Model-based clustering of genomic aberrations: Generalizing the Instability Selection Network Model

Hyuna Yang and Michael Newton

August 1, 2005

Contents

1 Introduction

We have developed two model-based clustering methodologies (instability selection tree-like network (ISTN) model and instability selection general network (ISGN) model) for characterizing dependency structure in cancer genomic aberrations recorded by comparative genomic hybridization (CGH), one of the techniques used to measure gains and losses (aberrations) in DNA copy number. Two methods are based on Instability Selection Network (ISN) model proposed by Newton (2002). The ISN model provides sets of genomic aberrations called *ensembles* and the co-occurrence of all aberrations in an ensemble in a progenitor cell is relevant to tumor progression. In the original ISN model, aberrations residing on one ensemble could not reside on another, to simplify computations. To extend this limitation Newton *et al.* (2003) introduced the ISTN model in which ensembles can be organized in a tree-like structure. Later ISGN model was developed, which has no restrictions on the set of ensembles and thus give more flexible

structure than a tree-like network. Two methodologies (ISTN and ISGN) are implemented in the R package ISN. This vignette provides a brief overview of methodologies and their implementation.

2 Instability Selection Network

ISN model analyzes CGH data using two biological features: genetic instability causing several forms of aberrations, and a cell-level selection that characterizes aberrations which are relevant to tumor growth.

Suppose there are n aberrations $\{1, 2, \ldots, n\}$ to be measured in each tumor. Let Z_i denote the indicator that aberration i occurs in a progenitor cell, and let X_i be the measurement of Z_i. The data from the tumor is $X = (X_1, \ldots, X_n)$, and both X_i and Z_i are treated as Bernoulli trials. Then the ISN model can be:

1. Genetic Instability.
   Let each $Z_i \sim$ iid Bernoulli(θ). This encodes, the idea of neutral random damage. Two error rates γ and δ indicate measurement error. Then we define CGH observation X_i conditionally on Z_i as $P(X_i = 1|Z_i = 0) = \gamma$ and $P(X_i = 0|Z_i = 1) = \delta$. Thus $X_i \sim$ iid Bernoulli (α), here $\alpha = (1 - \delta)\theta + \gamma(1 - \theta)$.

2. Cell-level Selection.
   Among the aberrations $Z = (Z_1, \ldots, Z_n)$, some might be noise and some might be critical aberrations for tumor progression. To describe those critical genomic aberrations, the ISN model uses sets of aberrations. The co-occurrence of aberrations in a set is beneficial to the tumor progress. To carry out this idea, let $\mathcal{C} = \{C_1, \ldots, C_K\}$ denote the *oncogenetic network*. For each $k = 1, \ldots, K$, *ensemble* C_k is a collection of aberrations whose co-occurrence is beneficial to the tumor progress in a certain sense. Also let $C_0 = \left[\bigcup_{k=1}^{K} C_k\right]^c$ be a collection of aberrations whose occurrence is irrelevant to tumor development.

   Selection, denoted by SEL, is defined SEL$=\bigcup_{k=1}^{K} A_k$, where $A_k = \bigcap_{i \in C_k} [Z_i = 1]$. Namely, SEL occurs if all aberrations occur in any ensemble. Since x is only observed when SEL occurs, the joint probability mass function for $x = (x_1, \ldots, x_n)$ is $p(x) = P(X = x|\text{SEL}) = \frac{P(x)P(\text{SEL}|x)}{P(\text{SEL})}$.

3 Instability Selection Tree-like Network (ISTN)

In ISTN model, ensembles can make a tree-like structure. To make the tree-like network, let edges $(\{e_1\}, \{e_2\}, \ldots)$ be a partition (i.e. mutually exclusive and exhaustive) of the full set of the relevant aberrations such that each ensemble C_k $(k = 1, \ldots, K)$ is a union

of edges. To each edge e_j we suppose there is a parent edge PA(e_j) which is either one of the other edges or a special "root". In tree-like network, a root is an imaginary starting point, and no edge can be a parent of the root or the root itself. If e^*=PA(e), then we say e is a child edge of e^*. A leaf edge is an edge having no child edge. To be a tree-like network, we assume that the edges which constitute an ensemble form a unique path from any leaf e to PA(e) to PA(PA(e)) and so on to the root. Each ensemble C_k ($k = 1, \ldots, K$) is such a path, and the set of ensembles make a tree-like network.

