How to deal with Missing Data in Time Series and the imputeTS package

useR! 2017, Brussels

Steffen Moritz, TH Köln

steffen.moritz10@gmail.com

07.07.17

Talk Overview

- 1. Introduction
- 2. Imputation landscape on CRAN
- 3. Time Series Imputation specifics
- 4. imputeTS Introduction

We are often facing missing data

Examples from our projects:

Water quality measuring station: sensor problems

Water reservoir: cell reception problems

- Especially sensor measurements are prone to missing data
- Avoiding missing data should be the prioritized over filling NAs

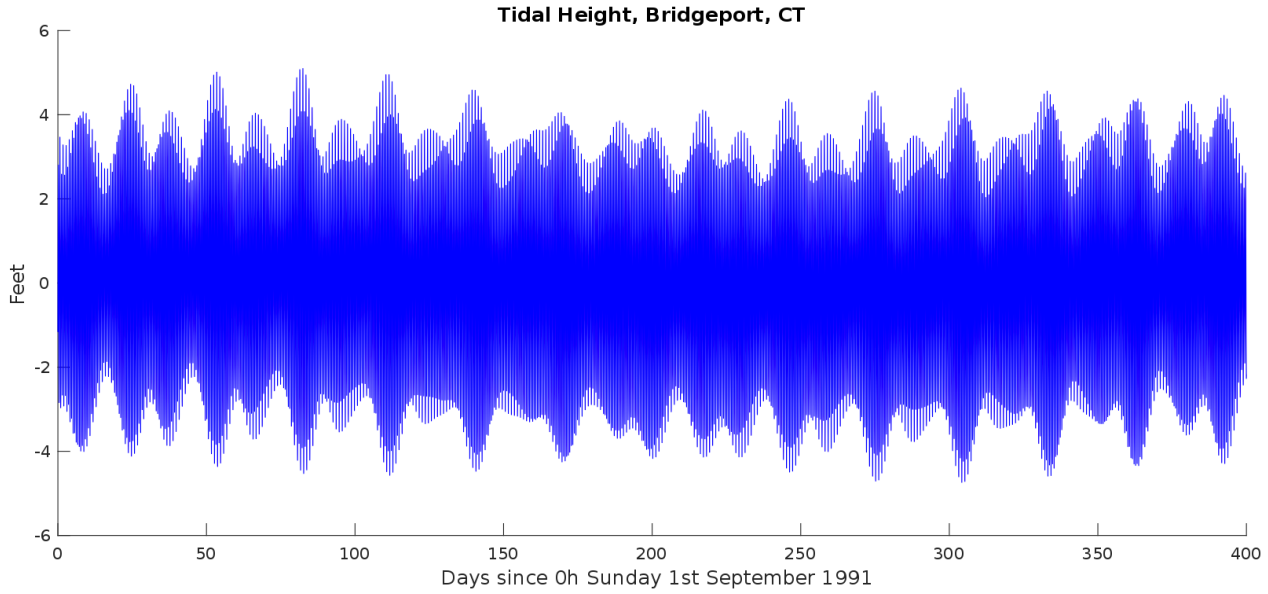
There are other people with the same problems…

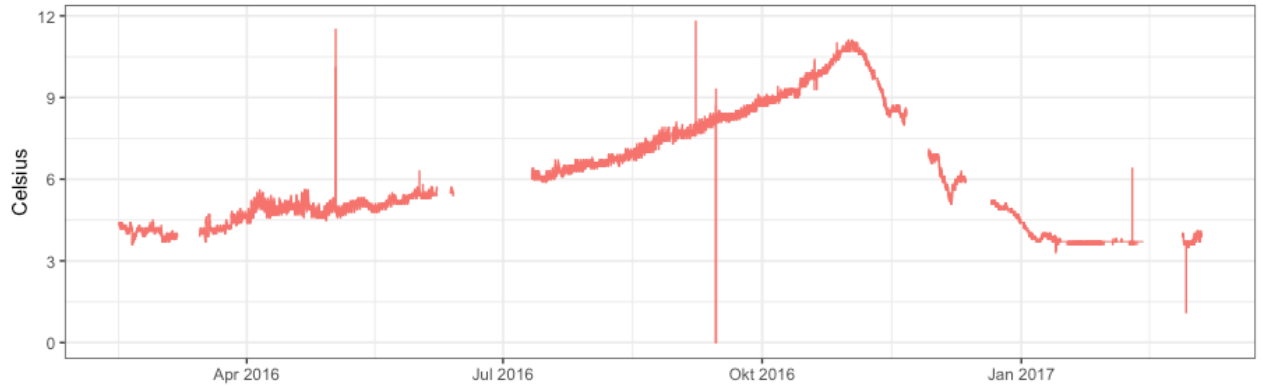
Field of expertise of people asking about imputeTS package:

- Hydrology
- Oceanography
- Quantitive Finance
- Meteorology

This included:

