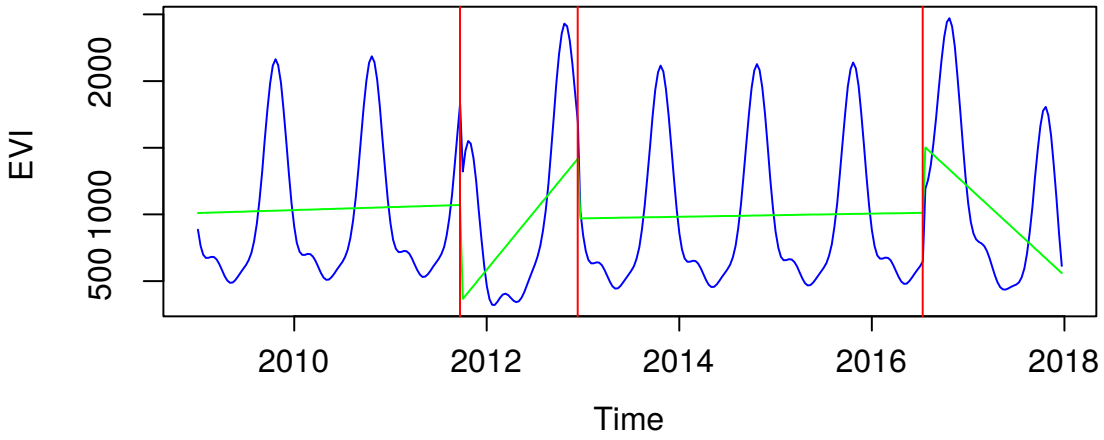
Automated global land cover disturbance monitoring using BFAST Lite

Dainius Masiliūnas, *Jan Verbesselt*

WAGENINGEN
UNIVERSITY & RESEARCH

About me

- PhD candidate and lecturer in the Laboratory of Geo-information Science and Remote Sensing, Wageningen University
- Thesis about global land cover mapping and updating using time series analysis
- Copernicus Global Land Services Land Cover project



WAGENINGEN
UNIVERSITY & RESEARCH

Other things I work on

- Lecturing (Master of Geo-information Science)
    - Geoscripting course: https://geoscripting-wur.github.io
- SENSECO project and sun-induced fluorescence
    - Time series analysis
    - Point-based hyperspectral measurements from drones for photosynthesis efficiency and plant stress
- OpenEO project, collaboration with VITO and Terrascope
- **BFAST** package maintenance

Land cover change detection for updating

- Reusing the same land cover classification model for the next year leads to too many spurious changes
- Expert rules: which transitions are possible/likely
- Use time series break detection to constrain changed pixels
- Many options for time series break detection algorithms!

Unlikely land cover change: from urban to water

Breaks For Additive Seasonal Trend (BFAST)

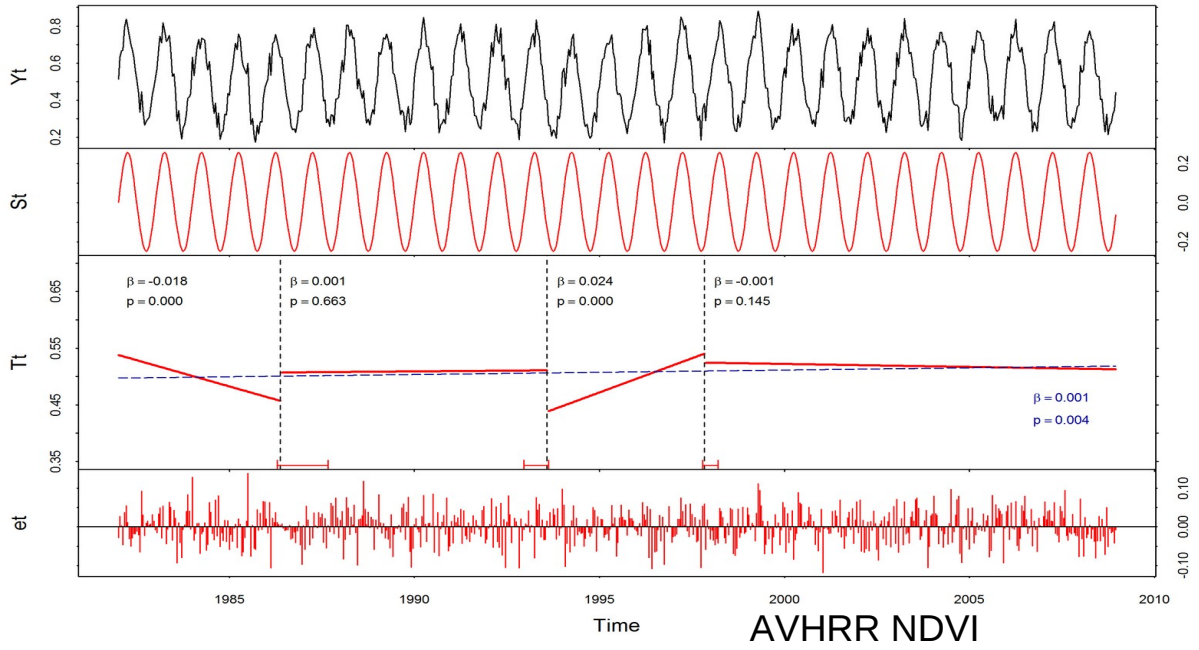
- The components of a time series (of a **vegetation index**):