To get a probability mass function, let us define some probabilities. Let "edge e is open" mean that for all aberrations $i \in e$, $z_i = 1$. For an e which is not a leaf edge, let "branch e is open" indicate there is at least one series of open edges from any leaf edge e^* to PA(e^*) to PA(PA(e^*)) and so on to e. Let "ensemble C_k is open" mean that for all aberrations $i \in C_k$, $z_i = 1$. Let p_e be the probability of each edge e being open, and $\tilde{p}_e$ be a probability of branch e being open, both according to the instability component. When e is a leaf edge,

$$\tilde{p}_e = p_e = P\left[\bigcap_{i \in e} [Z_i = 1]\right] = \theta^{m_k}$$

otherwise, when e is not a leaf edge,

$$\tilde{p}_e = P(\text{branch } e \text{ is open}) = p_e \left[1 - \prod_{h:PA(h)=e} \{1 - \tilde{p}_h\}\right].$$

Similarly let $p_e(x)$ be the conditional probability of an edge e being open given x, and $\tilde{p}_e(x)$ be a conditional probability of branch e being open given x. When an edge e is a leaf edge,

$$\begin{aligned} p_e(x) &= \tilde{p}_e(x) = P(\text{leaf edge } e \text{ is open}|X = x) = P\left[\bigcap_{i \in e} [Z_i = 1]|X = x\right] \\ &= P(Z_i = 1|X_i = 1)^{\sum_{i \in e}(x_i)} P(Z_i = 1|X_i = 0)^{\sum_{i \in e}(1-x_i)} = a^s b^{m_k - s}. \end{aligned}$$

Here $a = P(Z_i = 1|X_i = 1) = \frac{\theta(1-\delta)}{\theta(1-\delta)+(1-\theta)\gamma}$, $b = P(Z_i = 1|X_i = 0) = \frac{\theta\delta}{\theta\delta+(1-\theta)(1-\gamma)}$, and $s = \sum_{i \in e}(x_i)$, and m_k = length of e; the number of aberrations in the e.
Otherwise, when e is not a leaf edge, and has parent edge PA(e),

$$\tilde{p}_e(x) = P(\text{branch } e \text{ is open}|X = x) = p_e \left[1 - \prod_{h:PA(h)=e} \{1 - \tilde{p}_h(x)\}\right].$$

Thus using tree-like network, three rate parameters, θ, γ and δ, and Bayes rule, we can get the joint distribution $p(x)$ for the aberration profile $x = (x_1, \ldots, x_n)$.

$$p(x) = P(X = x|\text{SEL}) = \frac{P(x)P(\text{SEL}|x)}{P(\text{SEL})} = \alpha^{\sum_i x_i}(1-\alpha)^{\sum_i (1-x_i)} \left\{ \frac{1 - \prod_{e:\text{PA}(e)=\text{root}}[1 - \tilde{p}_e(x)]}{1 - \prod_{e:\text{PA}(e)=\text{root}}[1 - \tilde{p}_e]} \right\}$$

4 Instability Selection General Network model

In ISGN, the oncogenetic network is defined by $\mathcal{C} = \{C_1, \ldots, C_K\}$, and each ensemble $C_k, (k = 1, \ldots, K)$ can be any set of aberrations. We do require that no ensemble is the subset of the other, i.e., $C_i \not\subset C_j$ $(i, j = 1, \ldots, K)$.

