Stochastic chains with memory of variable length

Antonio Galves

Universidade de São Paulo

2009, Année de la France au Brésil

Stochastic modeling and linguistic rhythm retrieval from written texts

1. A discussion about stochastic modeling
2. The model is a stochastic chain with memory of variable length
3. A linguistic case study
4. Joint work with Charlotte Galves, Nancy Garcia and Florencia Leonardi.

Stochastic modeling and linguistic rhythm retrieval from written texts

1. A discussion about stochastic modeling
2. The model is a stochastic chain with memory of variable length
3. A linguistic case study
4. Joint work with Charlotte Galves, Nancy Garcia and Florencia Leonardi.

Stochastic modeling and linguistic rhythm retrieval from written texts

1. A discussion about stochastic modeling
2. The model is a stochastic chain with memory of variable length
3. A linguistic case study
4. Joint work with Charlotte Galves, Nancy Garcia and Florencia Leonardi.

Stochastic modeling and linguistic rhythm retrieval from written texts

1. A discussion about stochastic modeling
2. The model is a stochastic chain with memory of variable length
3. A linguistic case study
4. Joint work with Charlotte Galves, Nancy Garcia and Florencia Leonardi.

Linguistic motivation

- A long standing conjecture says that Brazilian Portuguese (BP) and European Portuguese (EP) implement different *rhythms*.
- But there is no satisfactory formal notion of linguistic rhythm.
- This is a challenging and important problem in linguistics.
- Even more difficult: we want to retrieve rhythmic patterns looking only to written texts of BP and EP!!!

Linguistic motivation

- A long standing conjecture says that Brazilian Portuguese (BP) and European Portuguese (EP) implement different *rhythms*.
- But there is no satisfactory formal notion of linguistic rhythm.
- This is a challenging and important problem in linguistics.
- Even more difficult: we want to retrieve rhythmic patterns looking only to written texts of BP and EP!!!

Linguistic motivation

- A long standing conjecture says that Brazilian Portuguese (BP) and European Portuguese (EP) implement different *rhythms*.
- But there is no satisfactory formal notion of linguistic rhythm.
- This is a challenging and important problem in linguistics.
- Even more difficult: we want to retrieve rhythmic patterns looking only to written texts of BP and EP!!!

Linguistic motivation

- A long standing conjecture says that Brazilian Portuguese (BP) and European Portuguese (EP) implement different *rhythms*.
- But there is no satisfactory formal notion of linguistic rhythm.
- This is a challenging and important problem in linguistics.
- Even more difficult: we want to retrieve rhythmic patterns looking only to written texts of BP and EP!!!

A few facts about BP and EP

- BP and EP share the same lexicon
- Even if they have different syntaxes, BP and EP produce a great number of superficially identical sentences
- We are looking for a needle in a haystack!

A few facts about BP and EP

- BP and EP share the same lexicon
- Even if they have different syntaxes, BP and EP produce a great number of superficially identical sentences
- We are looking for a needle in a haystack!

A few facts about BP and EP

- BP and EP share the same lexicon
- Even if they have different syntaxes, BP and EP produce a great number of superficially identical sentences
- We are looking for a needle in a haystack!

How to retrieve evidences that BP and EP have different rhythms?

- Recipe:
- Get samples of BP and EP rhythmic sequences
- Find a good class of models for these samples
- See if the models which best fit BP and EP samples coincide.

How to retrieve evidences that BP and EP have different rhythms?

- Recipe:
- Get samples of BP and EP rhythmic sequences
- Find a good class of models for these samples
- See if the models which best fit BP and EP samples coincide.

How to retrieve evidences that BP and EP have different rhythms?

- Recipe:
- Get samples of BP and EP rhythmic sequences
- Find a good class of models for these samples
- See if the models which best fit BP and EP samples coincide.

How to retrieve evidences that BP and EP have different rhythms?

- Recipe:
- Get samples of BP and EP rhythmic sequences
- Find a good class of models for these samples
- See if the models which best fit BP and EP samples coincide.

Getting samples of BP and EP rhythmic sequences

- The data we analyzed is an encoded corpus of newspaper articles.
- This corpus contains all the 365 editions of the years 1994 and 1995 from the daily newspapers *Folha de São Paulo* (Brazil) and *O Público* (Portugal).

