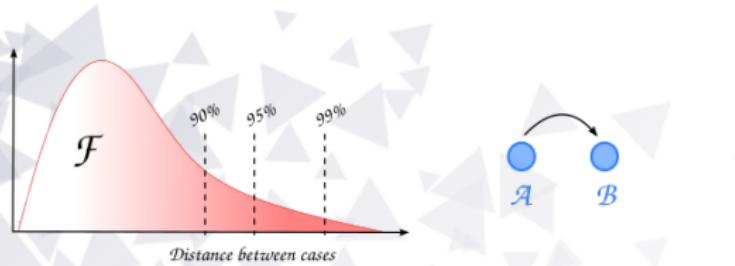


Pruning graphs: where to cut?

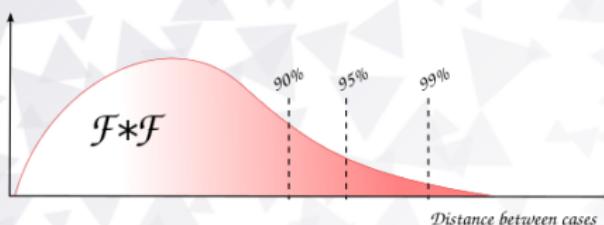
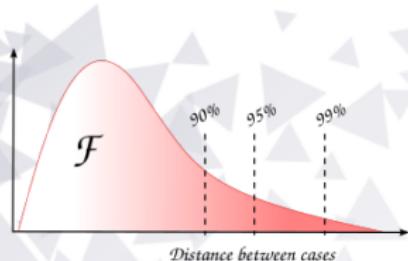
Assuming a known expected distribution between pairs of cases (e.g. serial interval, spatial kernel, molecular clock), different quantiles can be used:



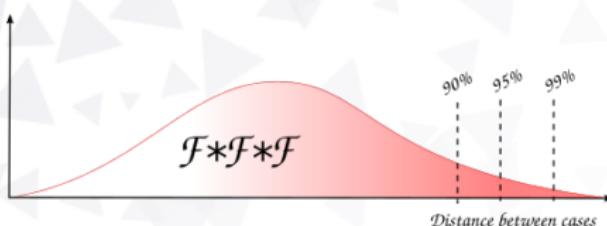
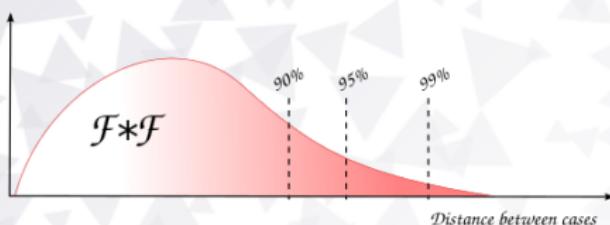
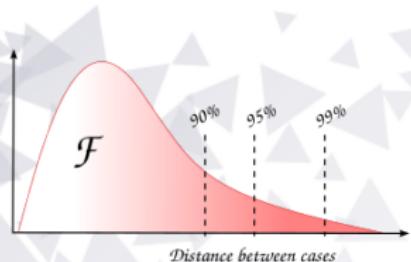
Pruning graphs: where to cut?



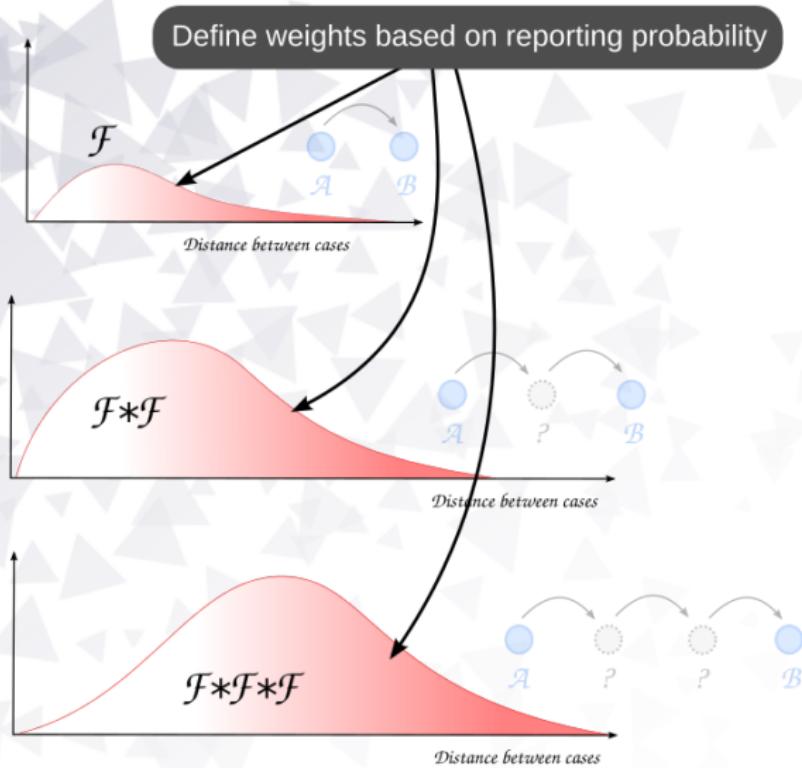
Pruning graphs: where to cut?