- gauge tide data
- sea-surface temperatures
- rainfall data





How to deal with Missing Data in Time Series

- 1. Visualization and statistics of missing data
- 2. Select Approach

Delete missing data

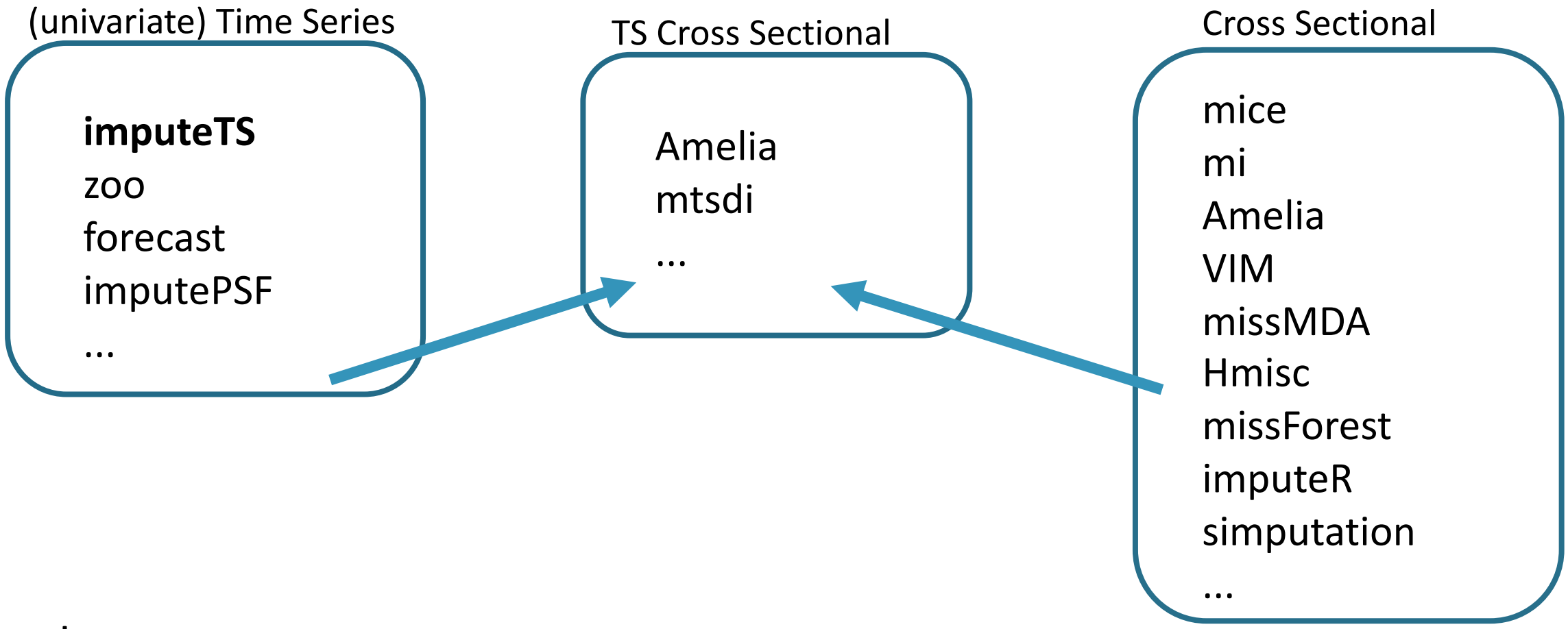
Keep missing data

Replace missing data

Imputation / gap filling

- 3. Select Algorithm

Simple Map of CRAN imputation packages



Input:

ts, vector, zoo, xts, timeSeries

data.frame, mts, zoo, xts, matrix, timeSeries

data.frame, matrix

Employing Correlations

V1	V2	V3	V4
91	91	91	91
NA	13	13	13
14	14	14	14
55	55	55	55
19	19	19	19
32	32	32	32
23	23	23	23
27	27	27	27
67	67	67	67

Cross Sectional

inter-variable

Time	V1	V2	V3
t1	13	33	15
t2	13	34	NA
t3	13	35	15
t4	13	36	16
t5	13	37	16
t6	14	38	16
t7	14	39	16
t8	14	40	17
t9	14	41	17

TS Cross Sectional

inter-variable + inter-time

Time	V1
t1	12
t2	12
t3	NA
t4	13
t5	13
t6	13
t7	14
t8	14
t9	14

Time Series

inter-time

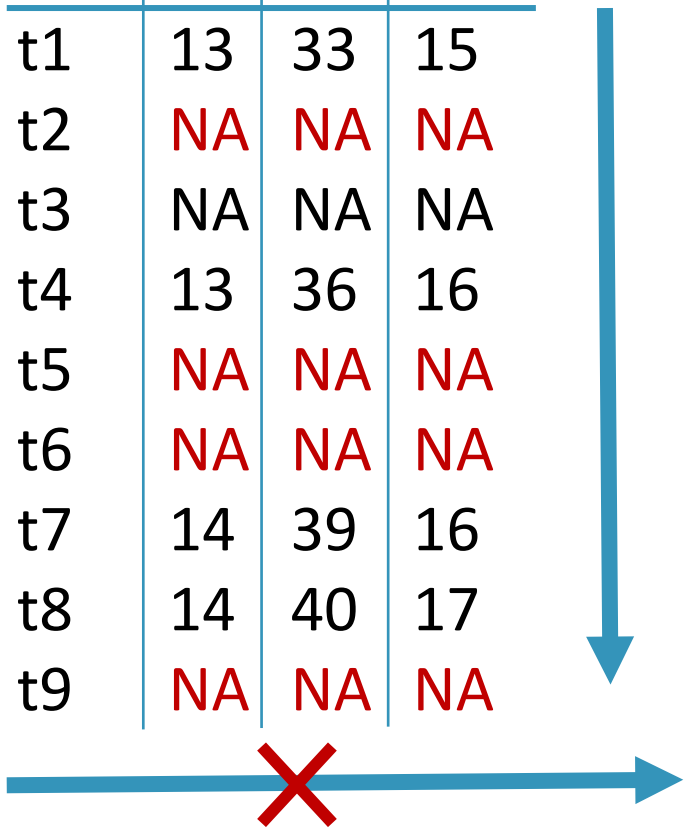
Time Series Imputation Specifics

- Considering time series characteristics like trend and seasonality is essential
- Although called univariate, time is an additional variable, which is implicitly given
- For MCAR / MAR / MNAR determination time has to be considered as a variable

Australian monthly gas production from forecast pkg

Also TSCS data needs univariate imputation sometimes

Time	V1	V2	V3
t1	13	33	15
t2	NA	NA	NA
t3	NA	NA	NA
t4	13	36	16
t5	NA	NA	NA
t6	NA	NA	NA
t7	14	39	16
t8	14	40	17
t9	NA	NA	NA



TS Cross Sectional

Problem:
Only whole observations are missing (V1,V2,V3 at one point in time)

This is often common for transmission problems

Thus inter-variable correlation can not be sufficiently employed

--> Pure time series imputation needed

imputeTS CRAN package

README.md

repo status Active | build passing | build passing | codecov 90% | CRAN 2.5 | CRAN 2017-06-13 | downloads 5588/month

imputeTS: Time Series Missing Value Imputation

The imputeTS package specializes on (univariate) time series imputation. It offers several different imputation algorithm implementations. Beyond the imputation algorithms the package also provides plotting and printing functions of time series missing data statistics. Additionally three time series datasets for imputation experiments are included.

Installation

The imputeTS package can be found on CRAN. For installation execute in R:

```
install.packages("imputeTS")
```

The idea behind the package

- Inspired from own sensor data use cases

  Rather big time series. Leading to combination of fast and advanced algorithms.

- Domain experts as users

  Easy and quick access to advanced functions. No multiple imputation.

