Detecting rogue taxa

James Fleming - j.f.fleming@nhm.uio.no

@JamesfvFleming

NHM UiO

What is a Rogue?

- Rogues "assume varying and often contradictory positions in the tree set"
- What does this mean?
  - Rogues might be benign, just occurring in odd places in your results
  - But they might warp the entire tree around them, causing false signals that draw sequences towards them.
  - They may also decrease the confidence of key nodes, pushing you to be more conservative in your interpretations
- Overall, bad news

What can a Rogue look like?

- A Rogue Sequence could be
  - A member of an under-sampled or depauperate clade with few nearby relations to form relationships with.
  - A member of a group with a unique evolutionary history that might be faster or slower than the rest of the tree
  - A member of an unrelated group included in the dataset by incorrect sampling, contamination, or an historic mischaracterisation
  - A sequence with high fragmentation, or an accumulation of sampling errors
- Rogues are defined by their effect on our data, not by their origin or form

Isn’t more data better?

- More data can be better
- But good data is far more important
- And the right data to answer the question most of all.
  - Sampling evenly from groups across a dataset
  - High quality and deep coverage sequencing where possible
  - Accounting for problems in the first two using modelling and testing methods, and removing data when necessary

Can we reform a Rogue?

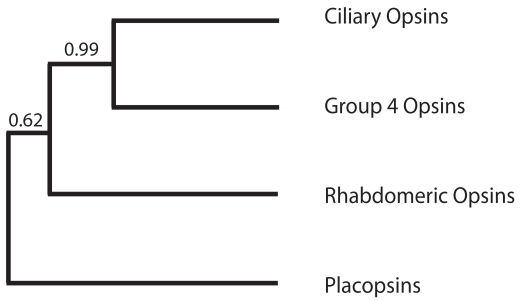
- In some cases, yes!
  - Sampling errors and contamination need to be removed
  - But insufficient sampling might just mean a sequence needs to be shelved for another day
  - A unique evolutionary history might need different or more powerful models
- And even in their rogueish behaviour, they might tell us important things
  - Has this taxa been historically misclassified?
  - **Could they root out other, more covert, rogue taxa?**

Detecting Rogues

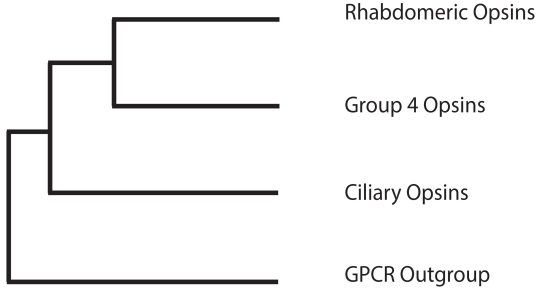
- Your dataset is special, and that means there are a range of tools to help deal with problems that might occur.
- As Rogue Sequences are identified by effect, different causes need different strategies.
  - Adding more samples to an undersampled group might anchor a rogue sequence there.
  - Adding more samples from a group with a complex and unique evolutionary history might magnify the effect of the rogues.

An Example:

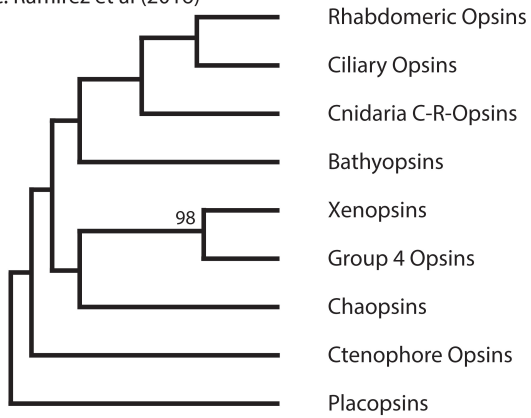
A. Schnitzler et al (2012), Feuda et al (2012, 2014), Hering & Meyer (2014)

