Analysis for Galton-Watson Harris paths

Romain Azaïs, Alexandre Genadot, and Benoît Henry

AGH is a Matlab toolbox available at `agh.gforge.inria.fr`, mainly developed by Benoît Henry in Spring and Summer 2016 that allows to easily compute estimators of the relative scale of trees. These estimators have been introduced for Galton-Watson trees conditioned on their number of nodes but may be computed for any ordered tree. The theoretical study of these estimators is presented in the paper [1] which should be consulted in parallel. Another estimator based on the paper [2] by Bharath *et al.* may also be calculated.

User documentation

Class aghTree

Description

This class is designed to create and manipulate tree structured data.

Methods

- **aghTree**

  Description

  This function is designed to create instances of the class aghTree.

  Usage

  To create a tree in a variable T, the function takes the form:

  ```
  T=aghTree(...)
  ```

  Arguments

  - `T=aghTree('ConditionnedGW',n,p)`
    is an optional set of arguments which allows to simulate a conditioned Galton-Watson tree with a given size n and a distribution p. Remark that n is an integer and p is a vector such that $p(i)$ is the probability to have $i-1$ children.
    Example:

    ```
    T=aghTree('ConditionnedGW',200,[0.2 0.6 0.2])
    ```

  - `T=aghTree('GaltonWatson',p)`
    is an optional set of arguments which allows to simulate a Galton-Watson tree with distribution p. As before p is a vector.
    Example:

```
T=aghTree('GaltonWatson',[0.6 0.4])
```

- `T=aghTree('childsList',L)`
  is an optional set of arguments which allows to create a tree from the list of children of each node in the tree (ordered in depth-first order). L is a vector of integer such that $L(1)$ is the number a children of the root of the tree, $L(2)$ is the number of children of the first child of the root (in depth-first order) and so on.
  Example:

```
T=aghTree('ConditionnedGW',200,[0.2 0.6 0.2])
L=T.getChildList
U=aghTree('childsList',L)
```

- `T=aghTree('Contour',H)`
  is an optional set of arguments which allows to create a tree from its Harris path. H is a vector of integers of size $2n+1$ such that n is the number of nodes of the tree and $H(i)$ is the value of the Harris path at time $i-1$.
  Example:

```
T=aghTree('ConditionnedGW',200,[0.2 0.6 0.2])
H=T.getContour
U=aghTree('Contour',H)
```

- `T=aghTree('Webpage',URL)`
  is an optional set of arguments which allows to create a tree for an HTML file stored on the world wide web. `URL` is a string containing the URL of the webpage.
  Example:

```
T=aghTree('Webpage','https://en.wikipedia.org/wiki/Main_Page')
```

Value

Return an instance of the class aghTree.

- **getLeafNumber**

  Description

  This function returns the number of leaves in a given tree `T`.

  Usage

```
T.getLeafNumber
```

  Arguments

  none

  Value

  An integer giving the number of leaves in a tree.

- **getOuterDegree**

  Description

  This function returns the outer degree of a given tree `T`.

  Usage

  T.getOuterDegree

  Arguments

none

Value

An integer giving the outer degree of a tree.

- **getHeight**

  Description

  A function which returns the height of a tree.

  Usage

  ```
  T.getHeight
  ```

  Arguments

  none

  Value

- **getChild**

  Description

  This function returns a subtree attached to the root.

  Usage

  ```
  T.getChild(i)
  ```

  Arguments

  – An integer `i` corresponding to the index of the desired subtree among the children of the root.

  Value

  An instance of the class aghTree.

- **getContour**

  Description

  This function allows to get the Harris path of a given tree `T`.

  Usage

  ```
  T.getContour
  ```

  Arguments

  none

  Value

  Return a vector `H` such that `H(i)` is the value of the Harris path at point $i - 1$.

- **plotContour**

  Description

  This function allows to plot the Harris path of a given tree `T`.

  Usage

`T.plotContour`

Arguments

none

Value

none

- **getChildList**

  Description

  This function returns a vector of containing the number of child of each node in the tree.

  Usage

  `T.getChildList`

  Arguments

  none

  Value

  A vector containing the number of children of each node in the tree ordered in preorder.

Class aghForest

Description

This class is the main class of the toolbox. It allows to perform estimation on forest of trees.

Methods

- **aghForest**

  Description

  This function is designed to create instances of the class aghForest.

  Usage

  To create a forest in a variable `F`, the function takes the form:

  `F=aghForest(...)`

  Arguments

  - `F=aghForest('FixedSizeConditionedGW','TreesSize',n,'ForestSize',f,'Distribution',p)` is an optional set of arguments which allows to simulate a forest of f conditioned Galton-Watson tree with a given size n and a birth distribution p. Remark that n is an integer and p is a vector such that $p(i)$ is the probability to have $i-1$ children.
    Example:

    ```
    p=[0.2 0.6 0.2] ;
    n=20 ;
    f= 10 ;
    F=aghForest('FixedSizeConditionedGW','TreesSize',n,'ForestSize',f,'Distribution',p)
    ```

– `F=aghForest('List',P)`
is an optional set of arguments which allows create of forest from a vector P of trees.
Example:

```
T1=aghTree('ConditionnedGW',200,[0.2 0.6 0.2])
T2=aghTree('ConditionnedGW',100,[0.1 0.7 0.2])
P=[T1 T2]
F=aghForest('List',P)
```

Value

Return an instance of the class aghForest.

- **estimeScale**

Description

This function allows to compute the estimator $\widehat{\lambda}_{ls}[\mathcal{F}]$ or $\widehat{\lambda}_{W}[\mathcal{F}]$ from a forest $\mathcal{F}$.

Usage

```
F.estimeScale(...)
```

Arguments

– `F.estimeScale('LSE')`
is an of arguments which allows to compute the estimator $\widehat{\lambda}_{ls}[\mathcal{F}]$. Remark that you can add the optional argument `'Elementswise'` which allows to obtain a vector containing the quantities $\widehat{\lambda}[\tau]$ for each tree τ in the forest.
– `F.estimeScale('Wasserstein')`
is an of arguments which allows to compute the estimator $\widehat{\lambda}_{W}[\mathcal{F}]$.
– `F.estimeScale('KPDRV')`
is an argument which allows to compute the estimator given in [2].

Value

A real number or a vector of real numbers.

- **printForest**

Description

This function allows to plot the contour path of a given forest `F`.

Usage

```
F.printForest
```

Arguments

none

Value

none

- **getForestSize**

Description

A function which returns the size of a forest.

Usage

`F.getForestSize`

Arguments

none

Value

An integer.

- **getTree**

  Description

  A function which returns a specific tree in a forest.

  Usage

  `F.getTree(...)`

  Arguments

  An integer i corresponding to the index of the tree in the forest.

  Value

  An instance of the class aghTree.

References

[1] Azaïs, R., Genadot, A., and Henry, B. Inference for conditioned Galton-Watson trees from their Harris path. *Preprint* (2016).

[2] Bharath, K., Kambadur, P., Dey, D., Arvin, R., and Baladandayuthapani, V. Inference for large tree-structured data. *Preprint* (2014).