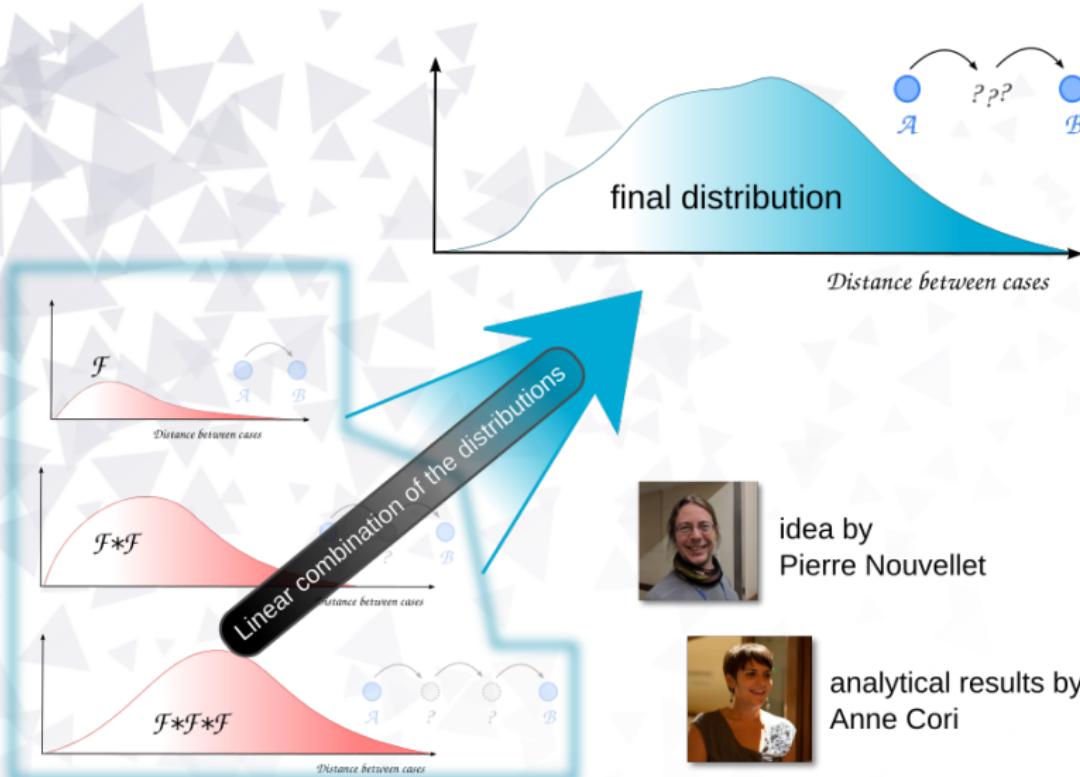
Pruning graphs: where to cut?



Pruning graphs: where to cut?



Pruning graphs: where to cut?



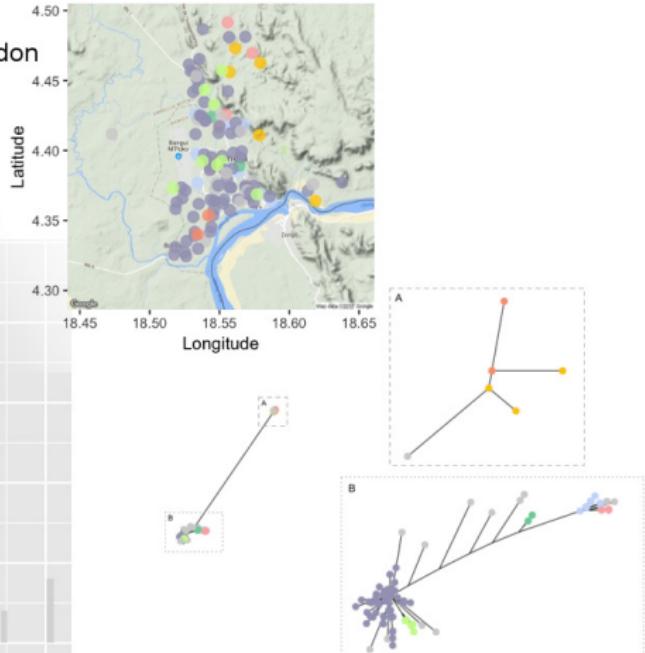
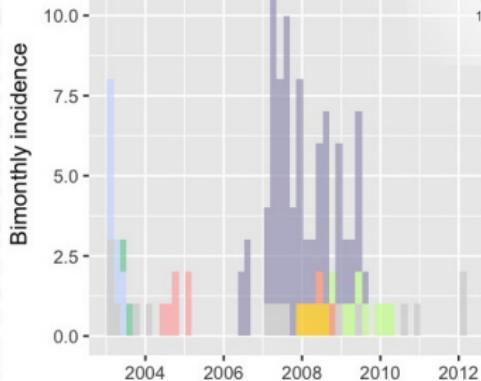
Application: dog rabies epidemics, Central African Republic