Encoding hypothetical rhythmic features

We encode the words by assigning one of four symbols to each syllable according to whether

(i) it is stressed or not;

(ii) it is the beginning of a prosodic word or not.

By *prosodic word* we mean a lexical word together with the functional non stressed words which precede it.

A five symbols alphabet

This double 0-1 classification can be represented by the four symbols alphabet $\{0, 1, 2, 3\}$ where

- $0 =$ non-stressed, non prosodic word initial syllable;
- $1 =$ stressed, non prosodic word initial syllable;
- $2 =$ non-stressed, prosodic word initial syllable;
- $3 =$ stressed, prosodic word initial syllable.

Additionally we assign an extra symbol (4) to encode the end of each sentence. We call $A = \{0, 1, 2, 3, 4\}$ the alphabet obtained in this way.

An example

Example: "O menino já comeu o doce" (The boy already ate the candy)

Sentence	O	me	ni	no	já	co	meu	o	do	ce	.
Code	2	0	1	0	3	2	1	2	1	0	4

Modeling samples of symbolic sequences

- The encoding described above produced sequences taking values in the alphabet A.
- At first sight we can't see any kind of regular (deterministic) behavior in these sequences.
- Apparently the same subsequences may appear in BP and EP texts.
- What can be a model for these sequences?
- Answer: use a probability measure on the set of infinite sequences of symbols in the alphabet A.

Modeling samples of symbolic sequences

- The encoding described above produced sequences taking values in the alphabet A.
- At first sight we can't see any kind of regular (deterministic) behavior in these sequences.
- Apparently the same subsequences may appear in BP and EP texts.
- What can be a model for these sequences?
- Answer: use a probability measure on the set of infinite sequences of symbols in the alphabet A.

Modeling samples of symbolic sequences

- The encoding described above produced sequences taking values in the alphabet A.
- At first sight we can't see any kind of regular (deterministic) behavior in these sequences.
- Apparently the same subsequences may appear in BP and EP texts.
- What can be a model for these sequences?
- Answer: use a probability measure on the set of infinite sequences of symbols in the alphabet A.

Modeling samples of symbolic sequences

- The encoding described above produced sequences taking values in the alphabet A.
- At first sight we can't see any kind of regular (deterministic) behavior in these sequences.
- Apparently the same subsequences may appear in BP and EP texts.
- What can be a model for these sequences?
- Answer: use a probability measure on the set of infinite sequences of symbols in the alphabet A.

Modeling samples of symbolic sequences

- The encoding described above produced sequences taking values in the alphabet A.
- At first sight we can't see any kind of regular (deterministic) behavior in these sequences.
- Apparently the same subsequences may appear in BP and EP texts.
- What can be a model for these sequences?
- Answer: use a probability measure on the set of infinite sequences of symbols in the alphabet A.

Chains with memory of variable length

- Introduced by Rissanen (1983) as a universal system for data compression.
- He called this model a *finitely generated source* or a *tree machine*.
- Statisticians call it *variable length Markov chain* (Bühlman and Wyner 1999).
- Also called *prediction suffix tree* in bio-informatics (Bejerano and Yona 2001).

Chains with memory of variable length

- Introduced by Rissanen (1983) as a universal system for data compression.
- He called this model a *finitely generated source* or a *tree machine*.
- Statisticians call it *variable length Markov chain* (Bühlman and Wyner 1999).
- Also called *prediction suffix tree* in bio-informatics (Bejerano and Yona 2001).

Chains with memory of variable length

- Introduced by Rissanen (1983) as a universal system for data compression.
- He called this model a *finitely generated source* or a *tree machine*.
- Statisticians call it *variable length Markov chain* (Bühlman and Wyner 1999).
- Also called *prediction suffix tree* in bio-informatics (Bejerano and Yona 2001).

Chains with memory of variable length

- Introduced by Rissanen (1983) as a universal system for data compression.
- He called this model a *finitely generated source* or a *tree machine*.
- Statisticians call it *variable length Markov chain* (Bühlman and Wyner 1999).
- Also called *prediction suffix tree* in bio-informatics (Bejerano and Yona 2001).

Heuristics

- When we have a symbolic chain describing
- a syntatic structure,
- a prosodic contour,
- a DNA sequence,
- a protein,....
- it is natural to assume that each symbol depends only on a **finite suffix** of the past
- whose **length depends on the past**.