To get the probability mass function in ISGN, again we use three parameter values, θ, γ and δ. As a definition, SEL happens if all aberrations in at least one ensemble occur in a tumor progenitor cell. $P(\text{SEL})$ evaluation is feasible using the inclusion-exclusion approach, i.e.

$$P(\text{SEL}) = \sum_{k=1}^{K} P(A_k) - \sum_{k=1}^{K-1} \sum_{l=(k+1)}^{K} P(A_k \cap A_l) + \ldots + (-1)^{K-1} P(A_1 \cap \ldots A_K).$$

here, $P(A_k) = P\big(\bigcap_{i \in C_k} [Z_i = 1]\big) = \theta^{s_k}$, and $s_k = \sum_{i \in C_k} X_i$. Similarly we can derive $P(\text{SEL}|X = x)$:

$$P(\text{SEL}|X = x) = \sum_{k=1}^{K} P(B_k) - \sum_{k=1}^{K-1} \sum_{l=(k+1)}^{K} P(B_k \cap B_l) + \ldots + (-1)^{K-1} P(B_1 \cap \ldots B_K).$$

here $P(B_k) = P\big(\bigcap_{i \in C_k} [Z_i = 1|X_i]\big) = a^{m_k} b^{m_k - s_k}$, $a = P(Z_i = 1|X_i = 1) = \frac{\theta(1-\delta)}{\theta(1-\delta)+(1-\theta)\gamma}$, $b = P(Z_i = 1|X_i = 0) = \frac{\theta\delta}{\theta\delta+(1-\theta)(1-\gamma)}$, $s_k = \sum_{i \in C_k} X_i$ and $m_k =$ the number of aberrations in the C_k. Using Bayes rule, the joint distribution for one tumor $x = (x_1, \ldots, x_n)$ is

$$\begin{aligned} p(x) &= P(X = x|\text{SEL}) = \frac{P(x)P(\text{SEL}|x)}{P(\text{SEL})} \\ &= \alpha^{\sum_i x_i}(1-\alpha)^{\sum_i (1-x_i)} \left\{ \frac{\sum_{k=1}^{K} P(B_k) - \sum_{k=1}^{K-1} \sum_{l=(k+1)}^{K} P(B_k \cap B_l) + \ldots + (-1)^{K-1} P(B_1 \cap \ldots B_K)}{\sum_{k=1}^{K} P(A_k) - \sum_{k=1}^{K-1} \sum_{l=(k+1)}^{K} P(A_k \cap A_l) + \ldots + (-1)^{K-1} P(A_1 \cap \ldots A_K)} \right\} \end{aligned}$$

5 ISN

To get the posterior distribution, MCMC method is used and functions are implemented in ISN package. The ISN package can be loaded by

```
> library(ISN)
```

The main functions available in ISN are:

Functions related with ISTN

Trisn	sample random binary vectors from a tree-like network using rejection sampling.
Tnet	simulate a tree-like network
Tlik	calculate the log joint probability density of aberrations in the ISTN model.
ISTN	fit the ISTN model using MCMC
Tcluster	conduct a hierarchical clustering
Tdraw	draw a tree-like network

Functions related with ISGN

Grisn	sample random binary vectors from a general network using rejection sampling.
Gnet	simulate a general network
Glik	calculate the log joint probability density of aberrations in the ISGN model.
ISGN	fit the ISGN model using MCMC
Gscore	scoring a network using 4 different methods
Ginter	generate interactions
Ginternum	plot the number of interactions

ISTN and ISGN methods are developed especially to analyze CGH data. CGH data record chromosome gain or loss using binary variable. 0 indicates no observed aberration and 1 indicates observed aberration. One example of CGH data is Renal cell carcinoma (RCC) profile obtained by F. Jiang and H. Moch (2000).
The data can be read in by

```
> data(rcc)
> rcc2 <- rcc$data
> nrow(rcc2)

[1] 116
```

plot.matrix function shows full data. In Figure 1, each dark shaded squares indicates an observed aberration.