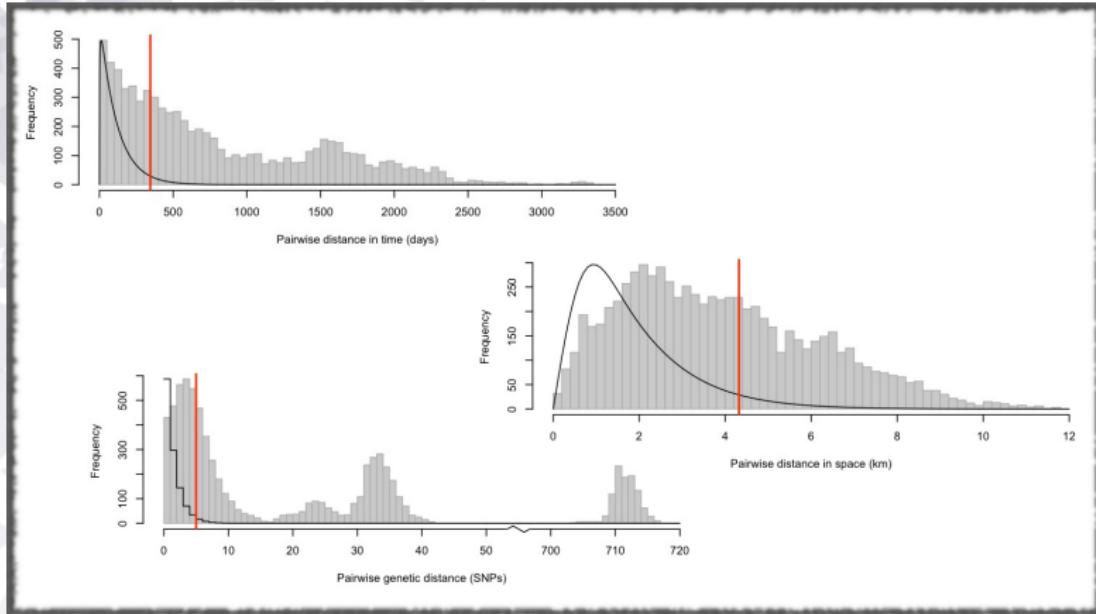
Anne Cori
Imperial College London



Pierre Nouvellet
University of Sussex

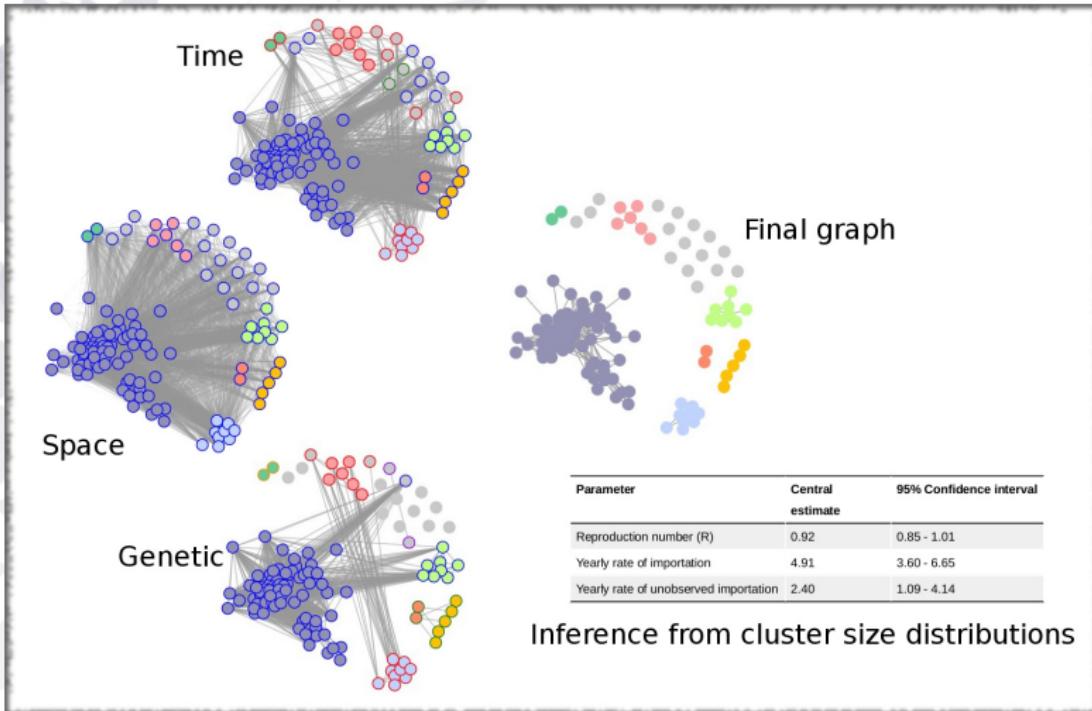


Distributions of distances between cases



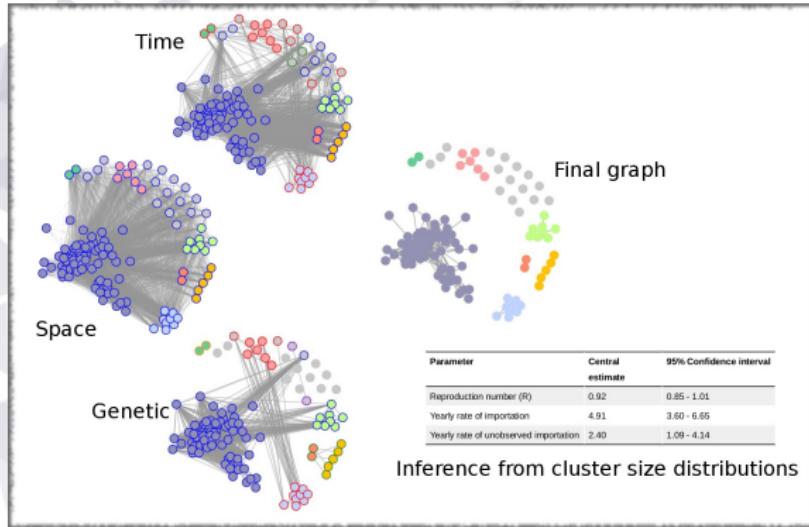
[material by Anne Cori]

Results



[material by Anne Cori and Pierre Nouvellet]

Results



- one large outbreak, low R_0 , $\sim 5 - 10$ introductions / year
- same results as more complex approaches (BEAST, epi model + particle filtering)
- much faster (<1s vs 1 week)

vimes: summary

- **flexible** approach for detecting outbreaks using different data types

- **flexible** approach for detecting outbreaks using different data types
- **threshold**: unsatisfying, but sensitivity study easy

vimes: summary

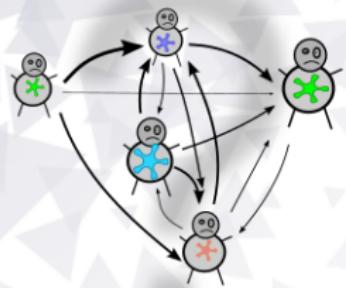
- **flexible** approach for detecting outbreaks using different data types
- **threshold**: unsatisfying, but sensitivity study easy
- **fast and scalable**: possible integration in routine surveillance

- **flexible** approach for detecting outbreaks using different data types
- **threshold**: unsatisfying, but sensitivity study easy
- **fast and scalable**: possible integration in routine surveillance
- can serve as **basis to other methods** for integrating different data sources