Heuristics

- When we have a symbolic chain describing
- a syntatic structure,
- a prosodic contour,
- a DNA sequence,
- a protein,....
- it is natural to assume that each symbol depends only on a **finite suffix** of the past
- whose **length depends on the past**.

Heuristics

- When we have a symbolic chain describing
- a syntatic structure,
- a prosodic contour,
- a DNA sequence,
- a protein,....
- it is natural to assume that each symbol depends only on a **finite suffix** of the past
- whose **length depends on the past**.

Heuristics

- When we have a symbolic chain describing
- a syntatic structure,
- a prosodic contour,
- a DNA sequence,
- a protein,....
- it is natural to assume that each symbol depends only on a **finite suffix** of the past
- whose **length depends on the past**.

Heuristics

- When we have a symbolic chain describing
- a syntatic structure,
- a prosodic contour,
- a DNA sequence,
- a protein,....
- it is natural to assume that each symbol depends only on a **finite suffix** of the past
- whose **length depends on the past**.

Heuristics

- When we have a symbolic chain describing
- a syntatic structure,
- a prosodic contour,
- a DNA sequence,
- a protein,....
- it is natural to assume that each symbol depends only on a **finite suffix** of the past
- whose **length depends on the past**.

Heuristics

- When we have a symbolic chain describing
- a syntatic structure,
- a prosodic contour,
- a DNA sequence,
- a protein,....
- it is natural to assume that each symbol depends only on a **finite suffix** of the past
- whose **length depends on the past**.

Warning!

- We are not making the usual **markovian assumption**:
- at each step we are under the influence of a suffix of the past whose **length depends on the past itself**.
- Even if it is finite, in general the length of the relevant part of the past is not bounded above!
- This means that in general these are **chains of infinite order**, not Markov chains.

Warning!

- We are not making the usual **markovian assumption**:
- at each step we are under the influence of a suffix of the past whose **length depends on the past itself**.
- Even if it is finite, in general the length of the relevant part of the past is not bounded above!
- This means that in general these are **chains of infinite order**, not Markov chains.

Warning!

- We are not making the usual **markovian assumption**:
- at each step we are under the influence of a suffix of the past whose **length depends on the past itself**.
- Even if it is finite, in general the length of the relevant part of the past is not bounded above!
- This means that in general these are **chains of infinite order**, not Markov chains.

Warning!

- We are not making the usual **markovian assumption**:
- at each step we are under the influence of a suffix of the past whose **length depends on the past itself**.
- Even if it is finite, in general the length of the relevant part of the past is not bounded above!
- This means that in general these are **chains of infinite order**, not Markov chains.

Contexts

- Rissanen called the relevant suffixes of the past **contexts**.
- The set of all contexts should have the **suffix property**: no context is a proper suffix of another context.
- This means that we can identify the end of each context without knowing what happened sooner.
- The suffix property implies that the set of all contexts can be represented as a **rooted tree with finite branches**.

Contexts

- Rissanen called the relevant suffixes of the past **contexts**.
- The set of all contexts should have the **suffix property**: no context is a proper suffix of another context.
- This means that we can identify the end of each context without knowing what happened sooner.
- The suffix property implies that the set of all contexts can be represented as a **rooted tree with finite branches**.

Contexts

- Rissanen called the relevant suffixes of the past **contexts**.
- The set of all contexts should have the **suffix property**: no context is a proper suffix of another context.
- This means that we can identify the end of each context without knowing what happened sooner.
- The suffix property implies that the set of all contexts can be represented as a **rooted tree with finite branches**.

Contexts

- Rissanen called the relevant suffixes of the past **contexts**.
- The set of all contexts should have the **suffix property**: no context is a proper suffix of another context.
- This means that we can identify the end of each context without knowing what happened sooner.
- The suffix property implies that the set of all contexts can be represented as a **rooted tree with finite branches**.

Chains with variable length memory

It is a stationary stochastic chain (X_n) taking values on a finite alphabet A and characterized by two elements:

- The tree of all contexts.
- A family of transition probabilities associated to each context.
- Given a context, its associated transition probability gives the distribution of occurrence of the next symbol immediately after the context.

Chains with variable length memory

It is a stationary stochastic chain (X_n) taking values on a finite alphabet A and characterized by two elements:

- The tree of all contexts.
- A family of transition probabilities associated to each context.
- Given a context, its associated transition probability gives the distribution of occurrence of the next symbol immediately after the context.