Here instead of analyzing the RCC data, we use a small simulated data to introduce the functions in ISN.
Trisn function generates a data set from a tree-like network. Following examples generates 100 data from 20 aberrations {a,b,...,t}. Note that among 20 aberrations, 6 are relevant.

```
> para <- c(0.05, 0.01, 0.01)
> partition <- c(1, 2, 3, 4, 5, 6, rep(1, 14))
> parent <- c(0, 0, 1, 1, 2, 2, rep(0, 14))
```

```
> plot.matrix(rcc2, xlab = "Aberrations", ylab = "RCC Tumor ID")
```

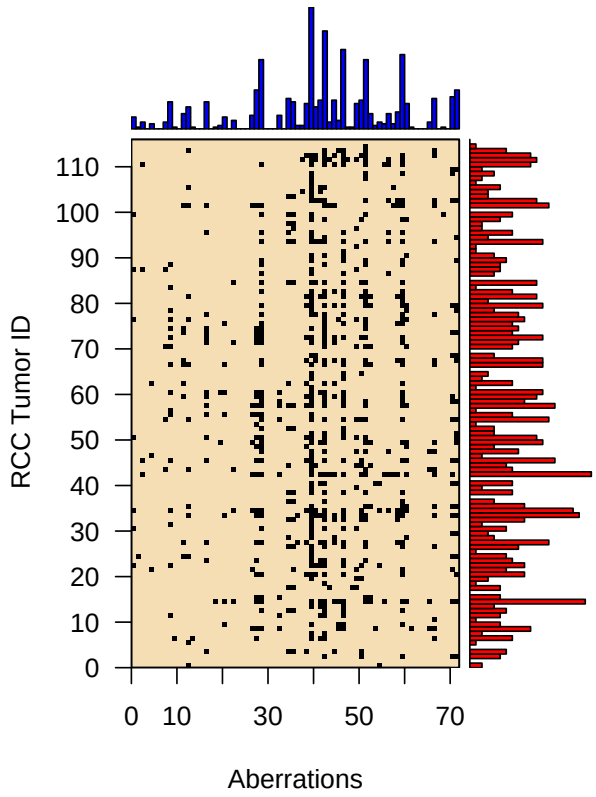

Figure 1: Renal Cell Carcinomas (RCC) cCGH data collected from 116 tumors by Jiang and Moch (Jiang *et al.*, 2000).

```
> neutral <- c(rep(0, 6), rep(1, 14))
> numtumor <- 100
> data1 <- Trisn(para, partition, parent, neutral, numtumor)
> colnames(data1) <- letters[1:length(partition)]
> data1[1, ]

a b c d e f g h i j k l m n o p q r s t
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
```

Tdraw function plot the tree-like network. It returns two figures: top one shows a tree-like network and bottom one shows number of tumors experiencing each ensemble. Using previous tree-like network and 100 simulated data, Figure 2 draws the network (top) and shows the number of tumors experiencing each edge and ensemble (bottom).
Usually to fit the ISTN model, we select some significant aberrations and run MCMC using selected aberrations. Function ISTN is a main function implementing MCMC. It saves one network in every nskip scans, and saves total nsave networks. Usually the MCMC run is longer than this example. Function ISTN returns the posterior probabilities of nsave networks along with the maximum likelihood network.

```
> sdata <- data1[, c(1:6)]
> mcmc <- list(nsave = 500, nskip = 10, nperm = 3, padI = 0.15,
+     padII = 0.15)
> res1 <- res(6)
> para <- c(0.05, 0.01, 0.01)
> mh <- ISTN(sdata, mcmc = mcmc, tau = 4, para, res1)
> mh$mtree

a b e
4 4 3
```

To draw the maximum log-likelihood network among nsave networks, again function Tdraw is used. Figure 3 shows MLE tree-like network.

Another way to summarize the posterior probabilities is conducting a hierarchical clustering. Figure 5 shows hierarchical clustering using nsave networks.

```
> Tdraw(partition, neutral, parent, data1, line1 = 1, num1 = 1,
+     letter1 = 1, main1 = "Tree-like Network (true network)",
+     main2 = "Number of tumors experiencing each edge/ensemble")
```