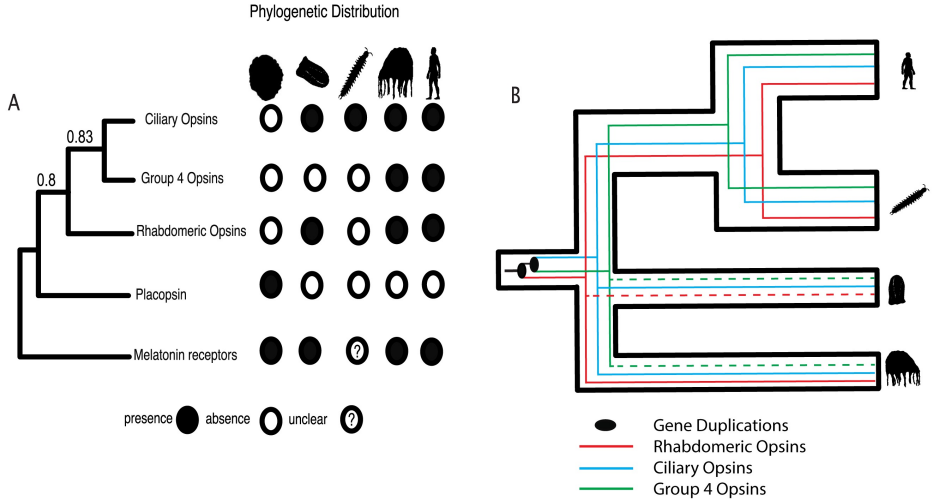


B. Porter et al (2011)



C. Ramirez et al (2016)



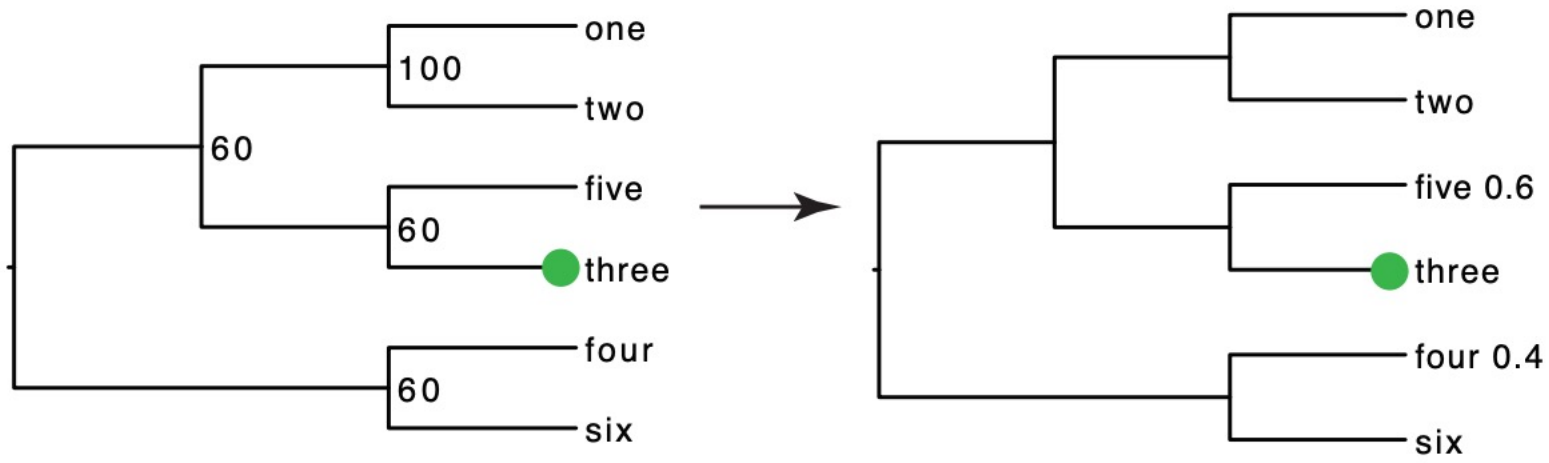


PhyUtility

- PhyUtility is a general purpose phylogenetics software released in 2007.
- It is useful for all kinds of tree manipulation:
  - Pruning
  - Rerooting
  - Creating consensus trees
- But it can also produce metrics to detect rogue sequences!

PhyUtility: Branch Attachment Frequency

- Very simply – where else is that branch arising?
- The Branch Attachment Frequency shows the alternative locations of a given branch to help us understand how and why the low support might be occurring.