- Whole imputation process in one package

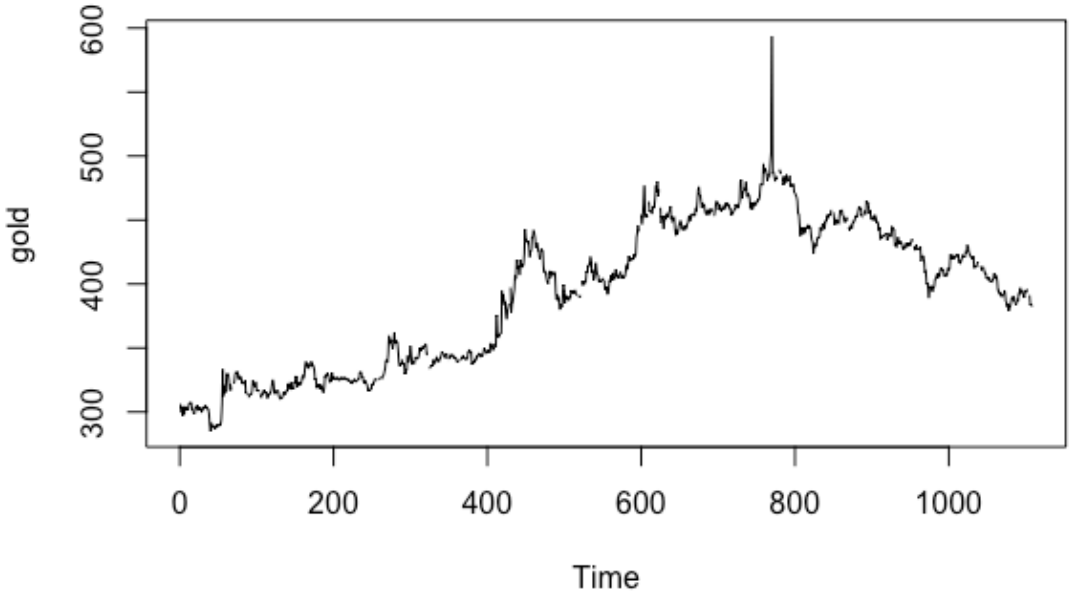
  Visualization + Imputation + Result Analysis

Package Scope

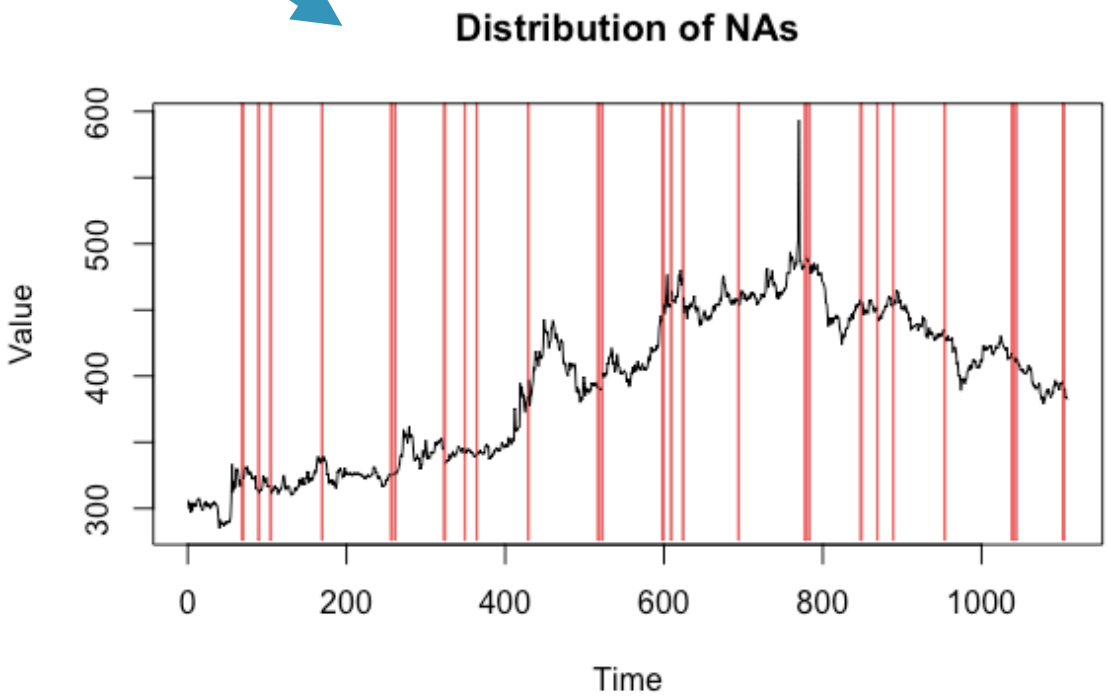
- Analysis before NA action
  - 3 Missing Data Plots
  - NA statistic text output
- Analysis after imputation
  - 1 Result Plot
- Imputation functions
  - 5 fast imputation functions
  - 4 more advanced functions
  - NA remove function
  - 3 Datasets for testing

Visualization of NA distribution

plotNA.distribution(yourInput)