1)Seasonality

2)Trend

3)Noise

$$Y_t = T_t + S_t + e_t$$

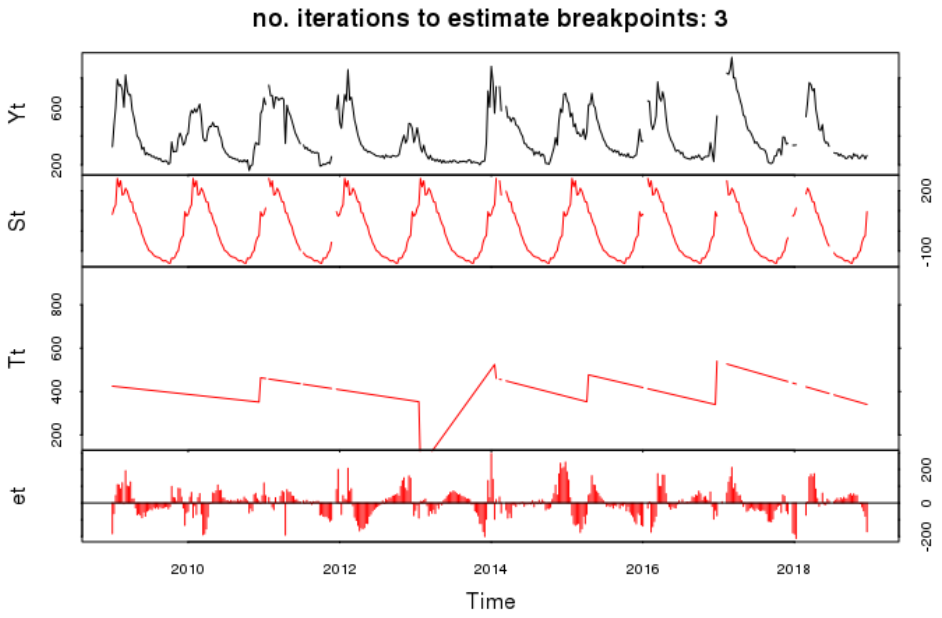


J. Verbesselt, R. Hyndman, G. Newnham, and D. Culvenor, Detecting trend and seasonal changes in satellite image time series, Remote Sensing of Environment, vol. 114, no. 1, pp. 106-115. (2010).