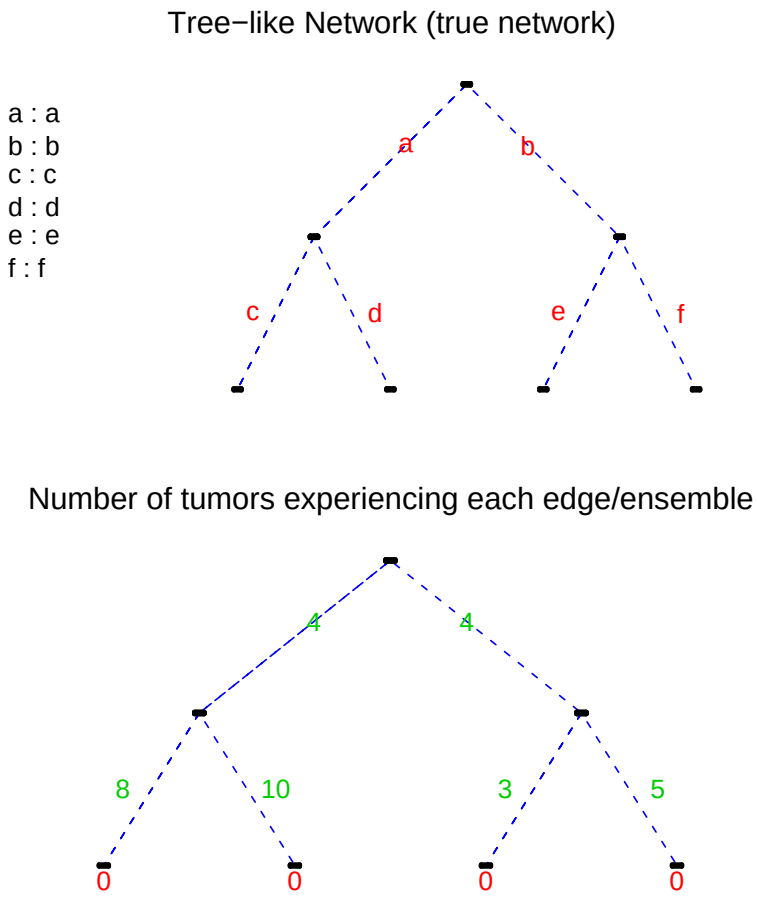

Figure 2: True tree-like network.

```
> index <- c(1:mcmc$nsave)
> i <- index[max(mh$loglik) == mh$loglik][1]
> Tdraw(mh$partition[i, ], mh$neutral[i, ], mh$parent[i, ], sdata,
+     line1 = 1, num1 = 1, letter1 = 1, main1 = "MLE tree-like network",
+     main2 = "Number of tumors experiencing each edge/ensemble")
```