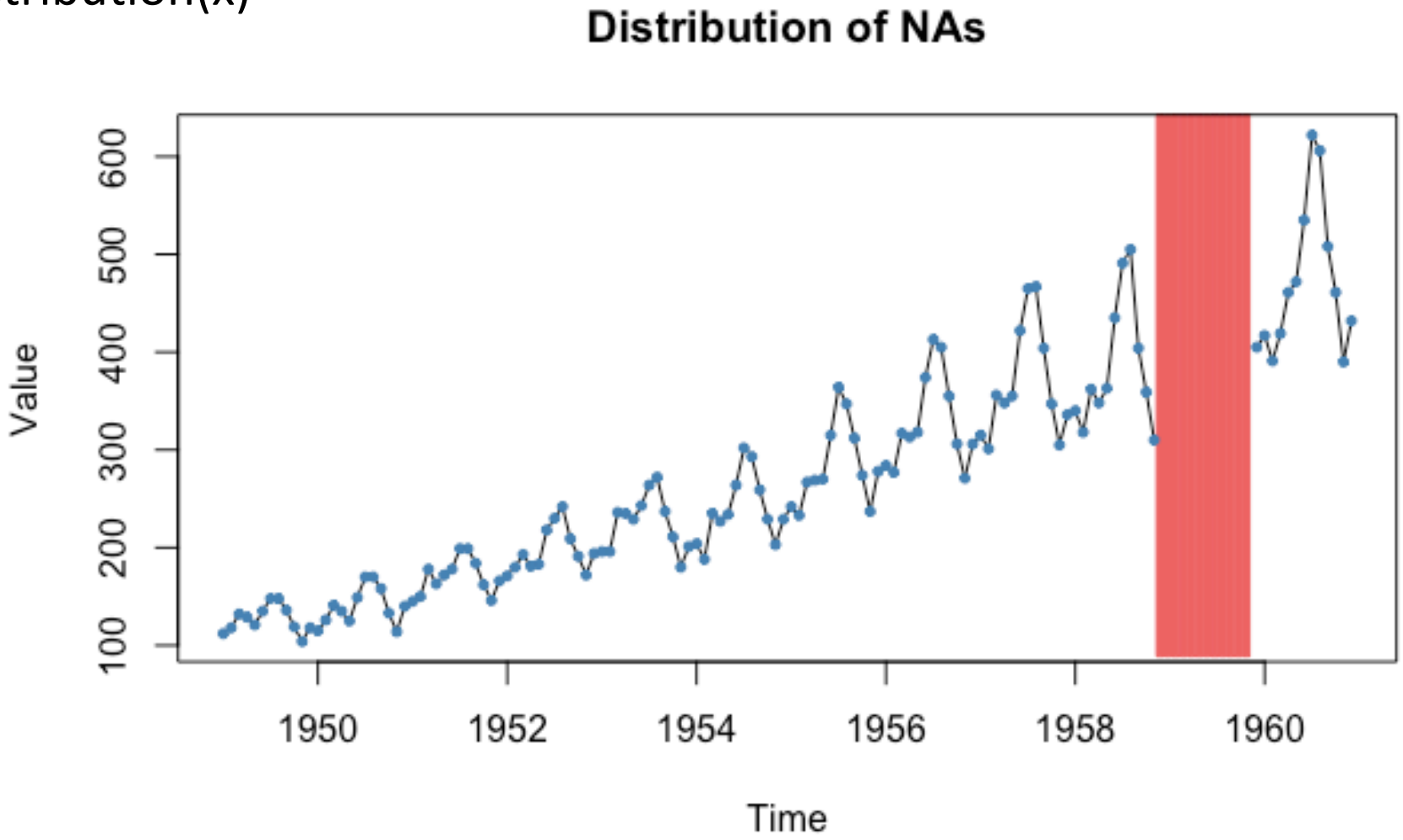
Daily morning gold prices from forecast package



Visualization of how the NAs are distributed in the series

Visualization of NA distribution

plotNA.distribution(x)

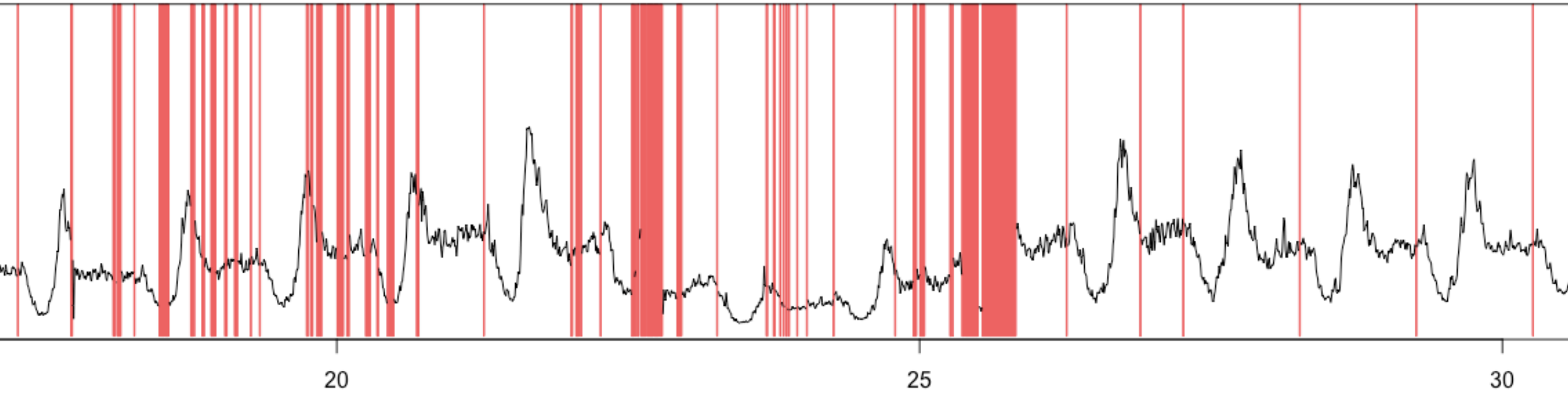


AirPassengers from datasets package with manually introduced NAs

Sometimes time series are just to long

plotNA.distribution(tsNH4)