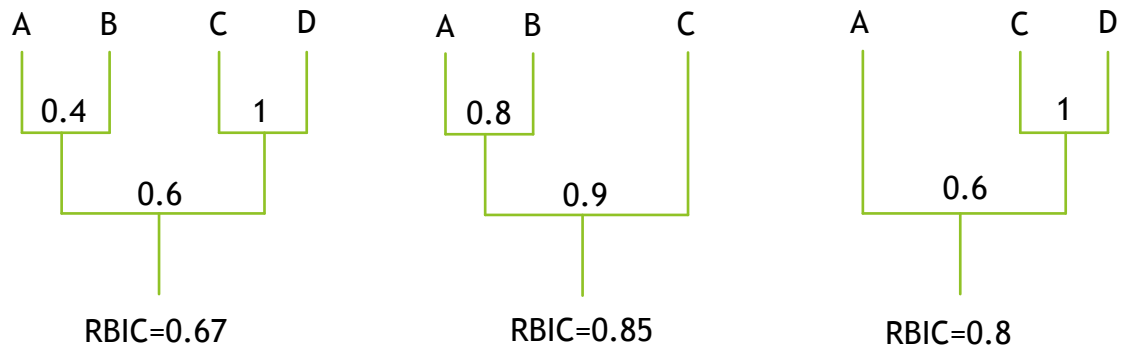


Roguenarok

- Roguenarok calculates three metrics to find rogue sequences:
  - “Relative Bipartition Information Criterion”
  - “Leaf Stability Index”
  - “Taxonomic Stability index”

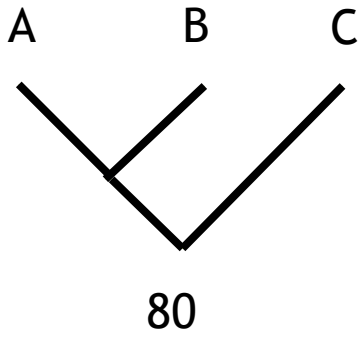
Roguenarok: Relative Branch Information Criterion

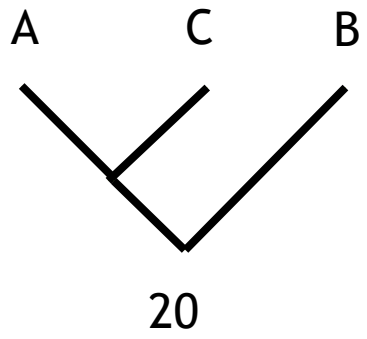
- That's the sum of all the support values divided by the maximum possible support value for the set. So a higher RBIC is a more robust dataset – more well-supported.
- Then, Roguenarok calculates the smallest number of taxa to be removed for the largest increase in RBIC over the data, by taking as input a list of trees.
- It seems like the A/B node is less stable, but removing D actually increases the RBIC the most, and so that is recommended by Roguenarok rather than removing the apparently more unstable node set!

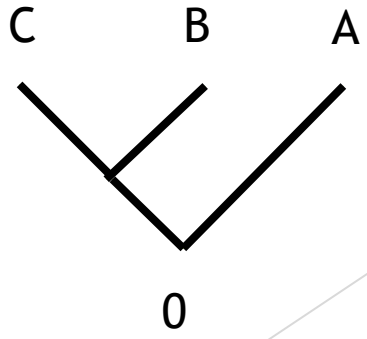


Leaf Stability

- First proposed by (Thorley & Wilkinson 1999)
- The relationship between three leaves can always be expressed as a triplet.
- This measure of triplet stability is the absolute difference between the BPs of the two best-supported 3-taxon statements for the triplet.
- This LS below is 60. A higher LS shows a more well-supported triplet