WAGENINGEN
UNIVERSITY & RESEARCH

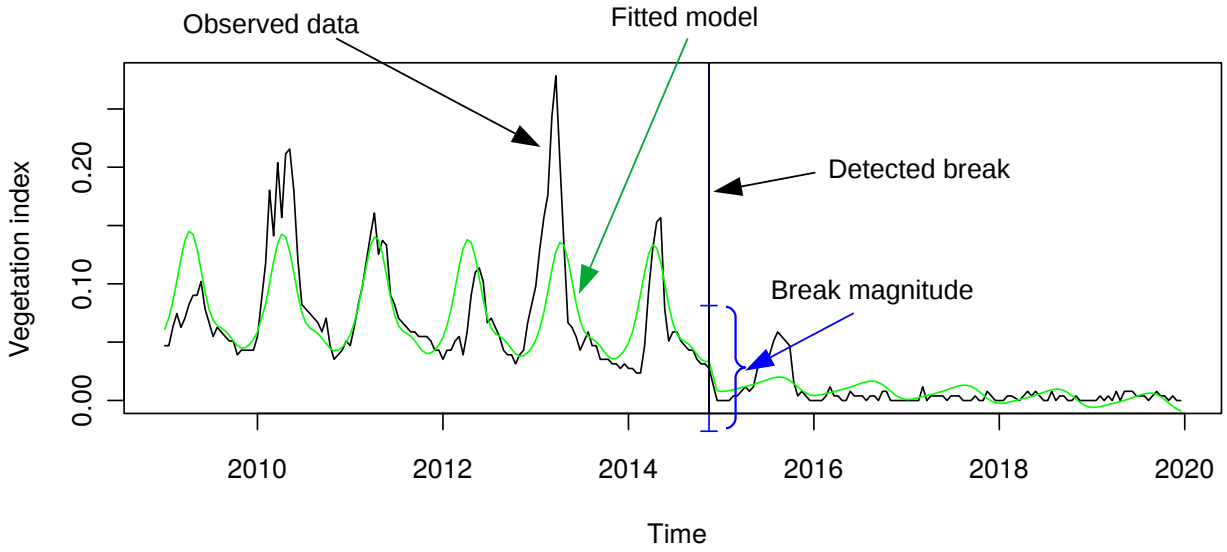
BFAST

- BFAST: Breaks For Additive Season and Trend
- Decomposition of time series into seasonal, trend and remainder components using stl()
- Components subdivided into stable segments, segment divisions are breaks
- Iterative (stl() on stable segments)
- Detects all breaks in the time series and specifies whether it's seasonality or trend break



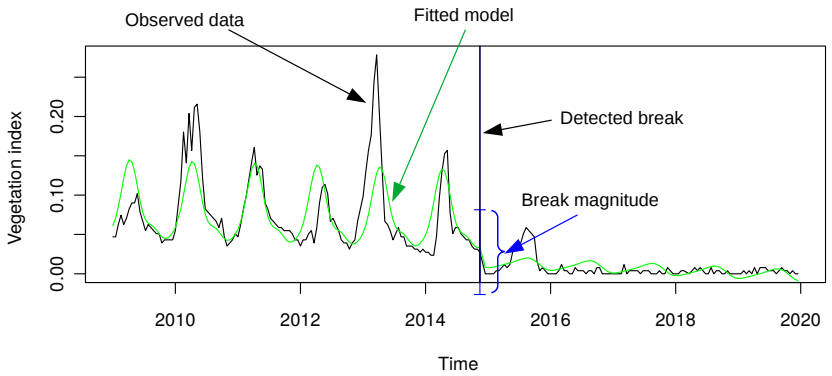
BFAST Lite

- Detecting breaks in all components at once in a single pass
- Can handle missing values
- More tunable parameters: can use harmonics (sin/cos) or seasonal dummies (multiplier per season) or external regressors to fit the data
- Is an order of magnitude faster than BFAST (in addition to speed improvements by Marius Appel)

