Day-2 Practical Session, 26 May 2021

Part 2: BIGS-IWE strategy for Line Intercept Sampling (LIS)

Li-Chun Zhang[1,2,3] and *Melike Oguz-Alper*[2]

[1]*University of Southampton (L.Zhang@soton.ac.uk)*, [2]*Statistics Norway*, [3]*University of Oslo*

In this illustration, we will apply BIGS-IWE strategy to line-intercept sampling (LIS) which is a method of habitat sampling in a given area, where a habitat is sampled if a chosen line segment transects it. An example of LIS is given by Becker (1991). The sketch of the observed tracks can be obtained by compiling R-function **skthLISBecker**. Visualisation of the BIG constructed based on the observed line-intercept samples can be obtained by using R-function **skthLISBeckerBIG**.

Description of the population and sampling strategies

- Population BIG: $\mathcal{B} = (F, \Omega; H)$, where Ω consists of all wolverine tracks in the region of interest, F contains the corresponding projection segments, and H consist of edges between the tracks and the segments, i.e. $(i\kappa) \in H$ for $i \in F$ and $\kappa \in \Omega$
  - **NB**. $\mathcal{B}$ cannot be consructed unless the whole area observed
- Observed BIG: $\mathcal{B}^* = (F^*, \Omega_s; H^*)$, where Ω_s contains the observed wolverine tracks, F^* contain the projection segments constructed based on the actual samples s and Ω_s, and H^* consists of the incident observational links from F^* to Ω_s
- β^*_κ: *ancestry* set of $\kappa \in \Omega_s$ in $\mathcal{B}^*$ and α^*_i: *successors* of $i \in F^*$ in $\mathcal{B}^*$
- Four systematic samples, A, B, C and D, each containing three positions drawn on the baseline. Equal distances of 12 miles between each position in a given draw
- Four wolverine tracks observed: $\Omega_s = \{\kappa_1, \kappa_2, \kappa_3, \kappa_4\}$; $y_\kappa = (1, 2, 2, 1)^\top$ and $L_\kappa = (5.25, 7.5, 2.4, 7.05)^\top$
- The baseline divided into seven projection segments given the tracks: $F^* = \{i_1, i_2, i_3, i_4, i_5, i_6, i_7\}$
- The probability that the ith segment selected under systematic sampling: $p_i = x_i/12$, where $\mathbf{x} = (5.25, 2.25, x_3, 2.4, x_5, 7.05, x_7)^\top$
- We have $s_1 = s_2 = \{i_1, i_5, i_6\}$, yielding $\Omega_s = \{\kappa_1, \kappa_2, \kappa_4\}$, and $s_3 = s_4 = \{i_4, i_6, i_7\}$, yielding $\Omega_s = \{\kappa_3, \kappa_4\}$

Formula sheet

- The parameter of interest: total number of wolverines in the region of interest

  $\theta = \sum_{\kappa \in \Omega} y_\kappa$, where y_κ number of wolverines in track κ

- Hansen-Hurwitz (HH) type estimators on the rth draw

  $\hat{\theta}_r = \sum_{i \in s_r} \frac{z_i}{p_i}$, where $z_i = \sum_{\kappa \in \alpha^*_i} w_{i\kappa} y_\kappa$

  - *Multiplicity* estimator; equal weights

    $w_{i\kappa} \equiv \frac{1}{|\beta^*_\kappa|}$

  - HH-type estimator with *unequal* weights: *probability and inverse degree-adjusted (PIDA) weights*

    $w_{i\kappa} = \frac{p_i}{|\alpha^*_i|^\gamma} \left(\sum_{i \in \beta^*_\kappa} \frac{p_i}{|\alpha^*_i|^\gamma} \right)^{-1}, \gamma \geq 0$

NB. When $\gamma = 0$, PIDA weights reduce to $w_{i\kappa} = p_i / p_{(\kappa)}$, where $p_{(\kappa)} = \sum_{i \in \beta^*_\kappa} p_i$

- HH-type estimators over all the draws

  $\hat{\theta} = \frac{1}{4} \sum_{r=1}^{4} \hat{\theta}_r$

- Variance estimator of the HH-type estimators over all the draws

  $\hat{\mathrm{V}}(\hat{\theta}) = \frac{1}{4} \frac{\sum_{r=1}^{4} (\hat{\theta}_r - \hat{\theta})^2}{r-1}$

NB. R-package **igraph** has to be installed before running R-functions below that generates bipartite graphs.

Description of R-function **skthLISBeckerBIG**

1. Function parameters

- **showplot**: Use **TRUE** to get the skecth of BIG; default **FALSE**

2. Main steps of the function

- A bipartite graph constructed based on the observed wolverine tracks and the sample line segments transecting them. R-package **igraph** used to generate the graph

3. Main outputs of the function

- BIG plot shown if **showplot**= **TRUE**
- The bipartite graph generated is returned as a graph object. It shall be called via **$G**

Description of R-function **zLISBecker**

1. Function parameters

- **graphstar**: graph to be used: the output of **skthLISBeckerBIG**
- **coefgamma**: coefficient to be used in the HH-type estimator with PIDA weights; default value 0. No effect of the choice if **multiplicity**= **TRUE**
- **probi**: a vector of the selection probabilities of the constructed projection segments
- **multiplicity**: Use **TRUE** to get z_i values based on equal weights, i.e. $w_{i\kappa} = \mid \beta^*_\kappa \mid^{-1}$; default **FALSE**

2. Main steps of the function

- Edge set derived from the input graph, as well as the labels of the vertices in F^* and Ω_s
- $|\alpha^*_i|$ and $|\beta^*_\kappa|$ calculated based on the edge set
- z_i values calculated for all $i \in F^*$ for chosen values of γ

3. Main outputs of the function

- z_i values returned

Description of R-function **mainLISBecker**

1. Function parameters

- **graphstar**: graph to be used: the output of **skthLISBeckerBIG**
- **coefgamma**: coefficient to be used in the HH-type estimator with PIDA weights; default value 0. No effect of the choice if **multiplicity**= **TRUE**
- **probi**: a vector of the selection probabilities of the constructed projection segments
- **multiplicity**: Use **TRUE** to get z_i values based on equal weights, i.e. $w_{i\kappa} = \mid \beta^*_\kappa \mid^{-1}$; default **FALSE**
- **showcat**: Use **FALSE** to avoid printing outputs from the *cat* function in R; default **TRUE**

2. Main steps of the function

- Edge set derived from the input graph
- $|\alpha^*_i|$ and $|\beta^*_\kappa|$ calculated based on the edge set
- z_i values obtained by calling function **zLISBecker**
- For each draw under systematic sampling, estimates obtained by using the HH-type estimator.
- The HH-type estimator over all draws applied by taking the average of the estimates. Variance of the estimator calculated

3. Main outputs of the function

- An estimate for the total number of wolverine tracks in the given area and its estimated variance. Variance estimate can be called via **$varest** for further analysis.