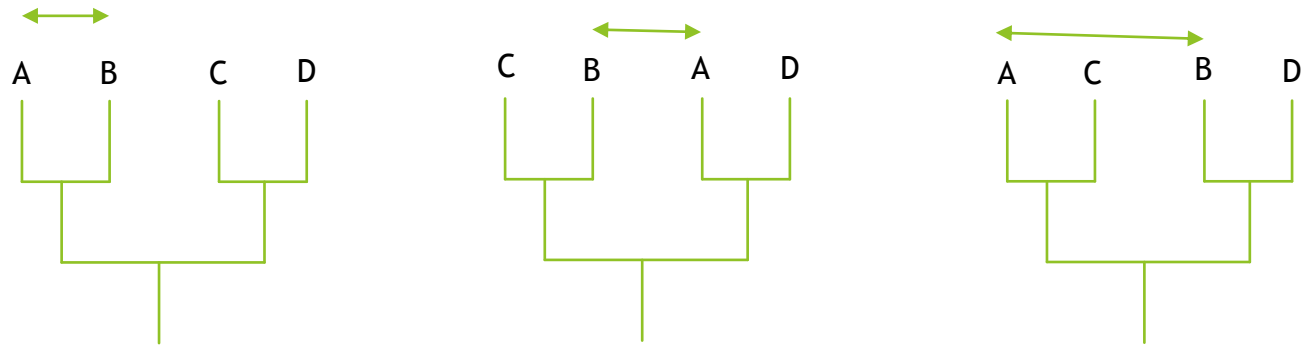


Leaf Stability

- But what if we have more than three taxa?
- Trees can be simplified down to repeated questions of three taxa by combining the leaf nodes above the node in question into a single hierarchy.
- Some implementations of Leaf Stability, including the one in Roguenarok, use quartets rather than triplets!
  - This takes longer, but can be more informative.

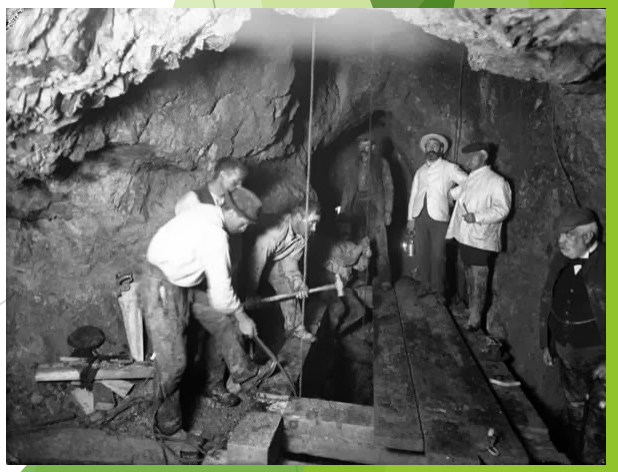
Roguenarok: Taxonomic Instability Index

- Taxonomic Instability Index is measures the difference in pairwise distances between taxa across the bootstrap.
- A larger pairwise distance for a taxa means that it is more mobile in the tree, and as such potentially a rogue sequence!
- In this example, C appears in three positions in the tree, suggesting it is more mobile. The relative position of D to the other sequences helps label it as the culprit, with a high TII than the other sequences.



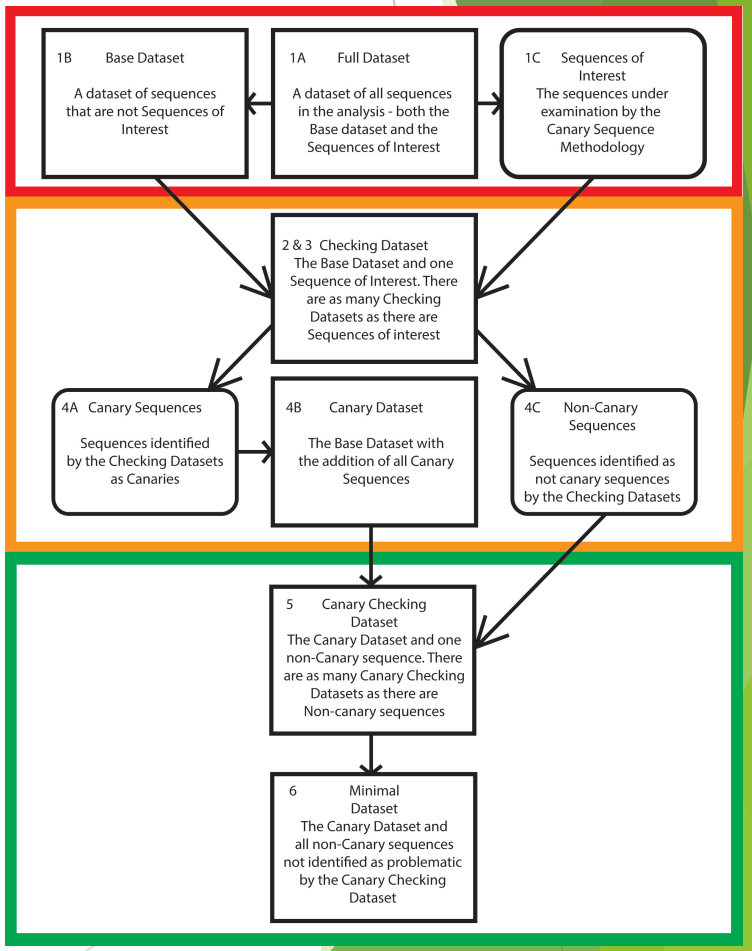
The Canary Sequence Approach

- The Canary Sequence Approach is an approach that identifies and removes problematic sequences from a single gene dataset, by identifying sequences that are mobile in the presence of problematic sequences.
- The approach was named after the Canary Mining practice – in the UK and Ireland, Canaries were often held in cages in coal mines – if a pocket of methane was exposed, the bird would begin to sing, and then quickly die from methane exposure, alerting the miners to vacate the shaft. Here, the mobile sequences are our ”Canaries”