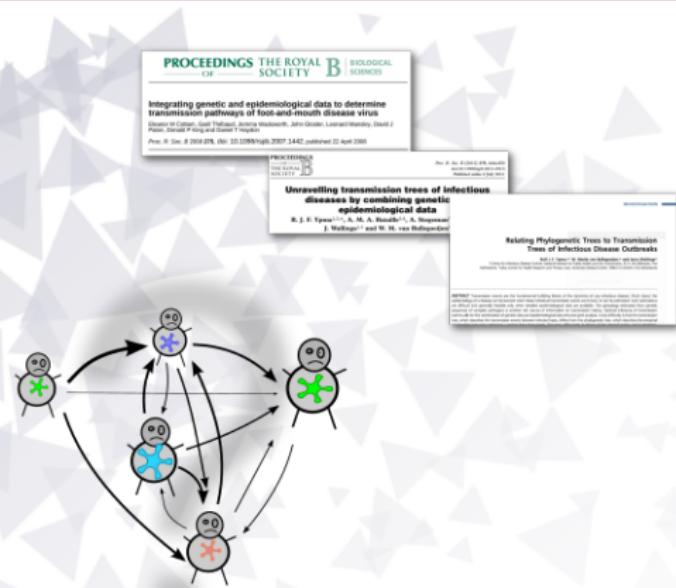


Reconstructing outbreaks from epidemiological and genetic data

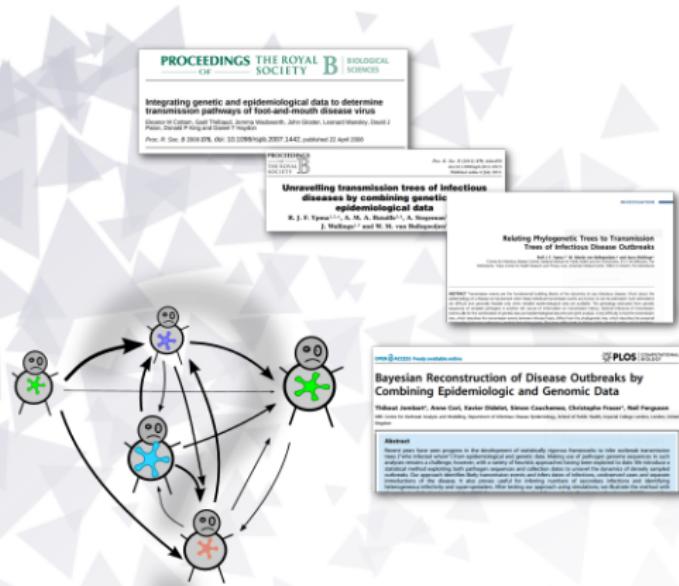
Who infects whom? Many answers for a single question



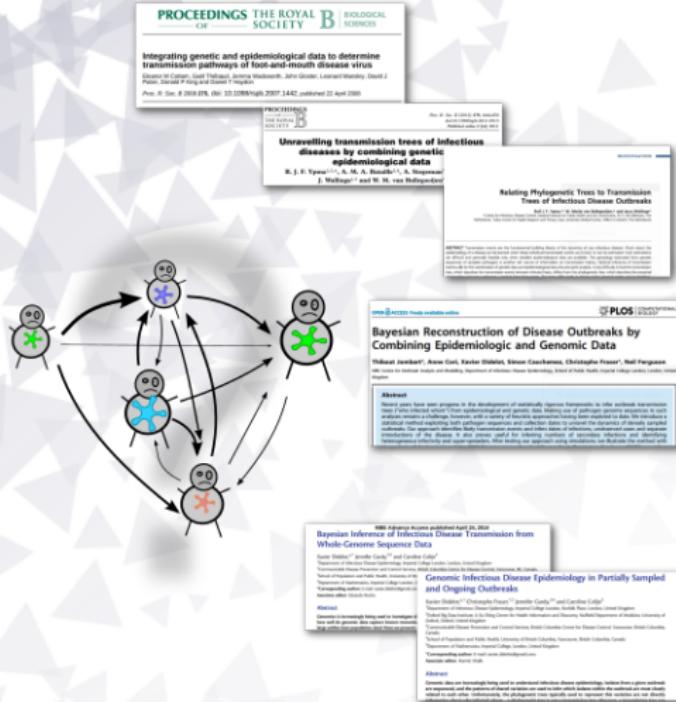
Who infects whom? Many answers for a single question



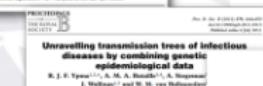
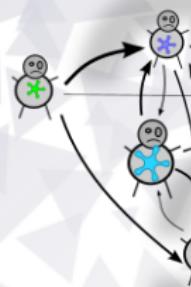
Who infects whom? Many answers for a single question



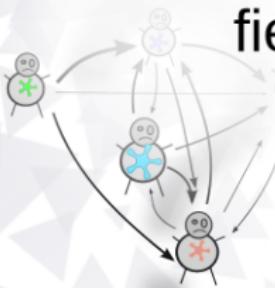
Who infects whom? Many answers for a single question



Who infects whom? Many answers for a single question



Who infects whom? Many answers for a single question



But fast growing methodological fields can get messy!

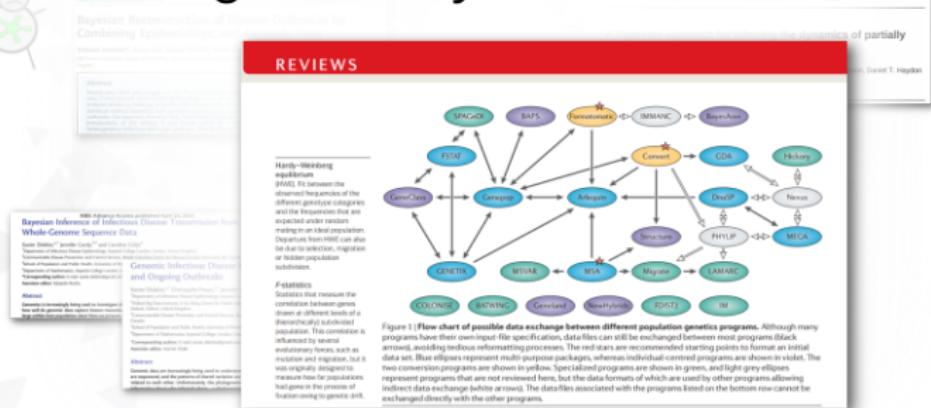


Figure 1 | Flow chart of possible data exchange between different population genetics programs. Although many programs have their own input–file specifications, data files can still be exchanged between most programs (black arrows), avoiding tedious reformatting processes. The red stars are recommended starting points to format an initial data set. Blue ellipses represent multi-purpose packages, while the small oval programs shown in violet, The programs that are not shown are those that are not reviewed here, but the data formats of which are used by other programs allowing indirect data exchange (bulky arrows). The data files associated with the programs listed on the bottom row cannot be exchanged directly with the other programs.