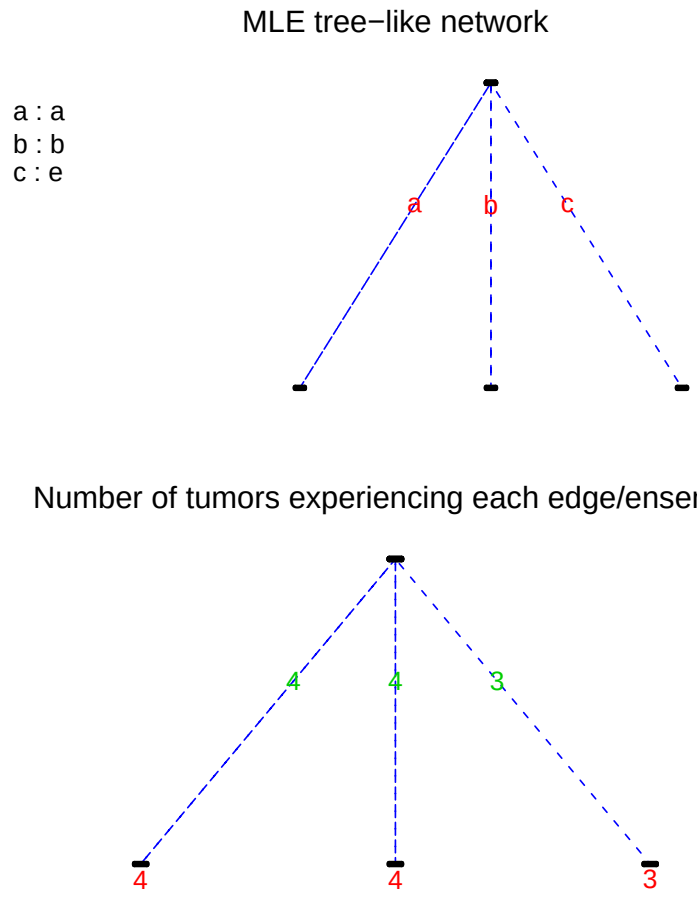

Figure 3: MLE tree-like network.

```
> Tcluster(mh, colnames(sdata))
```

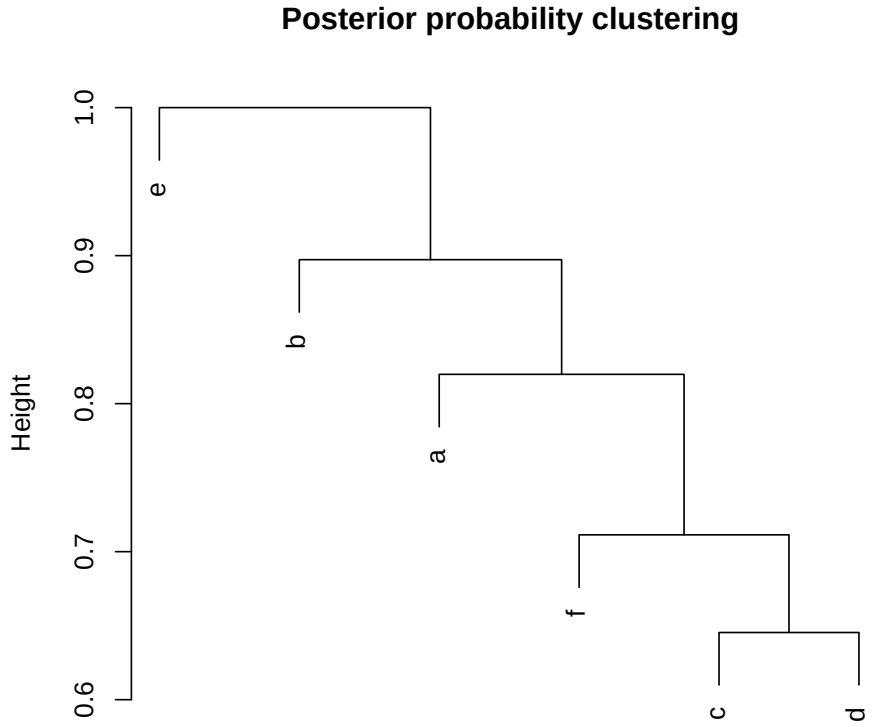

Figure 4: Hierarchical clustering.

Let us look at functions use in ISGN model.

Function Ginternum shows the number of single, pair, triple and quadruple interactions in RCC data.

```
> Ginternum(rcc2)
```