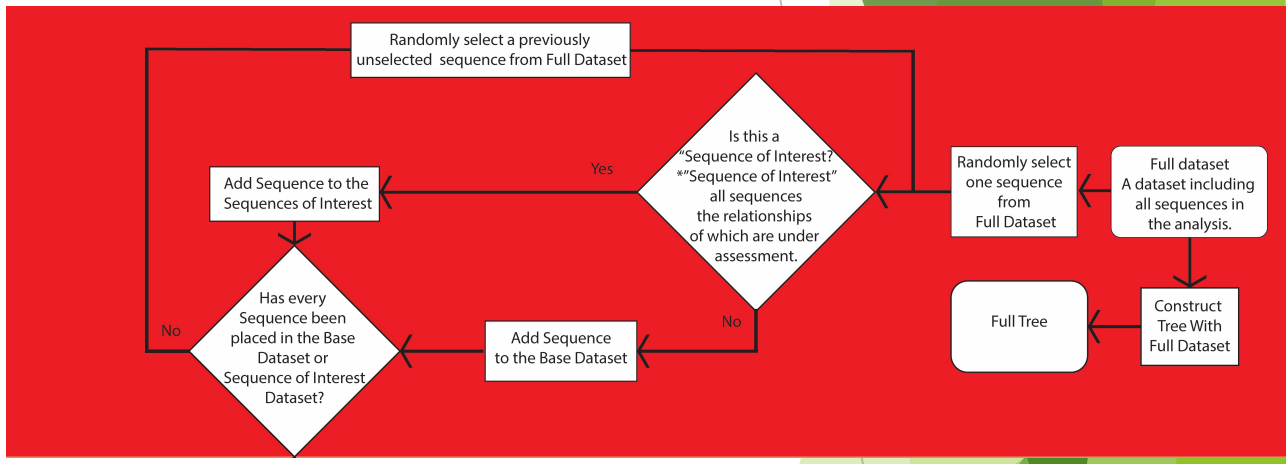
Canary Sequence Approach: Overview

- The Canary Sequence Approach takes place over three phases.
- In the first phase, an understanding of the dataset is first obtained.
- In the second, the canary sequences are identified.
- In the third, the canary sequences are used to identify and exclude problematic sequences, producing the final dataset.