Excoffier and Heckel 2006, Nature Reviews Genetics

Are different methods really... different?



Different models are often structurally similar, and can lead to near identical implementations.

Are different methods really... different?



Different models are often structurally similar, and can lead to near identical implementations.

- a, b, c : different types of data
- θ : parameters / augmented data

Likelihood components are often treated as *conditionally independent modules*:

$$p(a, b, c|\theta) = p(a|\theta)p(b|\theta)p(c|\theta)$$

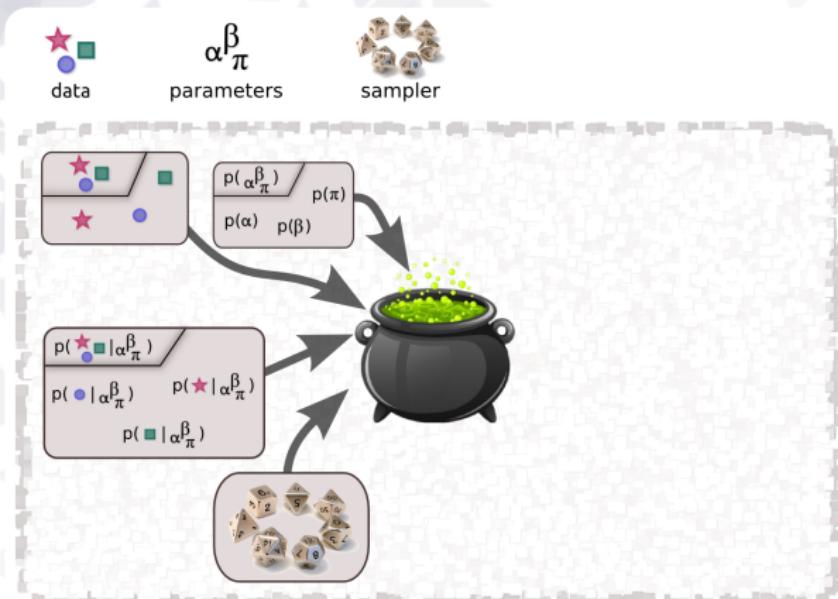
outbreaker2: a general tool for outbreak reconstruction

A **flexible** framework for outbreak reconstruction: bring your own data, prior, likelihood, MCMC.



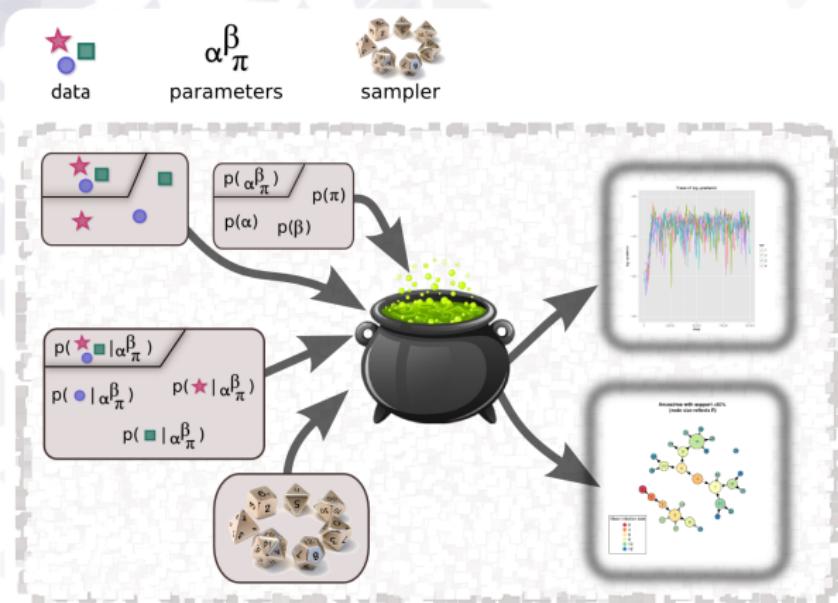
outbreaker2: a general tool for outbreak reconstruction

A **flexible** framework for outbreak reconstruction: bring your own data, prior, likelihood, MCMC.

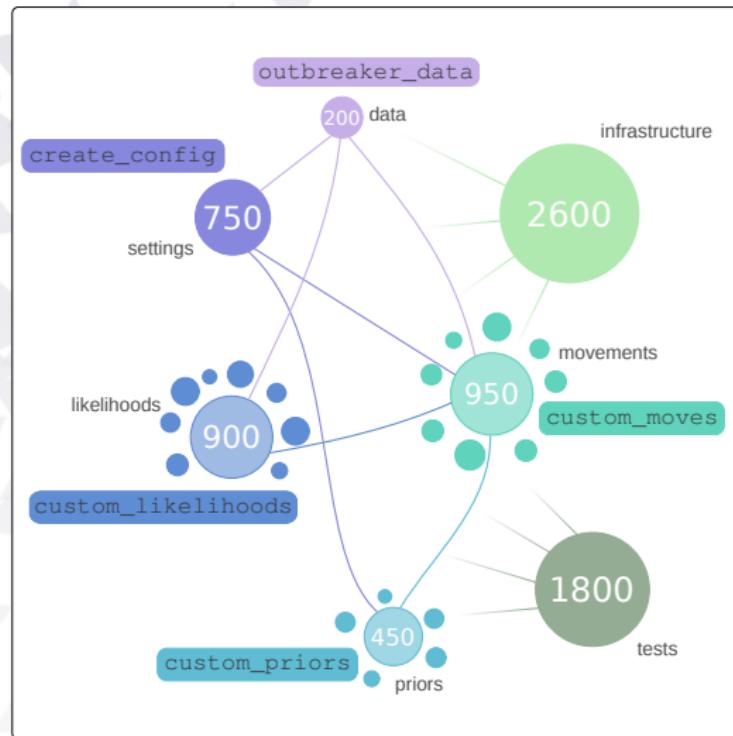


outbreaker2: a general tool for outbreak reconstruction

A **flexible** framework for outbreak reconstruction: bring your own data, prior, likelihood, MCMC.



