

# 1 Analysis

Analysis of this project will mainly consist of comparison to the implementation of the algorithms in R. It will compare the functionality of the algorithms on data and also the run speed of the code.

## 1.1 Initial Skeleton Generation

The skeleton generation code has almost identical functionality as the R implementation. Even the order in which edges are tested is the same. However the R implementation has a non-order dependent mode which is not currently implemented in the python version however could be added as an option in the future.

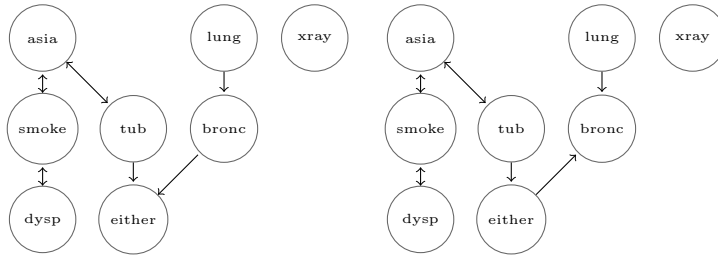
The Alarm\_1000, Asia\_1000, and Diabetes\_1000 datasets found in all produce identical skeletons as seen in Figure .

The  $\chi^2$  test for independence was not implemented in the pcalg library in R. Other independence test for discrete data, such as the  $G^2$  test, were implemented. However, the bnlearn library has an implementation of the  $\chi^2$  test to use for comparison.

Both tests return identical results on all tested data as seen in Table .

## 1.2 PC Algorithm Edge Alignment

### 1.2.1 Asia\_10000



### 1.2.2 Diabetes\_1000

### 1.2.3 Differences

There are some differences in the edge orientation of the different implementations. For example, in the orient of asia\_1000 network, the edge either-bronc is oriented in opposite directions by the two implementations. This is due to the order in which the colliders were oriented in the graph. In the R implementation the triple lung, bronc, either is oriented followed by the triple tub, either bronc. This causes the edge either-bronc to be overwritten and reoriented. In

the python implementation this is done in the opposite order resulting in a different network.

Neither order is more correct than the other however this orientation is used to estimate the final skeleton of the FCI algorithm so differences in the order can have a large knock on effect in the PAGs estimated by this algorithm.

### **1.3 FCI Algorithm Final Skeleton**

### **1.4 FCI Algorithm Edge Alignment**

#### **1.4.1 Asia\_10000**

#### **1.4.2 Diabetes\_1000**

#### **1.4.3 Differences**

### **1.5 Run Speed**

#### **1.5.1 Method**

#### **1.5.2 Measurements**

#### **1.5.3 Algorithms**

### **1.6 Skeleton Generation**

#### **1.6.1 Independence Tests**

The independence test was the main bottleneck of the algorithms, and was significantly slower than the R implementation. This is because the R implementation actually runs C code behind the scenes, c code is compiled rather than interpreted like python and R. This makes the code much faster.

However the independence test in the algoirthm is a parameter and thus the current slow test could be replaced by a faster version which calls C code like the R implementation.