Chains with variable length memory

It is a stationary stochastic chain (X_n) taking values on a finite alphabet A and characterized by two elements:

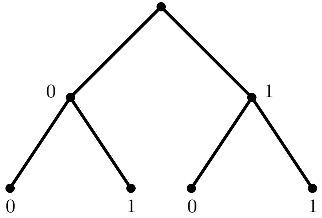
- The tree of all contexts.
- A family of transition probabilities associated to each context.
- Given a context, its associated transition probability gives the distribution of occurrence of the next symbol immediately after the context.

Stochastic chains with variable length memory

For example: if (X_t) is a Markov chain of order 2 on the alphabet $\{0, 1\}$, then

$$\tau = \{00, 01, 10, 11\}.$$

This set can be identified with the tree



Example: the renewal process on $\mathbb{Z}$

$$A = \{0, 1\}$$

$$\tau = \{1, 10, 100, 1000, \ldots\}$$

$$p(1 \mid 0^k 1) = q_k$$

where $0 < q_k < 1$, for any $k \geq 0$, and

$$\sum_{k \geq 0} q_k = +\infty \, .$$

A mathematical question

- Given a probabilistic context tree (τ, p) does it exist at least (at most) one stationary chain (X_n) compatible with it?
- First answer: verify if the infinite order transition probabilities defined by (τ, p) satisfy the sufficient conditions which assure the existence and uniqueness of a chain of infinite order.
- But this is a bad answer: what we really want to know is if there exists a stochastic process having contexts almost surely finite.
- Recently A. Gallo in his PhD dissertation gave sufficient conditions for this.

A mathematical question

- Given a probabilistic context tree (τ, p) does it exist at least (at most) one stationary chain (X_n) compatible with it?
- First answer: verify if the infinite order transition probabilities defined by (τ, p) satisfy the sufficient conditions which assure the existence and uniqueness of a chain of infinite order.
- But this is a bad answer: what we really want to know is if there exists a stochastic process having contexts almost surely finite.
- Recently A. Gallo in his PhD dissertation gave sufficient conditions for this.

A mathematical question

- Given a probabilistic context tree (τ, p) does it exist at least (at most) one stationary chain (X_n) compatible with it?
- First answer: verify if the infinite order transition probabilities defined by (τ, p) satisfy the sufficient conditions which assure the existence and uniqueness of a chain of infinite order.
- But this is a bad answer: what we really want to know is if there exists a stochastic process having contexts almost surely finite.
- Recently A. Gallo in his PhD dissertation gave sufficient conditions for this.

A mathematical question

- Given a probabilistic context tree (τ, p) does it exist at least (at most) one stationary chain (X_n) compatible with it?

- First answer: verify if the infinite order transition probabilities defined by (τ, p) satisfy the sufficient conditions which assure the existence and uniqueness of a chain of infinite order.
- But this is a bad answer: what we really want to know is if there exists a stochastic process having contexts almost surely finite.
- Recently A. Gallo in his PhD dissertation gave sufficient conditions for this.

Back to our case study

- How to assign probabilistic context trees to the samples of BP and EP encoded texts?
- Obvious answer: for each sample choose the one which maximizes the probability of the sample!
- Bad answer: this is just too naive...
- A bigger model will always give a bigger probability to the sample!

Back to our case study

- How to assign probabilistic context trees to the samples of BP and EP encoded texts?
- Obvious answer: for each sample choose the one which maximizes the probability of the sample!
- Bad answer: this is just too naive...
- A bigger model will always give a bigger probability to the sample!

Back to our case study

- How to assign probabilistic context trees to the samples of BP and EP encoded texts?
- Obvious answer: for each sample choose the one which maximizes the probability of the sample!
- Bad answer: this is just too naive...
- A bigger model will always give a bigger probability to the sample!

Back to our case study

- How to assign probabilistic context trees to the samples of BP and EP encoded texts?
- Obvious answer: for each sample choose the one which maximizes the probability of the sample!
- Bad answer: this is just too naive...
- A bigger model will always give a bigger probability to the sample!

A basic statistical question

Given a sample is it possible to estimate the smallest probabilistic context tree generating it ?