outbreaker2: a modular implementation



Example: implementing *TransPhylo* in *outbreaker2*

outbreaker likelihood

- $p(t, s|\alpha, T^{inf}, \kappa, \mu, \pi) = p(t|T^{inf})p(T^{inf}|\alpha, \kappa)p(s|\alpha, \kappa, \mu)p(\kappa|\pi)$
- i.e. *incubation* \times *generation time* \times *genetic (simple)* \times *missing cases*

Example: implementing *TransPhylo* in *outbreaker2*

outbreaker likelihood

- $p(t, s|\alpha, T^{inf}, \kappa, \mu, \pi) = p(t|T^{inf})p(T^{inf}|\alpha, \kappa)p(s|\alpha, \kappa, \mu)p(\kappa|\pi)$
- i.e. *incubation* \times *generation time* \times *genetic (simple)* \times *missing cases*

TransPhylo likelihood

- $p(G|\beta, \gamma, N_{eg}, \alpha) = p(G|N_{eg}, \alpha) \times p(\alpha|\beta, \gamma)$
- i.e. *phylogeny (coalescent)* \times *SIR*

Example: implementing *TransPhylo* in *outbreaker2*

outbreaker likelihood

- $p(t, s|\alpha, T^{inf}, \kappa, \mu, \pi) = p(t|T^{inf})p(T^{inf}|\alpha, \kappa)p(s|\alpha, \kappa, \mu)p(\kappa|\pi)$
- i.e. *incubation* x *generation time* x *genetic (simple)* x *missing cases*

TransPhylo likelihood

- $p(G|\beta, \gamma, N_{eg}, \alpha) = p(G|N_{eg}, \alpha) \times p(\alpha|\beta, \gamma)$
- i.e. *phylogeny (coalescent)* x *SIR*

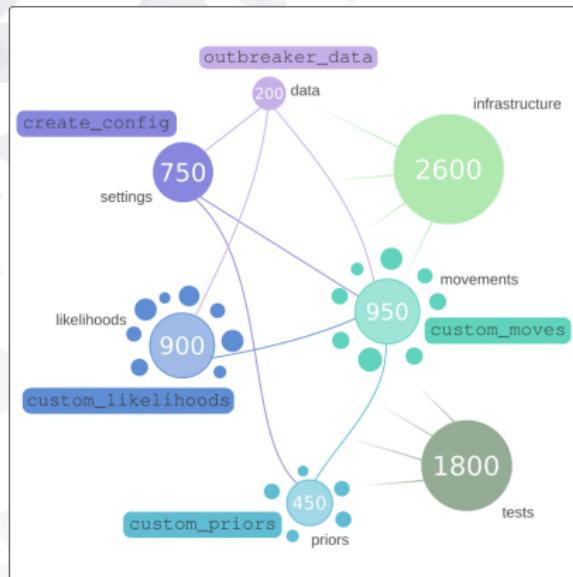
Can we combine the two models?

o2mod.transphylo: a new transmission model

$$p(t, G | \alpha, T^{inf}, \kappa, \pi, N_{eg}) =$$

$$p(t | T^{inf}) p(T^{inf} | \alpha, \kappa) p(G | N_{eg}, \alpha) p(\kappa | \pi)$$

i.e. *incubation x generation time x coalescent x missing cases*

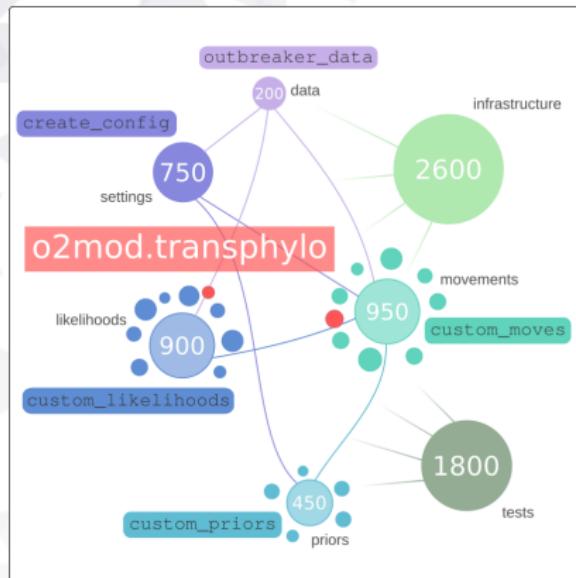


o2mod.transphylo: a new transmission model

$$p(t, G | \alpha, T^{inf}, \kappa, \pi, N_{eg}) =$$

$$p(t | T^{inf}) p(T^{inf} | \alpha, \kappa) p(G | N_{eg}, \alpha) p(\kappa | \pi)$$

i.e. *incubation x generation time x coalescent x missing cases*



o2mod.TransPhylo: TransPhylo module for outbreaker2

[implementation by Xavier Didelot and Finlay Campbell]