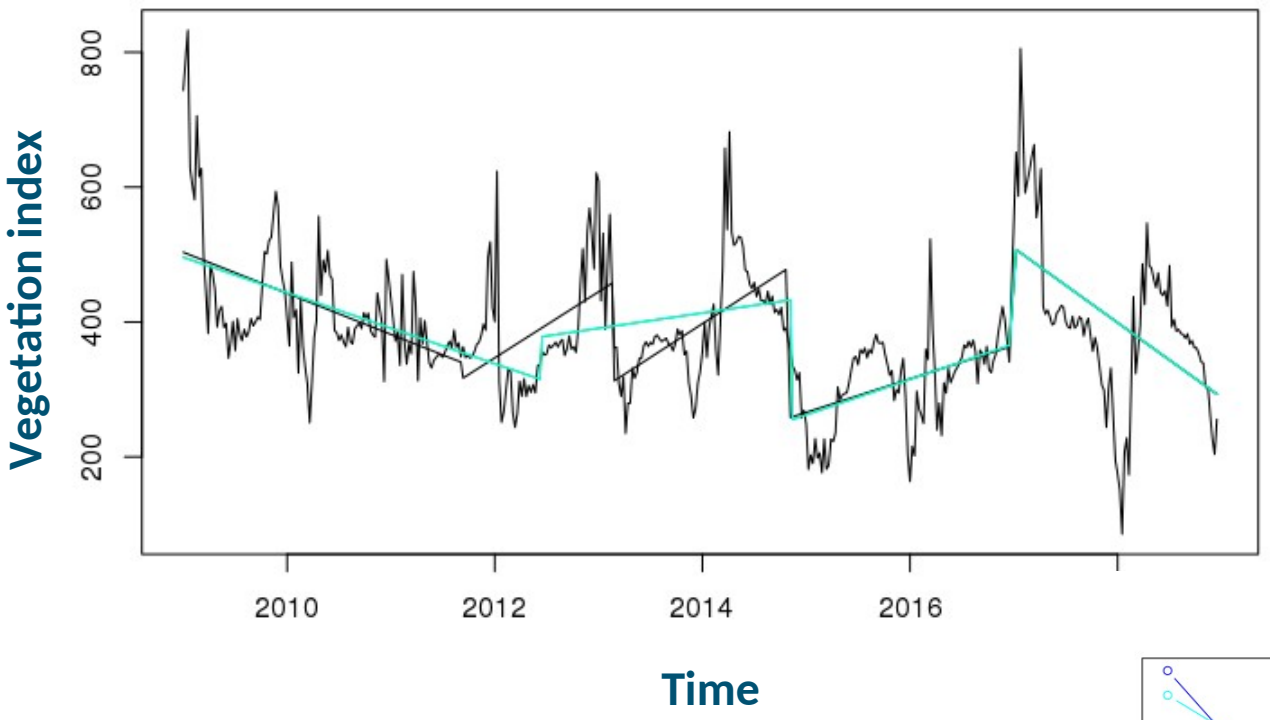


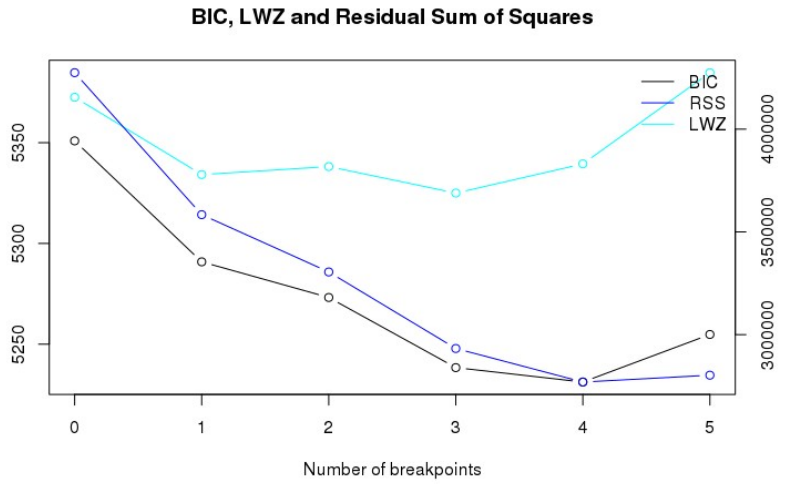
Principle of breakpoints()

- Piece-wise linear regression:
  - Given that we want one break, what's the optimal location to put it so that the RSS of two segments is minimised?
  - What if we want two breaks?
  - Etc. etc. to get a triangular matrix of possible breaks and model RSS
- But how many breaks does the time series have?
  - An Information Criterion: if we increase degrees of freedom by adding breaks, data will fit better, so penalise for each degree of freedom added
  - AIC (k=2) is too weak, BIC (k=log(n)) is also often too weak
  - LWZ ($k=0.299 \times \log(n)^{2.1}$) seems to do better



