# wiseR: Helping your decisions from complex datasets

Shubham Maheshwari, Tavpritesh Sethi

2021-07-26

# Getting Started

To install latest developmental version of wiseR in R

```
devtools::install_github('tavlab-iiitd/wiseR',build_vignettes=TRUE)
```

To launch the app:

```
wiseR::wiser()
```

# Bayesian Networks

## Why Bayesian Networks?

Networks are one of the most intuitive representations of complex data. However, most networks rely on pair-wise associations which limits their use in making decisions. Bayesian Networks (BNs) are a class of probabilistic graphical models which can provide quantitative insights from data. These can be used both for probabilistic reasoning and causal inference depending upon the study design.

A BN is a directed acyclic graph and provides a single joint-multivariate fit on the data with a list of conditional independencies defining the model structure. Unlike multiple pair-wise measures of association, fitting a model decreases the chance of false edges because the structure has to agree with global and local distributions. Importantly, unlike most other forms of Artificial Intelligence and Machine Learning, BNs are not a black-box model. These are transparent, interpretable and help the user in reasoning about the data generative process by looking at the motifs. This can be immensely useful in learning systems such as healthcare where a feedback between the clinicians and the learning system is paramount for adoption. The structure (independencies) of a BN can be learnt directly from data using machine learning or be specified by the user, thus allowing expert knowledge to be injected. After the dependence structure within the data is learnt (or specified) the parameters are learnt on the network using one of many possible approaches. wiseR provides most of the existing approaches such as constraint based algorithms and score-based algorithms for parametrization of the Bayesian Network. Recommendations as per the state-of-the-art in the literature are specified at each step of the BN learning process (i.e. the recommendatin of score based algorithms over constraint based algorithms for learning structure and Bayesian Information Criteria over Akaike Information Criteria for evaluating the fit). Parametrization of the network enables predictions and inferences. These inferences can be purely observational (“seeing” the state of a variable and its effect on neighbours) or causal (“doing” something to a variable, i.e. interventions and observing the effect on downstream nodes). wiseR provides scoring methods both for observational (e.g. Bayesian Information Criterion) and interventional (e.g. modified Bayesian Dirichlet Equivalent score) datasets.

The flexibility provided by BNs in probabilistic reasoning and causal inference has made these one of the most widely used artificial intelligence methods in Computer Science. However, the sophistication required to code all the features has limited their use by domain experts outside of computer science, e.g. clinicians. wiseR plugs this gap by creating a GUI for the domain experts and is an end-to-end solution for learning, decision making and deploying decision tools as dashboards, all from the wiseR platform.

## What do the motifs (junctions) reveal about the data-generating process

Carefully learnt BNs are representations of the generative process that produced the data. These generative processes are captured in the the form of three basic motifs present in the network: chain, fork and collider junctions.

### Chain (Mediator effect)

This motif is the simplest structure in a BN with a sequence of nodes pointing in the same direction. The intermediate nodes are called mediators and conditioning on any of the mediators makes the flanking nodes independent of each other. A mediator can be thought of as a mechanism, hence capturing the information about mechanism removes the need for capturing the triggering process, hence making the triggering process independent of the outcome.

### Fork (Confounders effect)

The second type of structure observed in a BN is a common parent pointing towards two or more child nodes. The parent represents the common cause and not accounting for the parent is a common source of error (confounding) in many statistical models built upon real world data. A folk example of such an effect is Age as a confounder (common cause) of IQ and shoe-size in children. Failure to condition upon age will lead to a spurious correlation between the IQ and shoe-size and change our reasoning (hence predictions) about the system being modeled.

### Collider or a V-structure (Inter-causal reasoning)

These are one of the most interesting motifs in an Bayesian Network and are indicated by two nodes pointing towards a single child node. Although the two parent nodes are not causally connected, conditioning on the state of the common child opens up a ghost-path for probabilistic inference flow (inter-causal reasoning) which can explain many intriguing paradoxes (e.g. the Monty Hall problem).