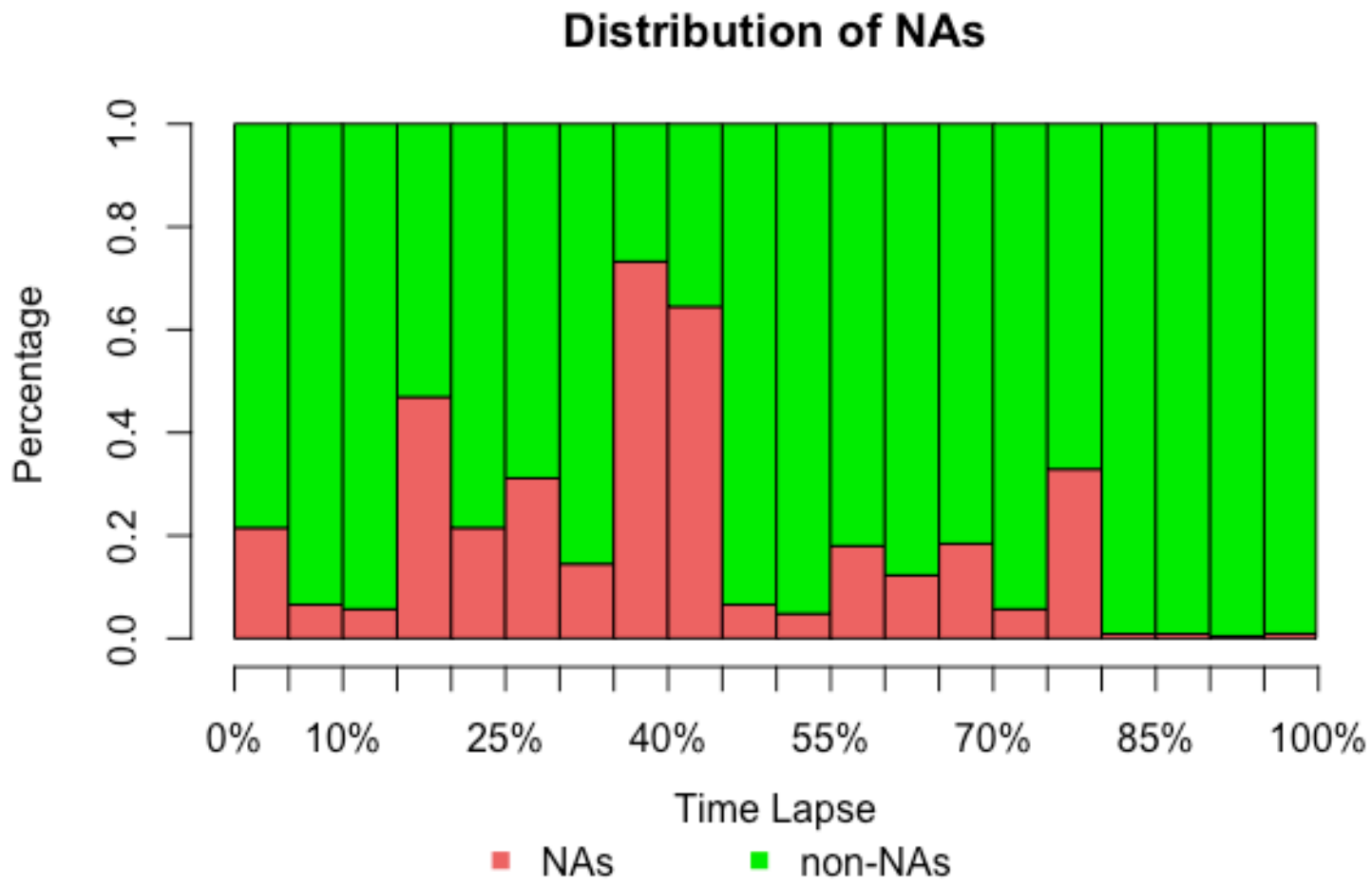
Just too long



tsNH4 from imputeTS package

Visualization of long time series

plotNA.distributionBar(tsNH4, breaks=20)



Additional Stats

statsNA(tsHeating)

```
"Length of time series:"
606837
"-------------------------"
"Number of Missing Values:"
57391
"-------------------------"
"Percentage of Missing Values:"
"9.46%"
"-------------------------"
"Stats for Bins"
"  Bin 1 (151710 values from 1 to 151710) :      0 NAs (0%)"
"  Bin 2 (151710 values from 151711 to 303420) :      29755 NAs (19.6%)"
"  Bin 3 (151710 values from 303421 to 455130) :      6153 NAs (4.06%)"
"  Bin 4 (151707 values from 455131 to 606837) :      21483 NAs (14.2%)"
"-------------------------"
"Longest NA gap (series of consecutive NAs)"
"258 in a row"
"-------------------------"
"Most frequent gap size (series of consecutive NA series)"
"2 NA in a row (occuring 104 times)"
"-------------------------"
"Gap size accounting for most NAs"
```

Imputation Options

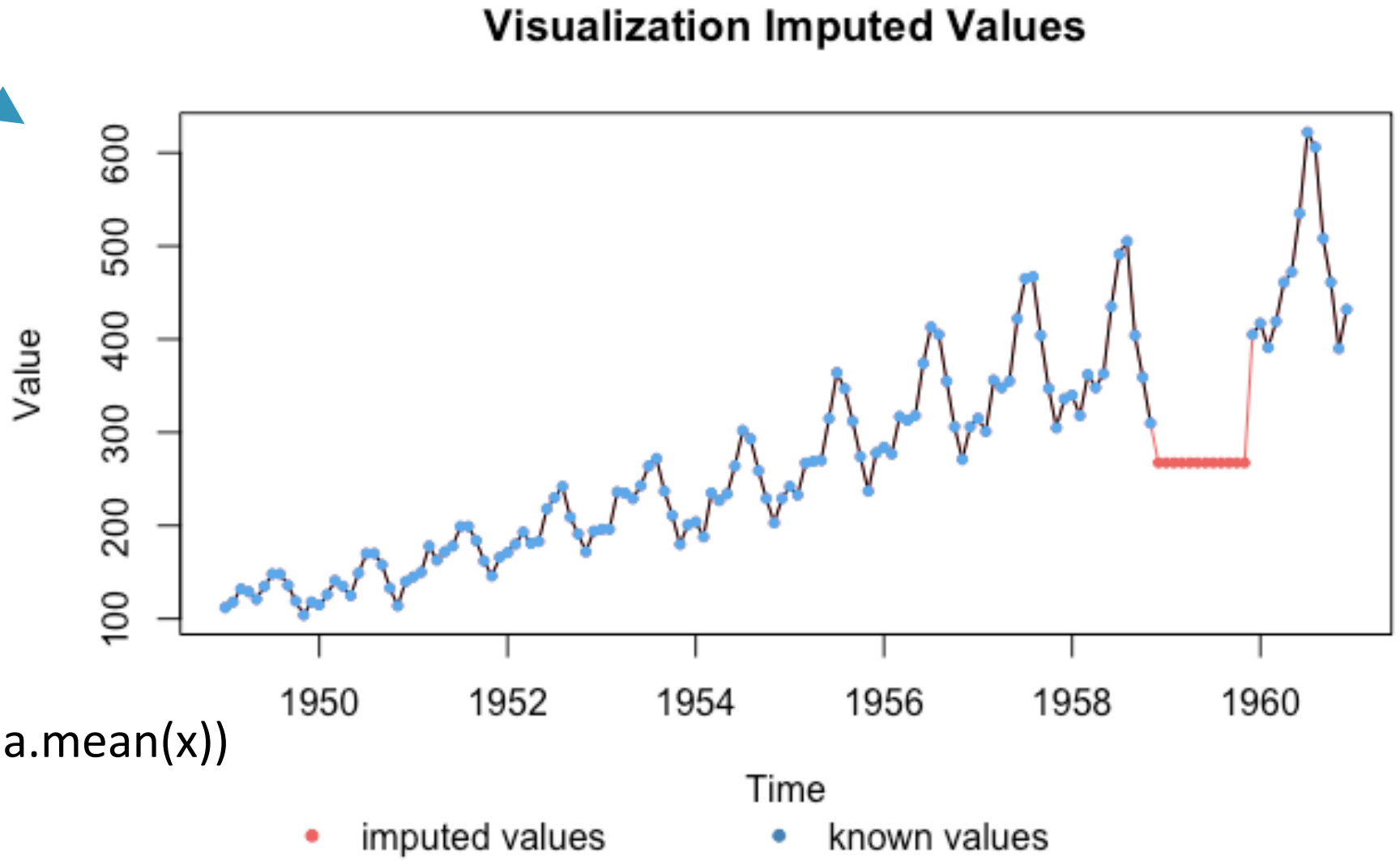
Function	Description
na.interpolation	Missing Value Imputation by Interpolation
na.kalman	Missing Value Imputation by Kalman Smoothing
na.locf	Missing Value Imputation by Last Observation Carried Forward
na.ma	Missing Value Imputation by Weighted Moving Average
na.mean	Missing Value Imputation by Mean Value
na.random	Missing Value Imputation by Random Sample
na.remove	Remove Missing Values
na.replace	Replace Missing Values by a Defined Value
na.seadec	Seasonally Decomposed Missing Value Imputation
na.seasplit	Seasonally Splitted Missing Value Imputation

Easy to use!