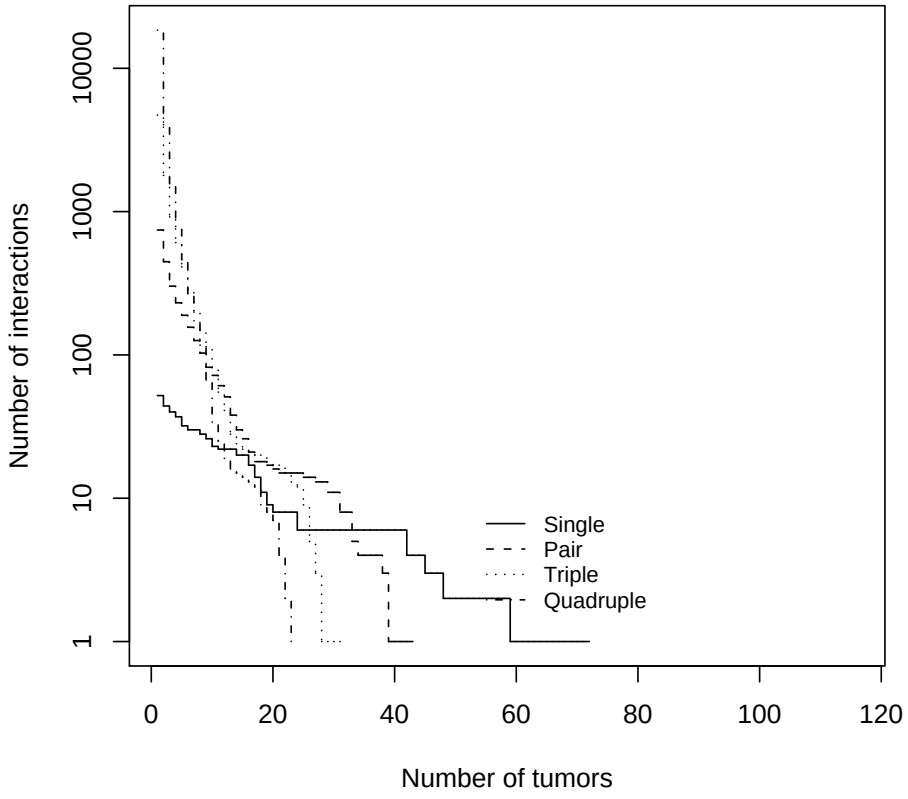

Figure 5: Hierarchical clustering.

To generate data from a general network function Grisn is used. Following example generates 100 data from 20 aberrations {a,b,. . .,t}.

```
> para <- c(0.05, 0.99, 0.01)
> numtumor <- 100
> C <- matrix(0, 3, 20)
> C[1, ] <- c(1, 1, 1, 0, 0, rep(0, 15))
> C[2, ] <- c(0, 0, 1, 1, 1, rep(0, 15))
> C[3, ] <- c(1, 1, 0, 0, 1, rep(0, 15))
> data1 <- Grisn(para, C, numtumor)
```

```
> colnames(data1) <- letters[1:ncol(C)]
> data1[1, ]

a b c d e f g h i j k l m n o p q r s t
0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

To analyze the simulated data using ISGN model, first, one needs to generate a candidate set. Function Gcandi is used to generate a candidate set from data.

```
> pval1 <- 0.1
> pval2 <- 0.1
> error <- 0.1
> candid <- Gcandi(data1, pval1, pval2, error)
> nrow(candid)

[1] 16
```

Using uniform prior, function ISGN runs MCMC. This example runs a short chain but it is recommended to run a long chain. Function ISGN returns nsave general networks along with maximum a posterior (MAP) network.

```
> mcmc <- list(padI = 0.01, nsave = 500, nskip = 10, tmax = 4)
> pri <- rep(0, 20)
> a <- ISGN(data1, mcmc, para, pri, candid)
> a$mnetwork

c("a", "b", "c") c("a", "b", "e") c("c", "d", "e")
              17               18               23
```

To see how well a network explains the data, function Gscore calculates the score based on 4 different method.

```
> Gscore(data1, a$bestnet)

 score  adj.s  cov.s dist.s
 0.875  1.472  0.058  3.103
```

6 References

1. Jiang, F., Desper, R., Papadimitriou, C.H., Schaffer, A.A., Kallioniemi, O.P., Richter, J., Schraml, P., Sauter, G., Mihatsch, M.J., Moch, H.(2000), Construction of evolutionary tree models for renal cell carcinoma from comparative genomic hybridization data, *Cancer Research*, **60**, 6503-6509.

2. Newton, M.A.(2002), Discovering combinations of Genomic aberrations associated with cancer, *Journal of the American Statistical Association*, **97**, 931-942.

3. Newton, M.A., Yang, H. Gorman, P., Tomlinson, T., Roylance, R.(2003), A statistical approach to modeling genomic aberrations in cancer cells (with discussion) *Bayesian statistics*, **7**, ed. Bernardo, J.M., Bayarri, M.J., Berger, J.O, Dawid, A.P., Hecherman, D., Smith, A.F.M., West, M., Oxford University Press.

4. Yang, H(2005), Model-based clustering of genomic aberrations: Generalizing the instability selection network model, Doctoral Dissertation, *Department of Statistics, UW-Madison*.