# Walkthrough of *wiseR* functionalities

## Pipeline

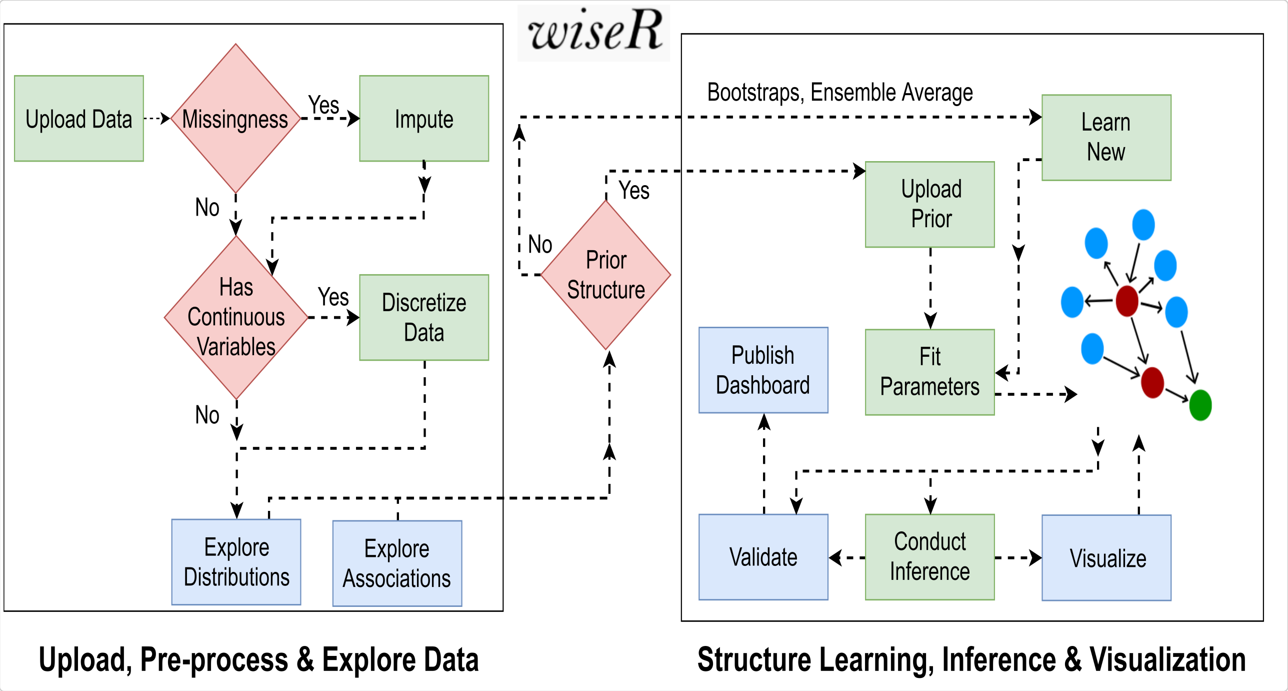


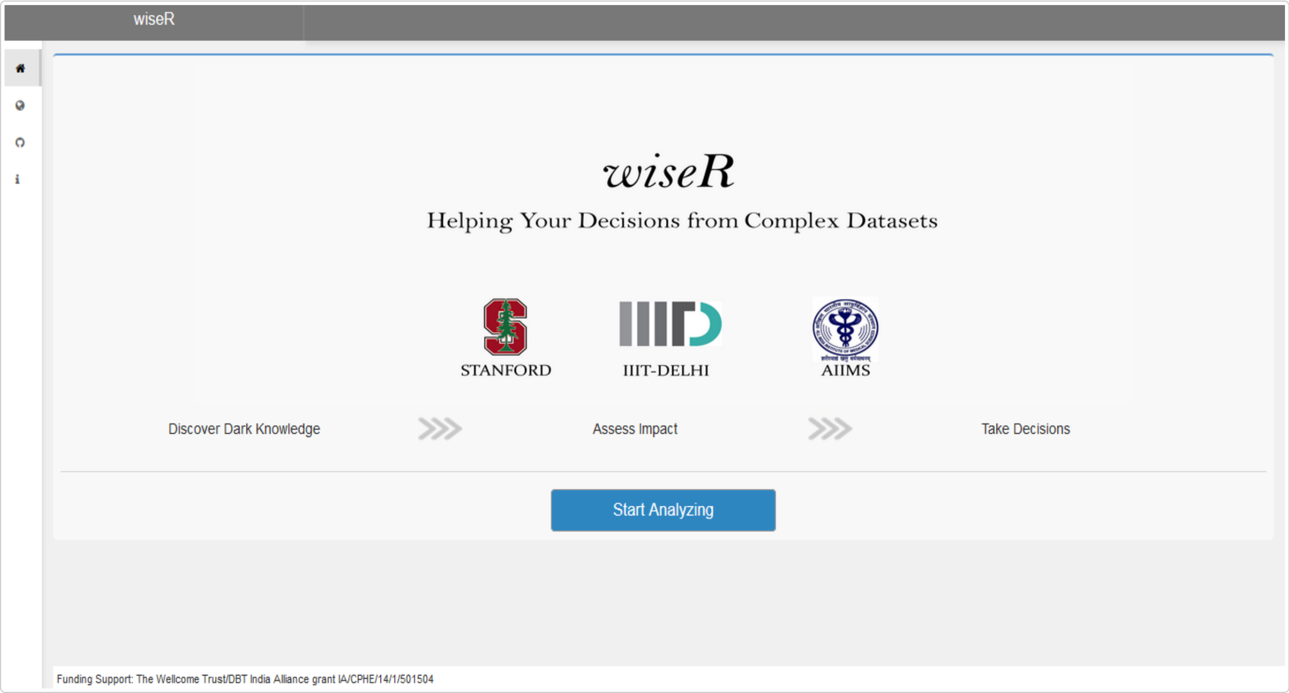
**Figure S1. Flowchart showing logic of *wiseR***

## Start

Calling the wiseR() function from the ***wiseR*** package launches the application in the default browser. The recommended browser for ***wiseR*** is Chrome. On launching the application for the first time, it conducts background checks, which takes 5-10 seconds depending upon the machine. This delay only happens on the first launch and all subsequent launches take less than a couple of seconds to initialize the user interface.

The toolbar on the left side of the page has icons for navigating to home, analysis engine, github development version and team information at any point of time.

Click “Start Analyzing” for navigating to the analysis engine with tabs specifying the functions. Hovering over most of the tabs brings up a tool-tip that briefly describes the function performed on that action.



wiseR

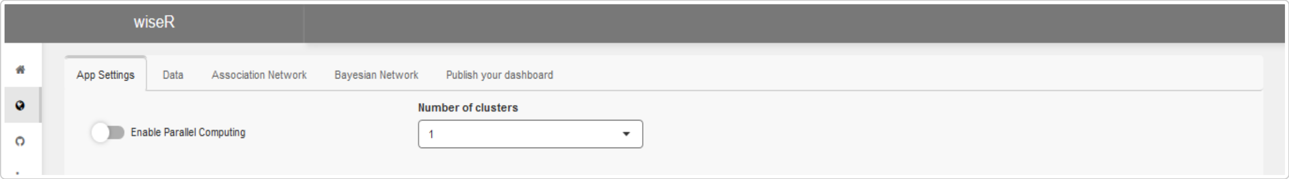
Helping Your Decisions from Complex Datasets

Funding Support: The Wellcome Trust/DBT India Alliance grant IA/CPHE/14/1/501504

Figure S2. Homepage of the *wiseR* application

## Main page of the *wiseR* engine

“Start Analyzing” takes us to the main page of the ***wiseR*** engine. An intuitive left-to-right ordering of tabs guides the user into the analysis. This page has 5 tabs named ‘App Settings’,‘Data’,‘Association Network’,‘Bayesian Network’ and ‘Publish your dashboard’, each tab covering a specific functionality (Figure S3). Each of these is described next.



Enable Parallel Computing

Number of clusters
1

**Figure S3. Functional tabs on the main page of the *wiser* engine and the first tab (App Settings) for parallel computation.**

## App Settings

This tab (Figure S3) is used to set the parallel computing option (number of cores) for learning BN structure. Learning structure is known to be NP-hard problem and may be time consuming on large datasets. Setting the number of clusters allows each core to learn a structure when bootstrap learning is performed. Since learning one structure cannot be parallelized, this is an example of task parallelization.

## Data

This tab is used to load, pre-process and explore the dataset (Figure S4.)

The dataset panel contains the upload and preprocess menu while explore panel is used to visualize the distribution of the data.