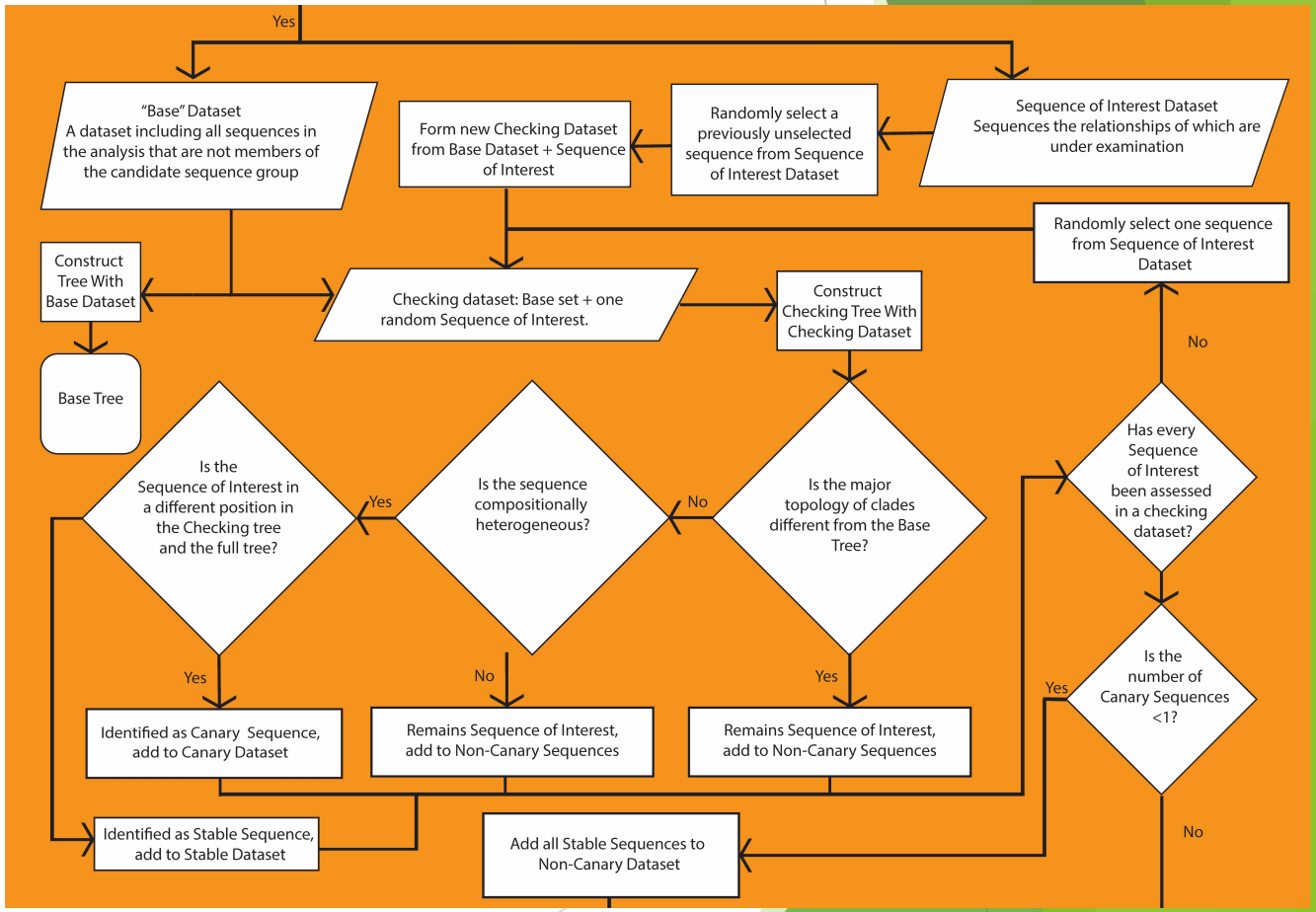
Canary Sequence Approach: Part 1

- During part one, a **Full Dataset** and a **Base Dataset** are constructed.
- The **Full Dataset** contains all of the studied sequences
- The **Base Dataset** contained all of the sequences that aren't part of your pre-determined **Sequences of Interest** (Sequences you suspect may be problematic)