Breakpoints using LWZ vs BIC





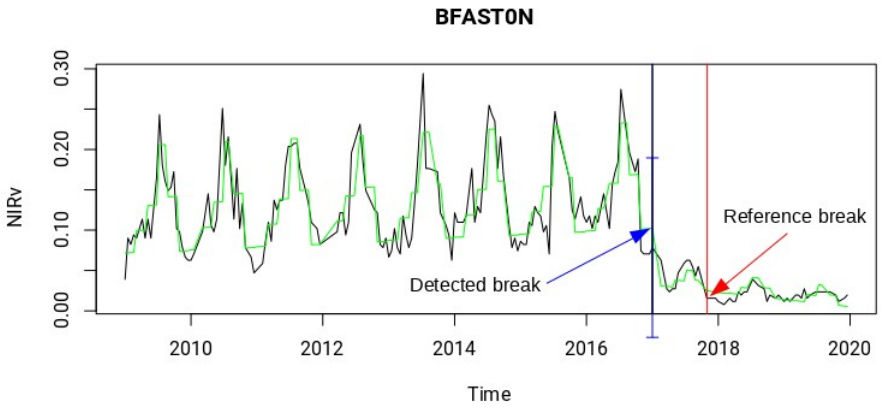
WAGENINGEN
UNIVERSITY & RESEARCH

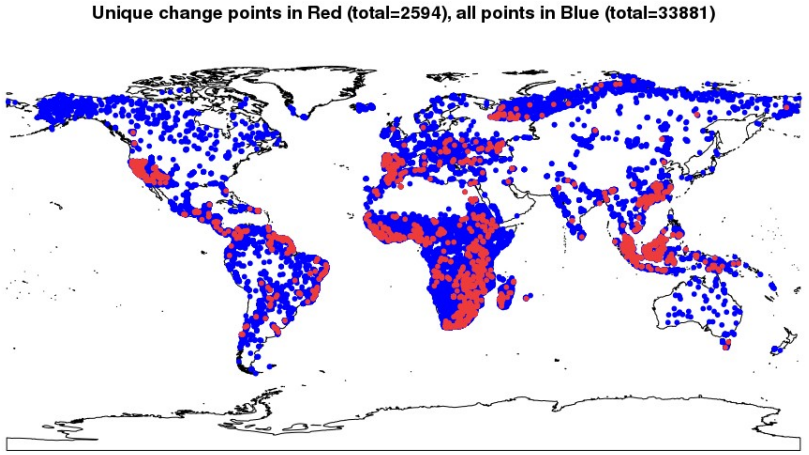
New in bfast 1.6

- Ability to use LWZ for selecting breaks
- Extra information when printing the results:
    - LWZ statistics
    - R^2
    - Break magnitude, using difference between segment models and the difference in last/first predicted value
- Parameter for customisable seasonal dummy number

BFAST Lite model parameter optimisation

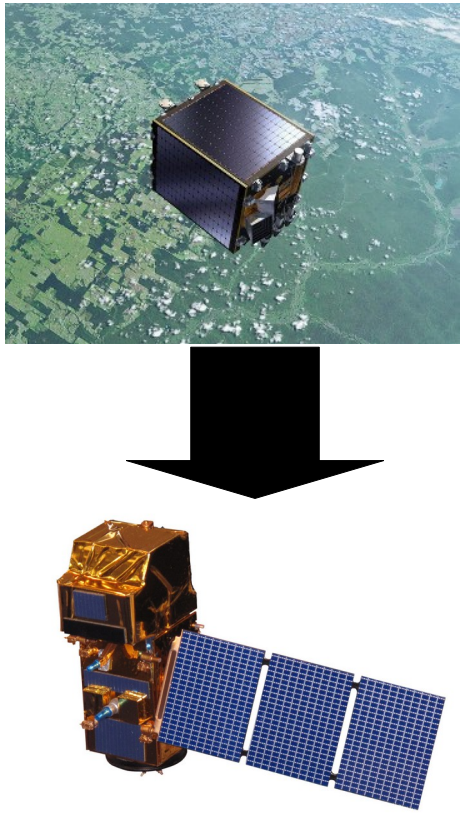
- We can detect changes, but how good can we do that?
- Change reference data by IIASA and WUR
- Optimised the parameters of BFAST Lite and BFAST Monitor using global data
- Generally overestimates change (BFAST Monitor more so)
- Pairing with classifier output and expert rules needed to further reduce spurious change





Future outlook: big data and deep learning

- Python and GEE versions
- Scaling down to 20 m
  - Land cover mapping: Sentinel-2 (20 m) instead of Proba-V (100 m), 25x
  - Change detection: Landsat (30 m) instead of MODIS (250 m), 70x
  - Add Sentinel-1 20 m data for gap filling
- Creating a new supervised variant
  - Based on RNNs + Python version
  - Uses all bands and automatically optimises parameters



Thank you for your attention!

