# [Title Slide]

Hi, my name is Nick Barrowman. I’m a statistician at the

* Clinical Research Unit of the
* Children’s Hospital of Eastern Ontario Research Institute
* in Ottawa, Canada.

Today I’m going to talk about my R package, **vtree**, and how it can be used to find hidden patterns in clinical data.

# Outline

* In the first part of my talk, I’ll introduce the basic functionality of vtree by showing *step by step* how to draw a CONSORT diagram.
* Then, in the second part of the talk, I’ll show how vtree can be used for data exploration.

# Part 1: Drawing a CONSORT flow diagram

* RCTs often called the **gold standard of clinical evidence**.
* To live up to this to this billing, an RCT has to be done *well* and be transparently reported.
* In 1996 there was a major step forward: the Consolidated Standards of Reporting Trials (the CONSORT statement), a key element of which is the CONSORT diagram.
* Some other tools are available for drawing CONSORT diagrams, but vtree is unique in two ways:
  1. It is reproducible
  2. It can be used for many other tasks (as we’ll see in part 2 of this talk)

# Does treatment of COVID-19 with paxlovid reduce hospitalization and death?

* I am going to start with an example: a randomized trial that was recently published in the New England Journal of Medicine.
* The trial studied the use of paxlovid, an oral antiviral, to treat Covid.
* This is the drug that President Biden was recently treated with.

# [1 - CONSORT diagram for paxlovid study]

* Shows flow of patients all the way from eligibility assessment, through randomization into two groups, and follow-up.

# [2 - CONSORT diagram for paxlovid study]

* As I was preparing this talk, I noticed an apparent error in the published CONSORT diagram:
* According to the diagram, 2396 patients were assessed for eligibility and then 137 were excluded (which would leave **2259**) but the diagram indicates that only **2246** were randomized, so there are **13 patients not accounted for**.
* I will assume 2396 is a misprint, and that in fact 13 fewer (2383) were assessed for eligibility

# A data frame for the paxlovid CONSORT diagram

# A single-layer tree

* vtree stands for “variable tree” because it uses a simple tree model
* Each variable in the tree generates a new layer.

# A two-layer tree

* This 2-layer tree shows the general structure of variable trees: nodes are nested subsets of the data frame.
* This tree includes a missing value node for **group**, because for those patients who were not randomized, there is of course no group.
* By default vtree shows missing values.
* Other tools sometimes hide missing values.
* (In some cases there is more missing data than non-missing data!)

# Pruning a node

* vtree has several different ways to specify which nodes to prune.
* Here we’re specifying that only the randomized patients should be followed.
* With each additional variable, the tree gets larger and larger. One way to keep a tree manageable is by pruning.

# A three-layer tree

# Adding labels

* vtree allows you to assign labels to nodes.
* Thanks to these node labels, the variable labels on the side are no longer needed and can be hidden.
* We have now produced a rough version of the CONSORT diagram with a fairly simple call to vtree.
* The diagram is still missing the details of exclusions and discontinuations.
* Next, I’ll briefly show how vtree provides a solution.

# Summaries in vtree

* [END OF SLIDE] A few additional details are needed to produce a CONSORT diagram, and given time constraints, I won’t go into all of these details.
* In the next slide, I show the finished product and the vtree call that generates it.

A detailed CONSORT diagram

* The vtree style also looks a bit different from the CONSORT style, but that’s just formatting.
* The key point is that it’s reproducible so it avoids errors.
* But now let’s move on to part 2 of this talk to see what else vtree can do.

# Part 2: Data Exploration

* I originally designed vtree to draw inclusion-exclusion flow diagrams like the CONSORT diagram.
* These diagrams are very useful for all kinds of study designs.
* But vtree’s generality makes it good for lots of other data exploration tasks.
* In my experience people often have data sets but don’t really know much about the data itself.
* I think this is partly because, beyond just visually examining a spreadsheet of data, people often don’t have easy-to-use tools to explore their data
* vtree makes it easy to find patterns in data that might otherwise be overlooked.

# A retrospective cohort study

# Frequencies

* Here’s an alternative version of a vtree applied to a single variable.
* This is called a “pattern tree”.
* In a moment we’ll see a more elaborate version.
* Note: the most frequent diagnoses are displayed at the bottom.

**For convenience, I'll assign some interpretable factor levels**

# Patterns

* We saw a pattern tree previously.
* Here’s a more complicated example.
* Pattern trees display a separate row for each combination of values.
* For convenience, pattern trees are arranged so that the most frequent combinations appear at the bottom.
* If there are missing values, this is particularly useful.

# CMV reactivation (R) post-transplant

* R represents the proportion of patients who experience CMV reactivation.

# Quantitative variables

* Although it is flexible, vtree is not suited to all types of data exploration.
* Vtree focuses on discrete variables.
* For quantitative variables there are many tools that are better suited to the task.
* Nevertheless, the summary feature is useful for summarizing quantitative variables.

# Dichotomizing a quantitative variable

* Vtree has lots more features and applications.
* In some situations it provides information like existing tools such as contingency tables, bar plots, etc.
* But in other situations, it provides information that would be difficult to obtain in other ways
* And it does this using a simple model.