3 Custom likelihood

In order to calculate the likelihood under the TransPhylo model, we need to (i) extract the transmission tree from the outbreaker2 parameter, (ii) combine this transmission tree with the phylogenetic tree to form a colored tree, and (iii) calculate the likelihood of this colored tree. Step (i) is easy since transmission tree are encoded almost in the same way in TransPhylo and outbreaker2. For step (ii) we have to write the `combine` function which is tedious but not especially interesting (this function is included in this Rnw file but its code is not shown in the pdf). For step (iii) we only need to call the appropriate function of the TransPhylo package which is `probTreeGivenTree`. During step (ii) messages can arise indicating that the transmission tree and phylogenetic tree are in fact incompatible, in which case the likelihood is returned as -Inf.

```
lik_TransPhylo <- function(data, param) {
  ttree <- list(ttree = chind(param$tt_inf, data$dates, param$alpha),
    nam = data$ptree$nam)
  ttrees$ttree[which(is.na(ttrees$ttree[,3])),3] <- 0
  txt <- capture.output(ctree <- combine(ttree, data$ptree))
  if (length(txt)>0) {
    prob <- probTreeGivenTree(ctree, neg = 365 * 0.25)
  } else {
    prob <- -Inf
  }
  return(prob)
}

## Function to calculate the likelihood
## Function (data, param, i = NULL, custom_functions = NULL)

## Function (data, param, i = NULL, custom_functions = NULL)
## Function (data, param, i = NULL, list_custom_ll = new_model) {
##   for (i in 1:datas$N) {
##     current_ll <- api$pp_ll_all(data, param, i = NULL, list_custom_ll)
##     modif <- sample(c(-100:-1,1:100), 1)
##     param$ll_inf[i] <- param$ll_inf[i] + modif
##     new_ll <- api$pp_ll_all(data, param = NULL, list_custom_ll)
##     if (log(modif)) > (new_ll - current_ll) {
##       param$ll_inf[i] <- param$ll_inf[i] + modif
##     }
##   }
##   return(param)
## }

new_moves <- custom_moves(t_inf = new_move_tinf)
new_moves

## 
## 
## ////////////// outbreaker movement functions ///
## 
## classes: outbreaker_moves list
## number of items: 8
## 
## /// movement functions //
## 
## 
```

o2mod.TransPhylo: TransPhylo module for outbreaker2

[implementation by Xavier Didelot and Finlay Campbell]

3 Custom likelihood

In order to calculate the likelihood under the TransPhylo model, we need to (i) extract the transmission tree from the outbreaker2 parameter, (ii) combine this transmission tree with the phylogenetic tree to form a colored tree, and (iii) calculate the likelihood of this colored tree. Step (i) is easy since transmission tree are encoded almost in the same way in TransPhylo and outbreaker2. For step (ii) we have to write the `combine` function which is tedious but not especially interesting (this function is included in this Rnw file but its code is not shown in the pdf). For step (iii) we only need to call the appropriate function of the TransPhylo package which is `probTreeGivenTree`. During step (ii) messages can arise indicating that the transmission tree and phylogenetic tree are in fact incompatible, in which case the likelihood is returned as `-Inf`.

```
lik_TransPhylo <- function(data, param) {
  ttree <- list(ttree = chind(param$t_inf, data$dates, param$alpha),
    nam = data$pmtree$nam)
  ttree$ttree[which(is.na(ttree$ttree[,3])),3] <- 0
  txt <- capture.output(ctree <- combine(ttree, data$pmtree))
  if (length(txt)>0) {
    prob <- probTreeGivenTree(ctree, neg = 365 * 0.25)
  } else {
    prob <- -Inf
  }
  return(prob)
}
```

likelihood

movement function

```
args(api$cpp_ll_all)

## function (data, paran, i = NULL, custom_functions = NULL)
## NULL

new_move_tinf <- function(paran, data, list_custom_ll = new_model) {
  for (i in 1:datas$N) {
    current_ll <- api$cpp_ll_all(data,paran, i = NULL, list_custom_ll)
    modif <- sample(c(-100:-1,1:100), 1)
    paran$ll.inf[i] <- paran$ll.inf[i] + modif
    new_ll <- api$cpp_ll_all(data,paran, i = NULL, list_custom_ll)
    if (log(modif[i]) > (new_ll - current_ll)) {
      paran$ll.inf[i] <- paran$ll.inf[i] + modif
    }
  }
  return(paran)
}

new_moves <- custom_moves(v_inf = new_move_tinf)
new_moves

##
## /////////////////////////////////////////////////////////////////// outbreaker movement functions ///
## 
## class: outbreaker_moves list
## number of items: 8
## 
## /// movement functions ///
## End
```

o2mod.TransPhylo: TransPhylo module for outbreaker2

[implementation by Xavier Didelot and Finlay Campbell]

3 Custom likelihood

In order to calculate the likelihood under the TransPhylo model, we need to (i) extract the transmission tree from the outbreaker2 parameter, (ii) combine this transmission tree with the phylogenetic tree to form a colored tree, and (iii) calculate the likelihood of this colored tree. Step (i) is easy since transmission tree are encoded almost in the same way in TransPhylo and outbreaker2. For step (ii) we have to write the `combine` function which is tedious but not especially interesting (this function is included in this Rnw file but its code is not shown in the pdf). For step (iii) we only need to call the appropriate function of the TransPhylo package which is `probTreeGivenTree`. During step (ii) messages can arise indicating that the transmission tree and phylogenetic tree are in fact incompatible, in which case the likelihood is returned as `-Inf`.

```
lik_TransPhylo <- function(data, param) {
  ttree <- list(ttree = chind(param$tt_inf, data$dates, param$alpha),
               nan = data$ptrue$nan)
  ttree$ttree[which(is.na(ttree$ttree[,3])),3] <- 0
  txt <- capture.output(ctree <- combine(ttree, data$ptrue))
  if (length(txt)==0) {
    prob <- probTreeGivenTree(ctree, neg = 365 * 0.25)
  } else {
    prob <- -Inf
  }
  return(prob)
}
```

likelihood

Total: 25 lines of R
outbreaker2: 7,500 lines of R/C++
Code difference: 0.3%

movement function

```
args(api$cpp_ll_all)

## function (data, param, i = NULL, custom.functions = NULL)
## NULL

new_move_tinf <- function(param, data, list_custom_ll = new_model) {
  for (i in 1:datasize) {
    current_ll <- api$cpp_ll_all(data, param, i = NULL, list_custom_ll)
    modif <- sample(c(-100:-1, 1:100), 1)
    param$ll.inf[i] <- param$ll.inf[i] + modif
    new_ll <- api$cpp_ll_all(data, param, i = NULL, list_custom_ll)
    if (log(modif[i]) > log_ll(current_ll)) {
      param$ll.inf[i] <- param$ll.inf[i] - modif
    }
  }
  return(param)
}

new_moves <- custom_moves(v_inf = new_move_tinf)
new_moves

## 
## //////////////////////////////////////////////////////////////////
## classes: outbreaker_moves list
## number of items: 8
## 
## // movement functions //
## End
```

o2mod.TransPhylo: TransPhylo module for outbreaker2

[implementation by Xavier Didelot and Finlay Campbell]