Copernicus Europe's eyes on Earth

Land Monitoring Service

WAGENINGEN UNIVERSITY & RESEARCH

Workshop

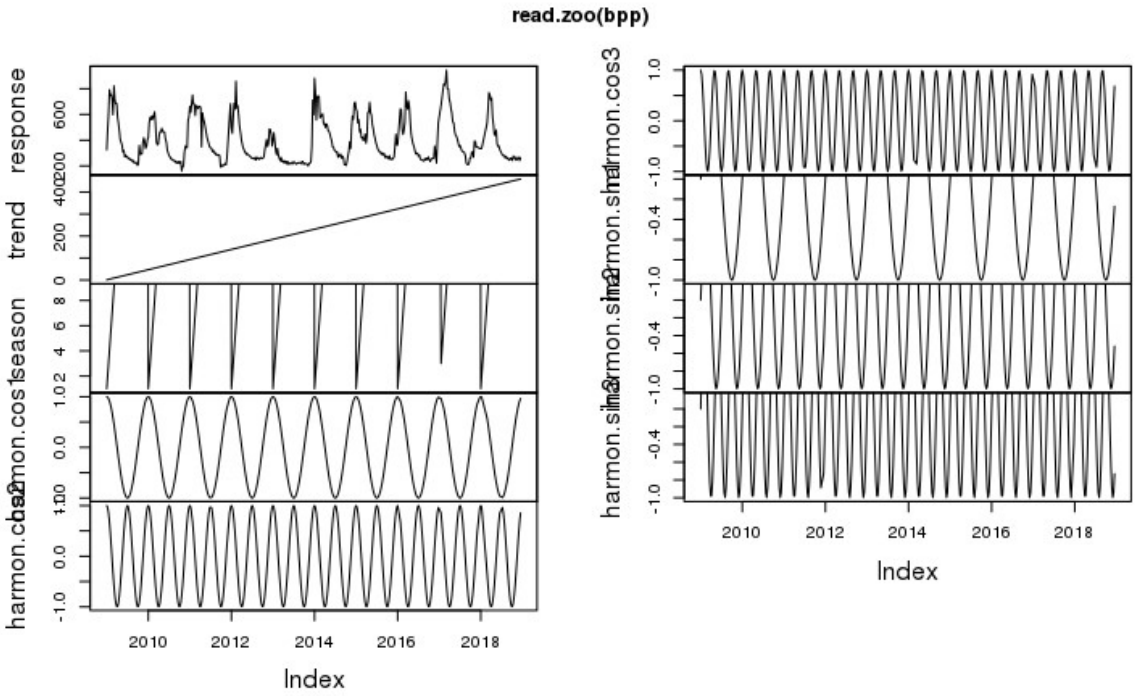
- Detect breaks in time series:

  https://verbe039.github.io/BFASTforAEO/

- If you'd like to get the original Rmarkdown file: https://github.com/verbe039/BFASTforAEO/blob/master/index.Rmd

bfastpp()

- How to get data with response ~ trend + harmon?
- bfastpp(ts, order): preprocessing of time series
    - ts must be a `ts` with frequency > 1
    - order is the harmonic order
- Output is a data.frame with:



bfastlite()

- In the bfast pacakge: install.pacakges("bfast")
- bfast::bfastlite(data, formula, h, ...)
    - bfast::bfastpp() + strucchangeRcpp::breakpoints()
    - data: a `ts` object (see bfastts() if you don't have one)
    - formula: e.g. response ~ trend + harmon
    - h: minimum segment size, either fraction of the time series length or integer defining the number of samples
- Output: a list containing a `breakpoints` object that indicates breakpoint timing and confidence interval, in sample numbers (mapping to `data`); you can use plot() and summary() for more info