- na.‘algorithmname‘(yourInput, add. param)
  - Same syntax also used by other packages like zoo, forecast
- Imputation functions take all kinds of inputs:
  - ts, mts, data.frame, matrix, zoo, xts, vector

Imputation with na.mean

na.mean(x)

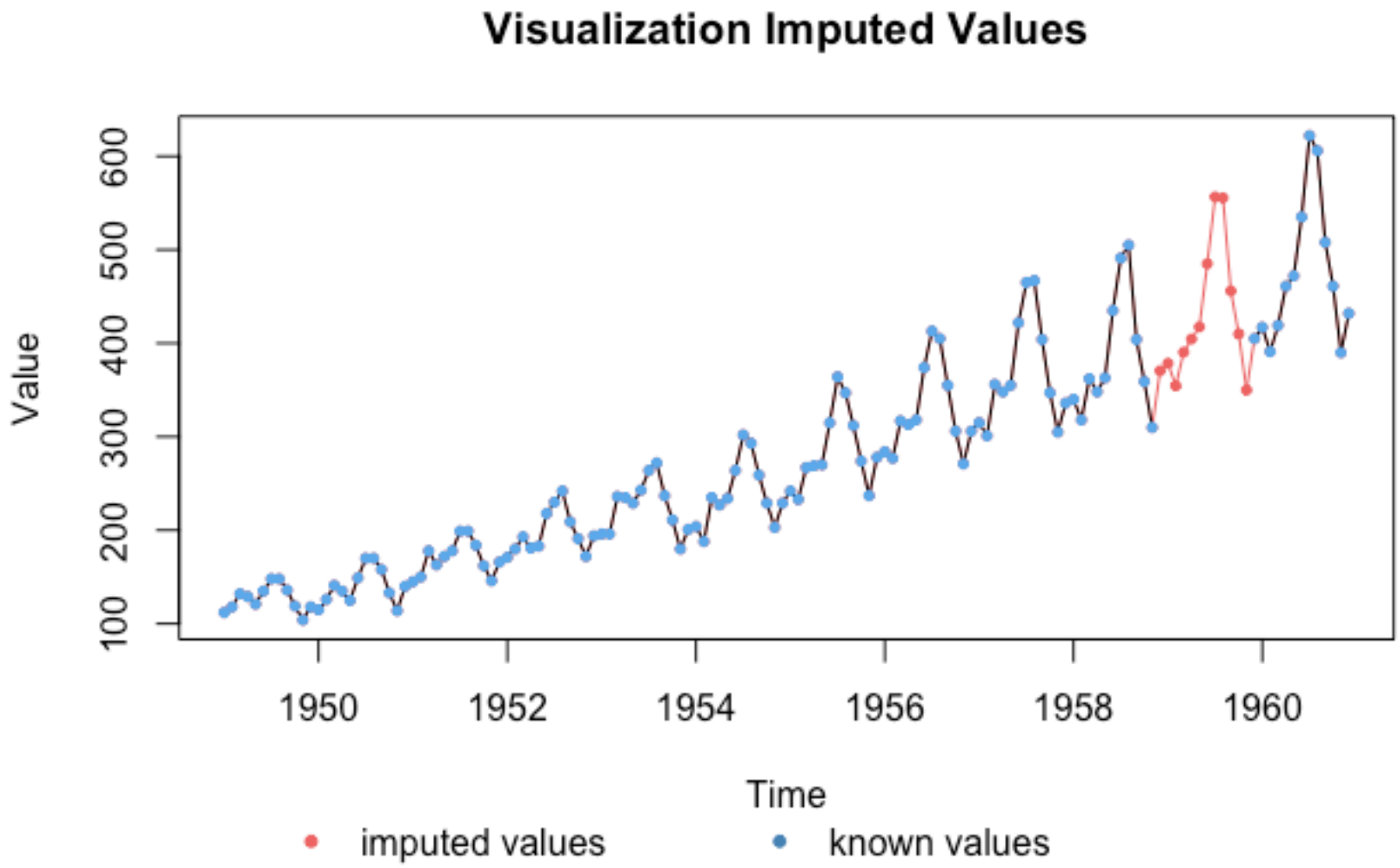


plotNA.imputations(x, na.mean(x))