3 Custom likelihood

In order to calculate the likelihood under the TransPhylo model, we need to (i) extract the transmission tree from the outbreaker2 parameter, (ii) combine this transmission tree with the phylogenetic tree to form a colored tree, and (iii) calculate the likelihood of this colored tree. Step (i) is easy since transmission tree are encoded as lists in outbreaker2 in a very similar way to phylogenetic trees. For step (ii), we have to write the `combine` function. This is not as easy as it sounds, especially interesting (this function is included in the package `outbreaker2`). The reason is that the likelihood is calculated only to call the appropriate function in the `outbreaker2` parameter `likelihood`. If the likelihood is not defined, (iii) messages can arise indicating that the transmission tree and phylogenetic tree are in fact incompatible. In this case the likelihood is returned as `-Inf`.

```
lik_TransPhylo <- function(data, param) {
  ttree <- list(ttree = chind(param$t.inf, data$dates, param$alpha),
               nam = data$ptree$nam)
  ttree$ttree[which(is.na(ttree$ttree[,3]))] <- 0
  tt <- capture.output(ptree <- combine(ttree, data$ptree))
  if (length(tt)==0) {
    prob <- 1
  } else {
    prob <- exp(-sum(log(ptree)))
    if (prob <= 0.25) {
      prob <- 0
    }
  }
  return(prob)
}
```

likelihood

Total: 25 lines of R
outbreaker2: 7,500 lines of R/C++
Code difference: 0.3%

New stuff

Old wheel

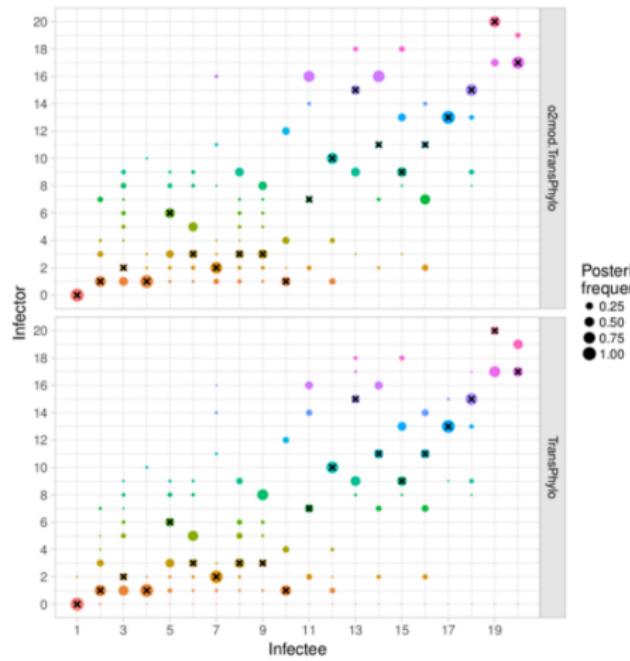
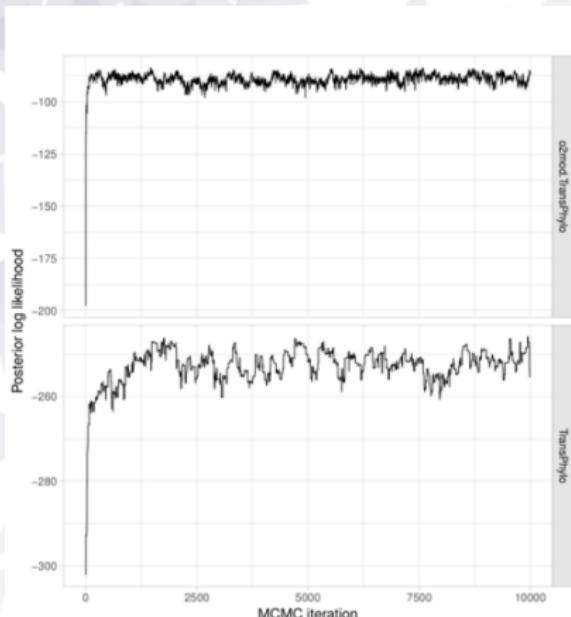
movement

all) power, t = 900, type = "S"

new_moves new_moves

o2mod. TransPhylo: results

[material by Finlay Campbell]



outbreaker2: summary

- when implemented, even very different models for outbreak reconstruction need to use the same “machinery”

outbreaker2: summary

- when implemented, even very different models for outbreak reconstruction need to use the same “machinery”
- different components implemented as separate **modules** facilitate customisation

outbreaker2: summary

- when implemented, even very different models for outbreak reconstruction need to use the same “machinery”
- different components implemented as separate **modules** facilitate customisation
- *outbreaker2* implements this infrastructure, using by default to the original *outbreaker* model (with some improvements)

outbreaker2: summary

- when implemented, even very different models for outbreak reconstruction need to use the same “machinery”
- different components implemented as separate **modules** facilitate customisation
- *outbreaker2* implements this infrastructure, using by default to the original *outbreaker* model (with some improvements)
- provides a new **platform** for model development and comparison

Thanks

- Lulla Opatowski
- *vimes crew:* Anne Cori, Pierre Nouvellet, Tini Garske, Hervé Bourhy, Emmanuel Nakouné
www.repidemicsconsortium.org/vimes
- *outbreaker2 crew:* Finlay Campbell, Anne Cori, Rich FitzJohn, Xavier Didelot, Neil Ferguson
www.repidemicsconsortium.org/outbreaker2
- *funding:* HPRU-NIHR, MRC

RECON

www.repidemicsconsortium.org

Questions?