A basic statistical question

Given a sample is it possible to estimate the smallest probabilistic context tree generating it ?

In the case of finite context trees, Rissanen (1983) introduced the *algorithm Context* to estimate in a consistent way the probabilistic context tree out from a sample.

The algorithm Context

- Starting with a finite sample $(X_0, \ldots, X_{n-1})$ the goal is to estimate the context at step n.
- Start with a candidate context $(X_{n-k(n)}, \ldots, X_{n-1})$, where $k(n) = \log n$.
- Then decide to shorten or not this candidate context using some *gain function*. For instance the log-likelihood ratio statistics.

The algorithm Context

- Starting with a finite sample $(X_0, \ldots, X_{n-1})$ the goal is to estimate the context at step n.
- Start with a candidate context $(X_{n-k(n)}, \ldots, X_{n-1})$, where $k(n) = \log n$.
- Then decide to shorten or not this candidate context using some *gain function*. For instance the log-likelihood ratio statistics.

The algorithm Context

- Starting with a finite sample $(X_0, \ldots, X_{n-1})$ the goal is to estimate the context at step n.
- Start with a candidate context $(X_{n-k(n)}, \ldots, X_{n-1})$, where $k(n) = \log n$.
- Then decide to shorten or not this candidate context using some *gain function*. For instance the log-likelihood ratio statistics.

Good and bad news

- Recently this algorithm was extended for the case of unbounded trees and its tconsistency was proved by several authors (Csiszar and Talata, Galves and Leonardi, Ferrari and Wyner,...).
- The hidden difficulty: there is always a threshold constant C in the gain function that we use to decide to shorten or not the candidate context.
- For asymptotic consistency results, the specific value of C is irrelevant.
- But if you are an applied statistician and you must select the context tree based on a finite sample, the choice of C matters!

Good and bad news

- Recently this algorithm was extended for the case of unbounded trees and its tconsistency was proved by several authors (Csiszar and Talata, Galves and Leonardi, Ferrari and Wyner,...).
- The hidden difficulty: there is always a threshold constant C in the gain function that we use to decide to shorten or not the candidate context.
- For asymptotic consistency results, the specific value of C is irrelevant.
- But if you are an applied statistician and you must select the context tree based on a finite sample, the choice of C matters!

Good and bad news

- Recently this algorithm was extended for the case of unbounded trees and its tconsistency was proved by several authors (Csiszar and Talata, Galves and Leonardi, Ferrari and Wyner,...).
- The hidden difficulty: there is always a threshold constant C in the gain function that we use to decide to shorten or not the candidate context.
- For asymptotic consistency results, the specific value of C is irrelevant.
- But if you are an applied statistician and you must select the context tree based on a finite sample, the choice of C matters!

Good and bad news

- Recently this algorithm was extended for the case of unbounded trees and its tconsistency was proved by several authors (Csiszar and Talata, Galves and Leonardi, Ferrari and Wyner,...).
- The hidden difficulty: there is always a threshold constant C in the gain function that we use to decide to shorten or not the candidate context.
- For asymptotic consistency results, the specific value of C is irrelevant.
- But if you are an applied statistician and you must select the context tree based on a finite sample, the choice of C matters!

The smallest maximizer criterion

- Assume that the sample was really produced by a probabilistic context tree $(\tau^\star, p^\star)$.
- Consider now the set of candidate context trees maximizing the probability of the sample for each number of *degrees of freedom*.
- It turns out that this sample of champion trees is *totally ordered* and contains the tree $\tau^\star$.
- Moreover, there is a change of regime in the gain of likelihood at $\tau^\star$.
- In the case the tree $\tau^\star$ is bounded this is a rigorous result.
- A similar result for a different class of models was recently pointed out by Massart and co-authors.

The smallest maximizer criterion

- Assume that the sample was really produced by a probabilistic context tree $(\tau^\star, p^\star)$.
- Consider now the set of candidate context trees maximizing the probability of the sample for each number of *degrees of freedom*.
- It turns out that this sample of champion trees is *totally ordered* and contains the tree $\tau^\star$.
- Moreover, there is a change of regime in the gain of likelihood at $\tau^\star$.
- In the case the tree $\tau^\star$ is bounded this is a rigorous result.
- A similar result for a different class of models was recently pointed out by Massart and co-authors.