AirPassengers from datasets package with manually introduced NAs

Imputation with na.seasplit

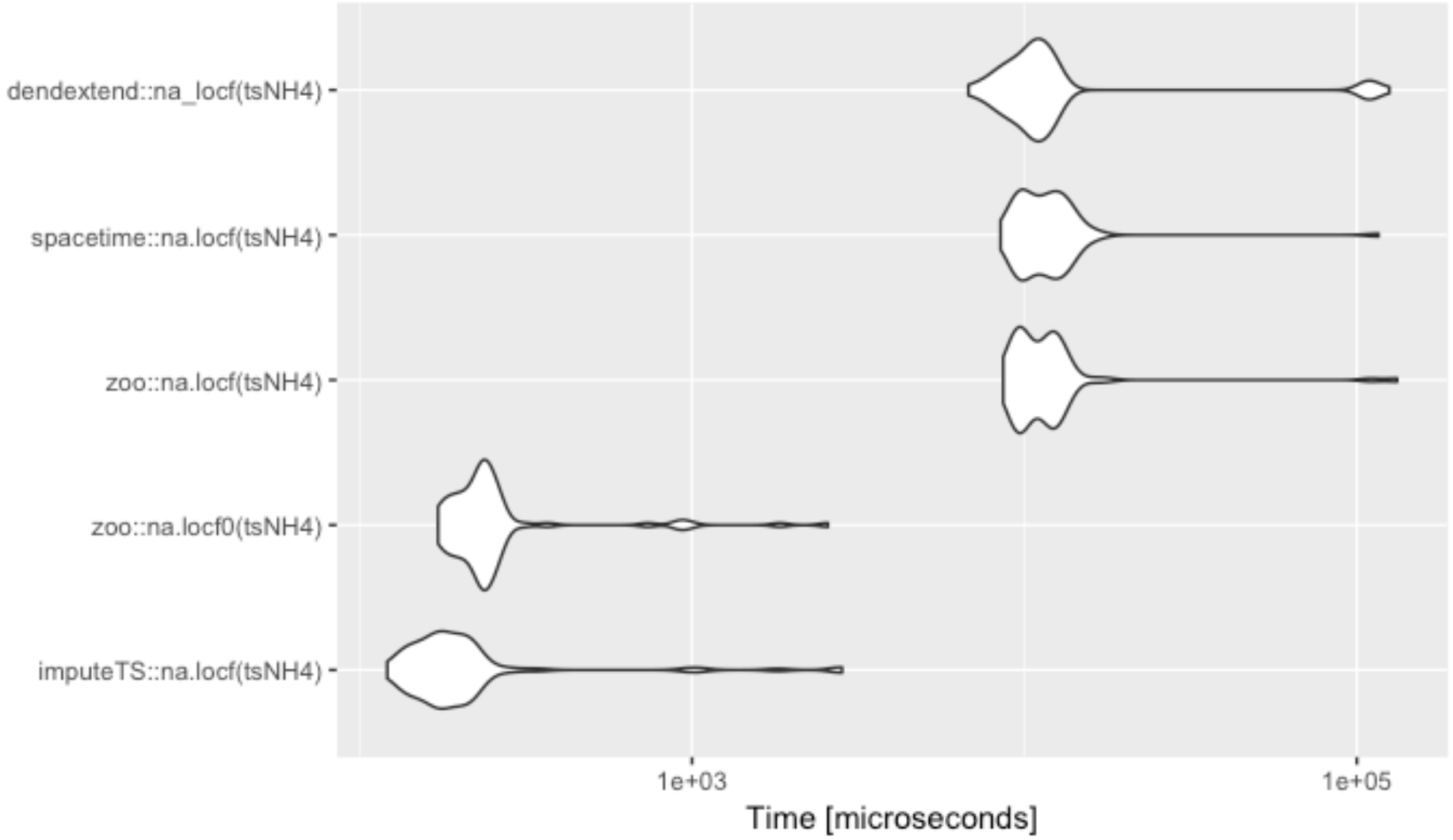
na.seasplit(x)