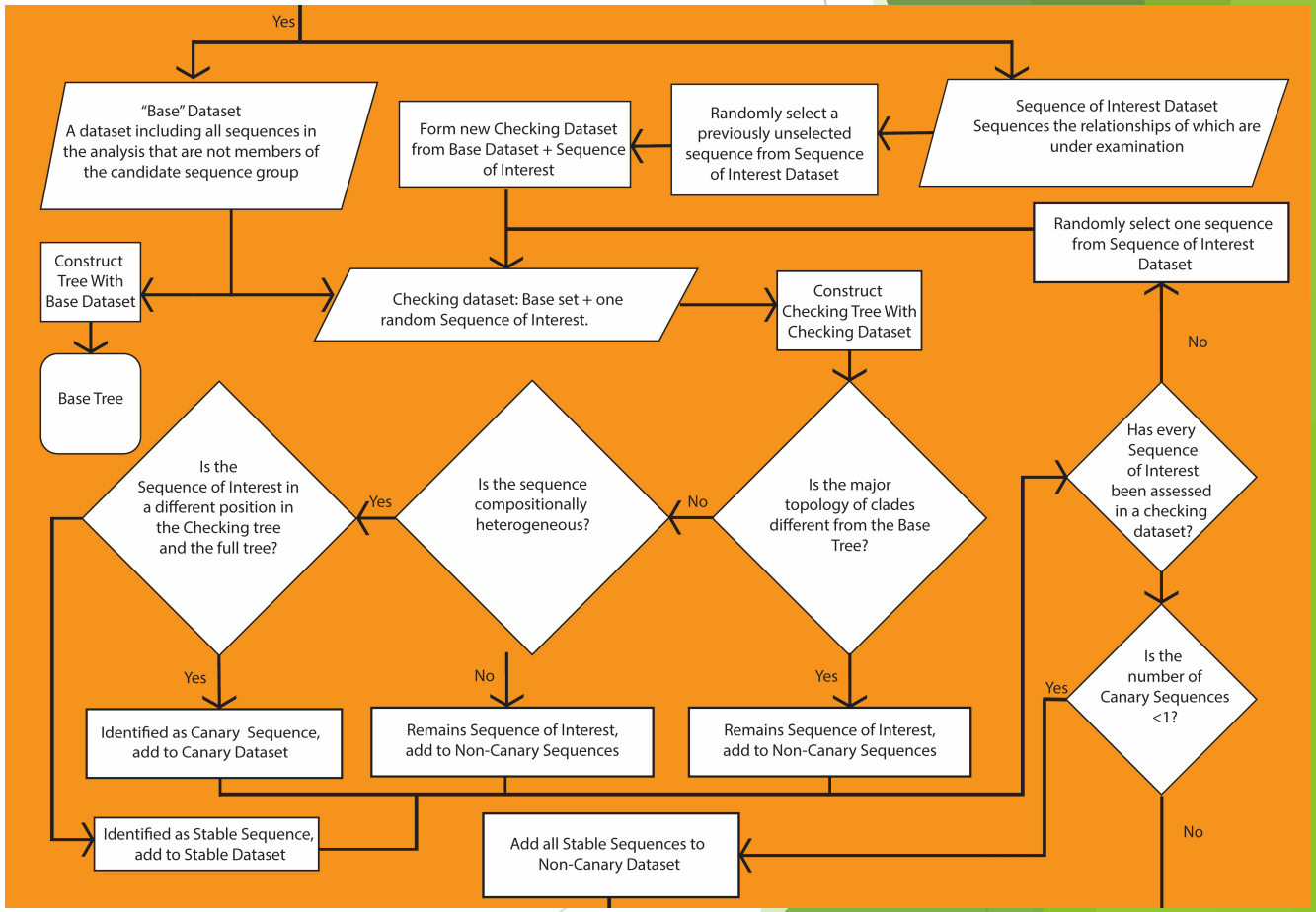
Canary Sequence Approach: Part 2

- During part two, Canary Sequences are identified.
- One **Checking Dataset** is constructed for each Sequence of Interest, consisting of the **Base Dataset + 1 Sequence of Interest.**
- The position of the Sequence of interest in the **Checking Tree** and the **Full Tree** is then observed.