The smallest maximizer criterion

- Assume that the sample was really produced by a probabilistic context tree $(\tau^\star, p^\star)$.
- Consider now the set of candidate context trees maximizing the probability of the sample for each number of *degrees of freedom*.
- It turns out that this sample of champion trees is *totally ordered* and contains the tree $\tau^\star$.
- Moreover, there is a change of regime in the gain of likelihood at $\tau^\star$.
- In the case the tree $\tau^\star$ is bounded this is a rigorous result.
- A similar result for a different class of models was recently pointed out by Massart and co-authors.

The smallest maximizer criterion

- Assume that the sample was really produced by a probabilistic context tree $(\tau^\star, p^\star)$.
- Consider now the set of candidate context trees maximizing the probability of the sample for each number of *degrees of freedom*.
- It turns out that this sample of champion trees is *totally ordered* and contains the tree $\tau^\star$.
- Moreover, there is a change of regime in the gain of likelihood at $\tau^\star$.
- In the case the tree $\tau^\star$ is bounded this is a rigorous result.
- A similar result for a different class of models was recently pointed out by Massart and co-authors.

The smallest maximizer criterion

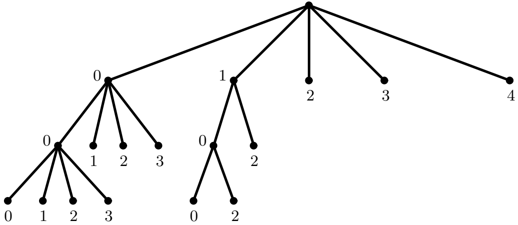
- Assume that the sample was really produced by a probabilistic context tree $(\tau^\star, p^\star)$.
- Consider now the set of candidate context trees maximizing the probability of the sample for each number of *degrees of freedom*.
- It turns out that this sample of champion trees is *totally ordered* and contains the tree $\tau^\star$.
- Moreover, there is a change of regime in the gain of likelihood at $\tau^\star$.
- In the case the tree $\tau^\star$ is bounded this is a rigorous result.
- A similar result for a different class of models was recently pointed out by Massart and co-authors.

The smallest maximizer criterion

- Assume that the sample was really produced by a probabilistic context tree $(\tau^\star, p^\star)$.
- Consider now the set of candidate context trees maximizing the probability of the sample for each number of *degrees of freedom*.
- It turns out that this sample of champion trees is *totally ordered* and contains the tree $\tau^\star$.
- Moreover, there is a change of regime in the gain of likelihood at $\tau^\star$.
- In the case the tree $\tau^\star$ is bounded this is a rigorous result.
- A similar result for a different class of models was recently pointed out by Massart and co-authors.

A simulation study

We simulate a sequence $x_1, \ldots, x_n$ over the alphabet $A = \{0, 1, 2, 3, 4\}$ using the following context tree



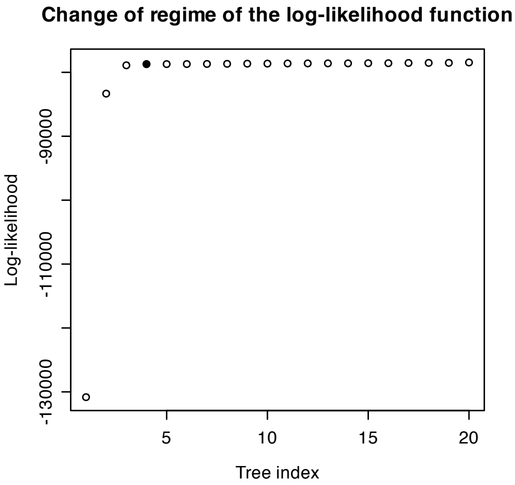
To perform the simulation we assign transition probabilities to each branch of the tree.

Using the tree and the transition probabilities we simulate 100,000 symbols.

A simulation study

- The candidates champion tress have successively $1, 8, 11, 13, 16, 17, \cdots$ leaves. The tree with 13 leaves corresponds to the correct tree (the tree we use to simulate the data).
- When we plot the log-likelihood of the sample as a function of the number of leaves we see a change of regime, as stated by our Theorem.