Fast: Last observation carried forward

LOCF
dendextend
spacetime
zoo
imputeTS

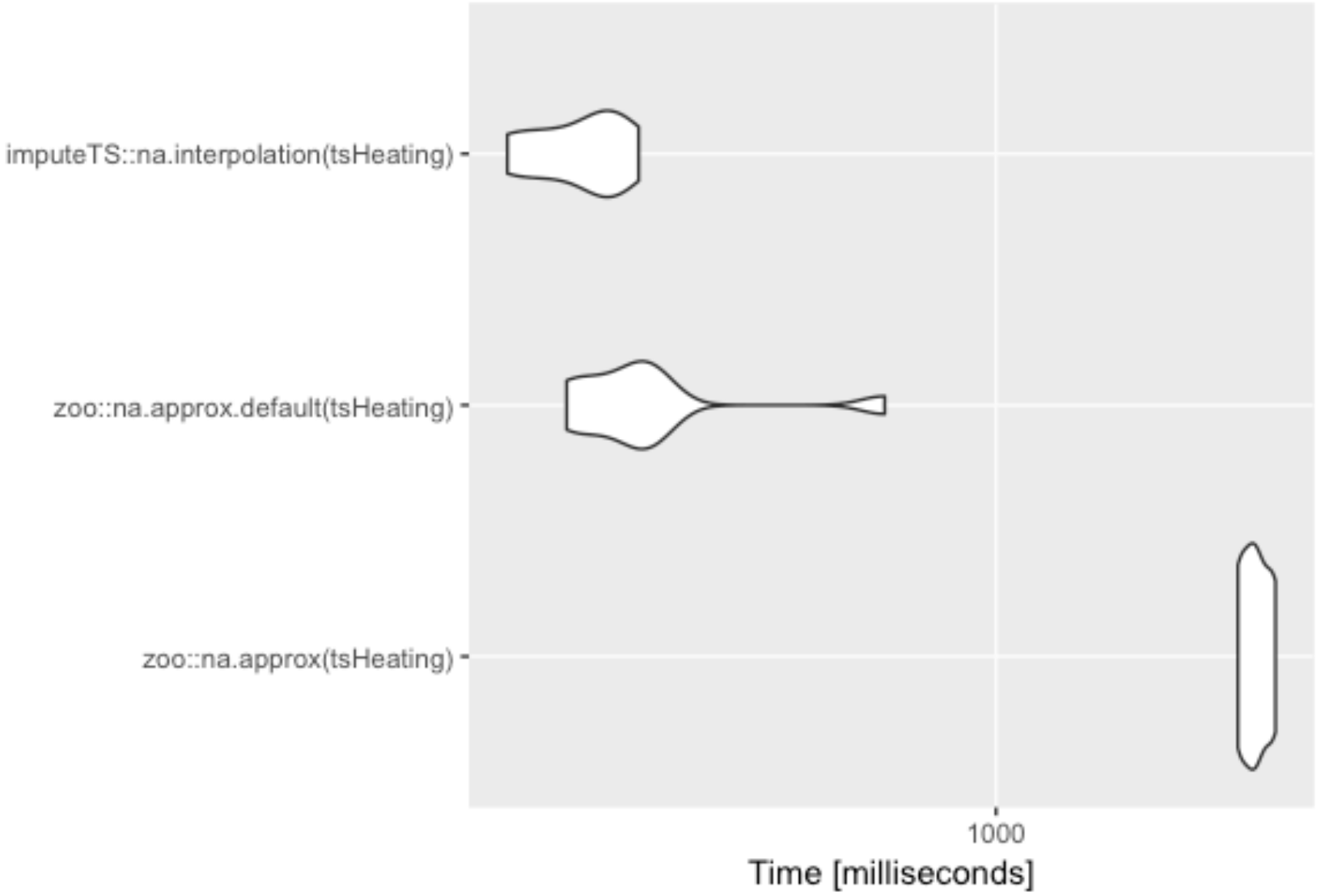
tsNH4
Length: 4552
NAs: 883



Fast: Linear Interpolation

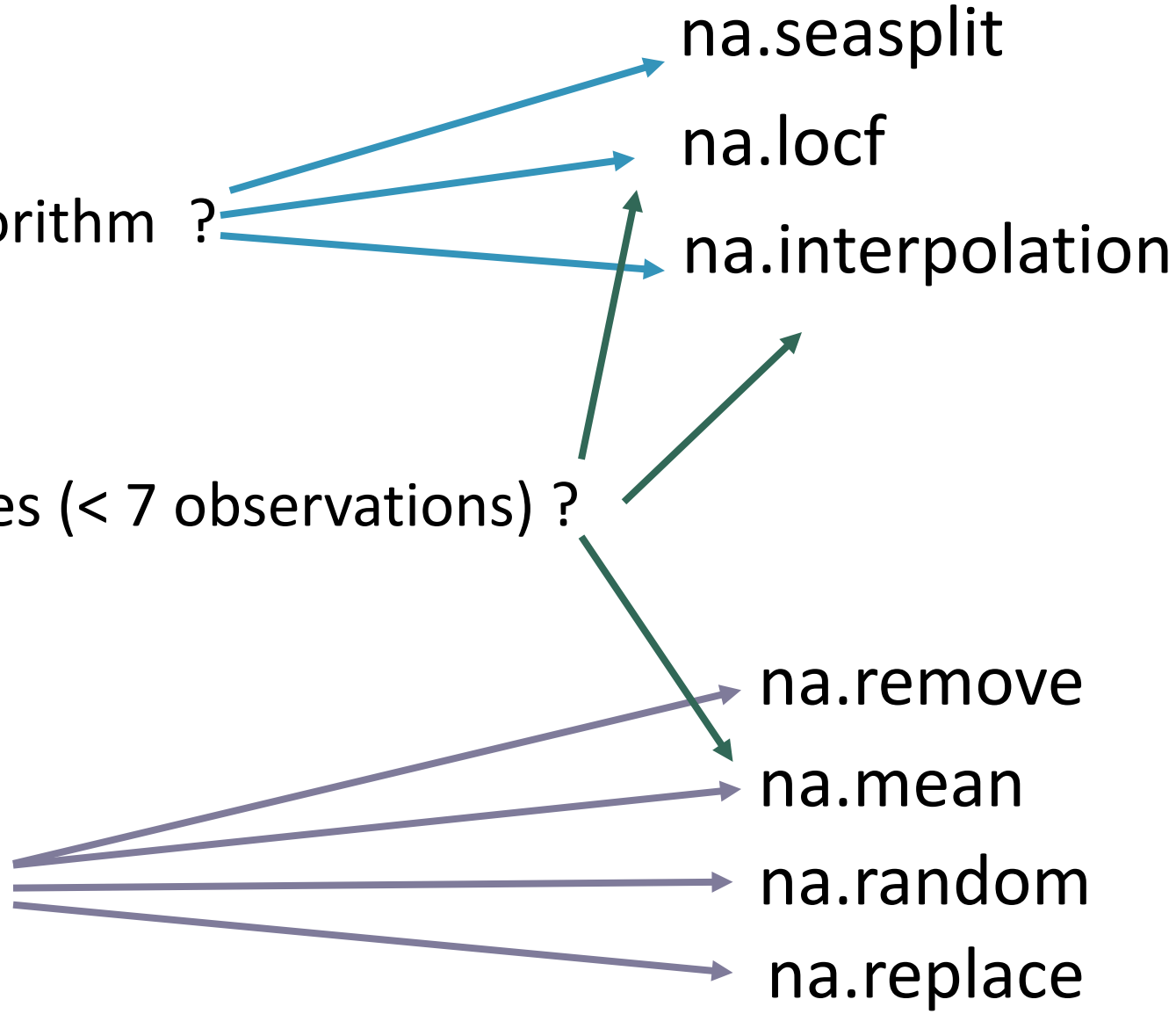
Linear interp.
imputeTS
zoo

tsHeating
Length: 606837
NAs: 57391



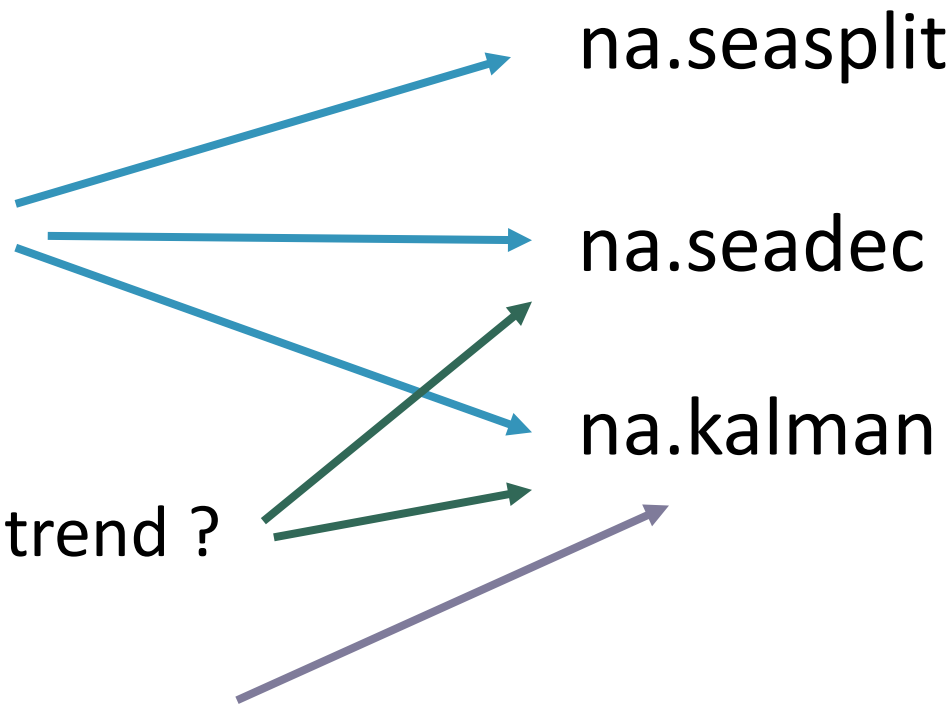
Choosing the right algorithm

- Do I need a very fast algorithm ?
- Do I have very short series (< 7 observations) ?
- Usually not a good idea!



Choosing the right algorithm

- Does my data have seasonality ?
- Does my data have seasonality + trend ?
- Most of the times a good but slow choice



Trying and assessing different algorithms is always a good idea.

Get in contact & download the package

steffen.moritz10@gmail.com