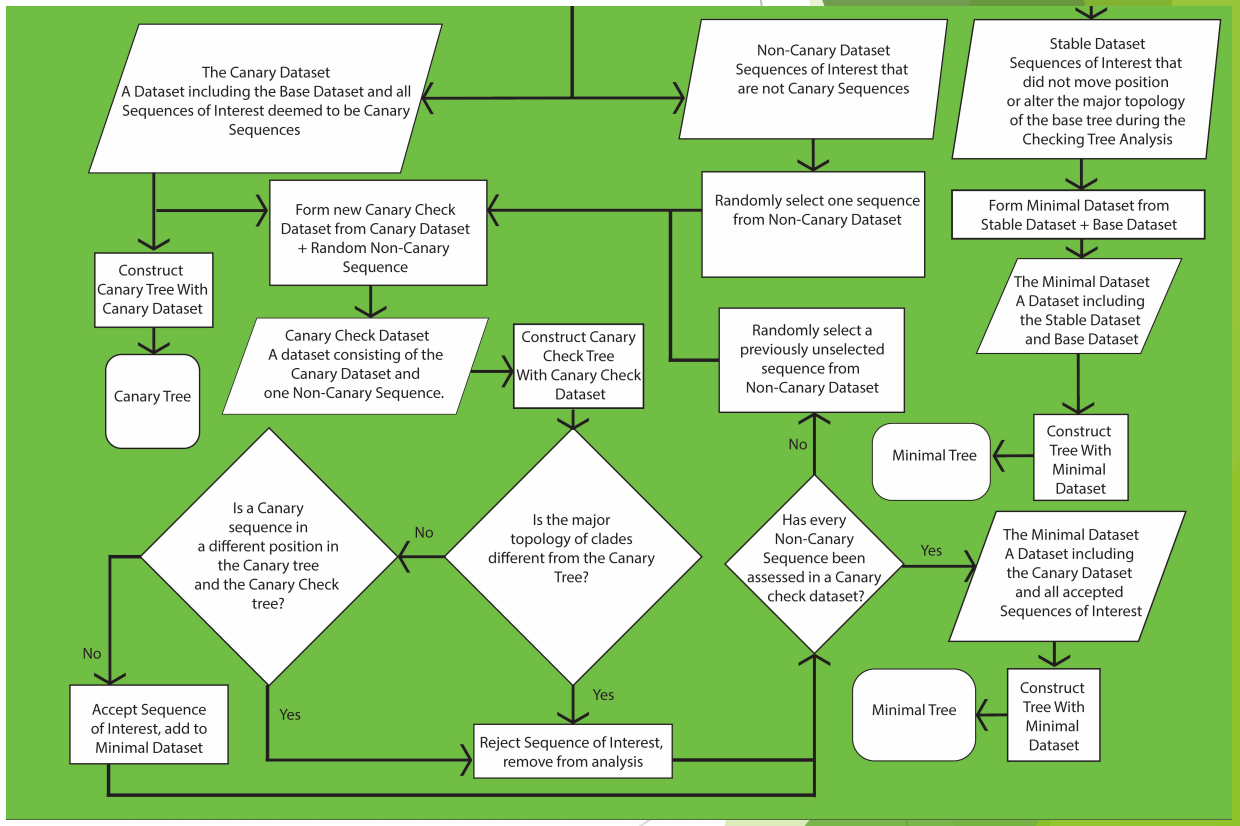
Canary Sequence Approach: Part 2

- A **Canary Sequence**:
  - Is compositionally heterogeneous.
  - Is in a different position in the **Checking Tree** and **Full Tree**.
  - However, otherwise, the **Checking Tree** and **Base Tree** are isomorphic.
- Sequences that satisfy this criteria are Canary sequences, otherwise they are Non-Canary Sequences.



Canary Sequence Approach: Part 3

- The **Canary Sequences** are then combined into a single dataset with the **Base Dataset**, called the **Canary Dataset**.
- One **Canary Checking Dataset** is constructed for each remaining Sequence of Interest, consisting of the **Canary Dataset + 1 Sequence of Interest**.
- If the tree remains isomorphic between the **Canary Tree** and the **Canary Checking Tree**, the sequence is deemed not Problematic.
- **The Final Dataset** is formed from the Canary dataset plus all of the sequences deemed not problematic by the approach.



Efficacy in real and simulated datasets

- When applied to classic cases of Long Branch Attraction in Carranzo et al (1997) and Aguinaldo et al (1997), the Canary Sequence Approach resolved both issues, recovering the monophyly of Lophotrochozoa and Ecdysozoa, respectively.
- On simulated datasets generated based upon Aguinaldo et al (1997), the Canary Sequence Approach succeeded at identifying problematic sequences and recovering monophyly of Ecdysozoa in 66% of cases. In the 33% of failures, it did also not produce a false positive result, and so is conservative.

Advantages & Disadvantages of the Method

Pros:

- Model Scalable: rather than being tied to a particular phylogenetic method or system, the Canary Sequence Approach is system-agnostic, and as the models used to generate checking trees improve, so to will it.

Cons:

- 66% efficacy on simulated data: however, robust to false positives and superior to prior status quo.
- Requires a pre-defined set of “Sequences of Interest”
- Removes data from datasets – whilst it identifies the data as low quality, removal of data is a very rough approach to analysis.
- Low computational efficiency.

Robusticity and Reliability

- Both of today's software methods – Roguenarok and Phyutility, prioritise Robusticity.
- This means that they use the metric of whether a particular arrangement of taxa can be repeatedly recovered as a goal.
- But something can be reliable, or even accurate, without being robust.
- A small dataset may contain single sequences that provides a lot of phylogenetic information, for example.
- Model misspecification might cause downstream errors when using Roguenarok or Phyutility.
- Great tools, but use with awareness!

Summary and Conclusions

- Rogues "assume varying and often contradictory positions in the tree set"
- There are a multitude of ways to deal with them
  - Some work better for different datasets
  - Canary: Small single gene datasets with easily identifiable worrying sequences
  - RoguenaRok and Phyutility: Larger datasets where robusticity is important

Introducing the Exercise

- RoguenaRok and PhyUtility
- Canary is a bit too long-winded for the cookbook style
- We'll try and identify rogues using both these programs, and compare the results!