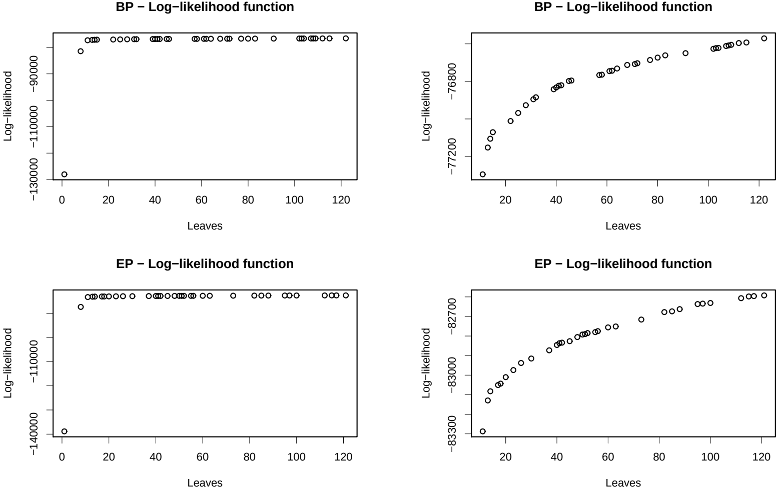
A simulation study



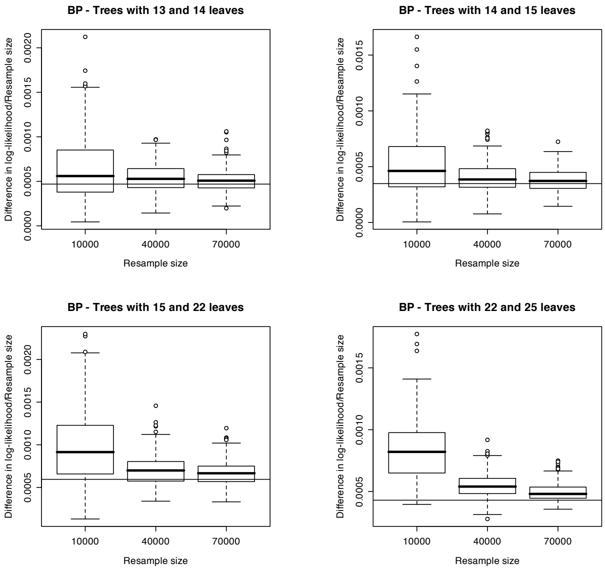
Application to the linguistic data set

- The sample consists of 80 articles randomly selected from the 1994 and 1995 editions.
- We chose 20 articles from each year for each newspaper.
- We ended up with a sample of 97,750 symbols for BP and 105,326 symbols for EP.

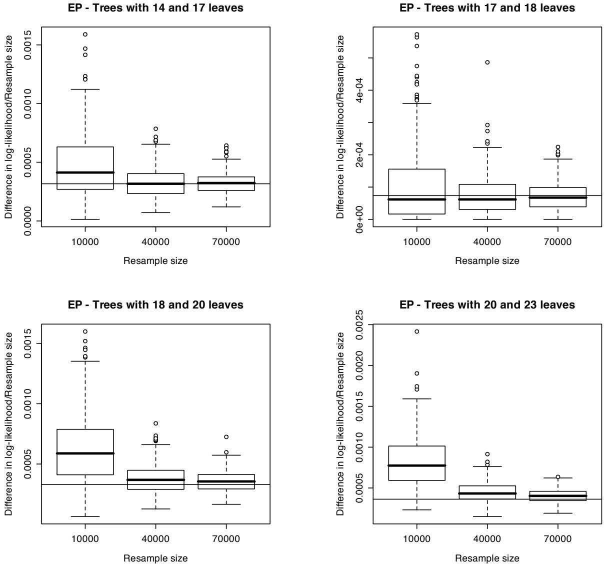
Application to the linguistic data set



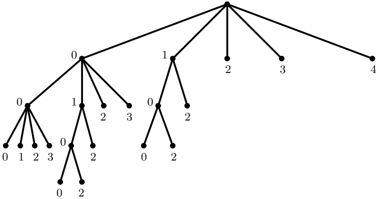
Application to the linguistic data set



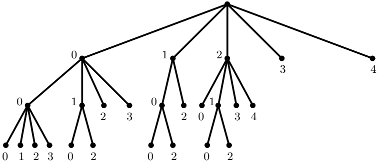
Application to the linguistic data set



Application to the linguistic data set



BP tree



EP tree