1 Final project - Task 5

Deadline: 04/05, 21:00.

Research question

In this task, we return to the GISTEMP dataset. Your objective, as in the previous assignment, is to uncover the underlying temperature trend over time. This time using state-space methods.

Implementation

To begin, prepare the dataset by removing the seasonal component to isolate the long-term behavior. For the Seasonal DLM, you will use the full dataset (with seasonality).

Dynamic Linear Models (DLMs) You will implement and compare the following models:

- **Random Walk plus Noise**: This model assumes the observed temperature series Y_t follows a latent process μ_t, which evolves slowly over time:

$$\begin{aligned} Y_t &= \mu_t + \epsilon_t, & \epsilon_t &\sim \mathcal{N}(0, \sigma^2) \\ \mu_t &= \mu_{t-1} + \eta_t, & \eta_t &\sim \mathcal{N}(0, \tau^2) \end{aligned}$$

- **Locally Linear Trend Model**: This extension allows both the level and slope to change gradually:

$$\begin{aligned} Y_t &= \mu_t + \epsilon_t, & \epsilon_t &\sim \mathcal{N}(0, \sigma^2) \\ \mu_t &= \mu_{t-1} + \beta_{t-1} + \eta_t, & \eta_t &\sim \mathcal{N}(0, \tau^2) \\ \beta_t &= \beta_{t-1} + \zeta_t, & \zeta_t &\sim \mathcal{N}(0, \gamma^2) \end{aligned}$$

Your tasks:

- Fit each model to the full time series.
- Extract the latent components (level, trend, seasonality) using smoothing techniques.
- Evaluate the models prediction based on their interpretability and quality.
- Address: Is this class of models more suitable than HMM for the task at hand?

Guidelines

Remember that the assignments (as the final project) are also a useful exercise of **presentation**. Below you find suggestions on how your analysis should be presented (they were posted, and are still available, on BBoard).

Submission

- Report in pdf (if from rmarkdown: Knit to PDF. do not export HTML and then print)
- Code in .R or .rmd
- Name of file = group name

Length The PDF file must be no longer than **3** pages.

First page includes

- Group name
- Names of group components

Format Remember: you are supposed to send your code so the report should not include any!

- NO R console output: use tables
- NO R messages
- NO R code anywhere ever
- NO code chunks
- NO mention of the functions you use, and no explanation of your code
- Can the report be read 100% the same way if the code was not written in R? If yes, then good; if not, then make it independent of the code. The analyses and your interpretations are important, not the specifics of your code. Good code will lead to more elegant analyses, plots and overall presentation
- NO screenshots

Contents

- All models are written in formulas
- Notation is consistent

- Estimates for all unknowns are reported in tables/plots or discussed in the text and interpreted
- Model comparisons are meaningful

Plots, figures and tables

- All plots have short description/title/caption and are numbered
- All figures numbered sequentially
- All figures are mentioned in text, in the order in which they appear
- All plots have meaningful axis titles (if not redundant e.g. in the title)
- All plots are well positioned in the page (centered)
- All text in the plots is readable without zooming in
- No text is too big in the plot
- Plots are not "warped"
- No plot is pixelated or blurry or with jpeg artifacts
- All plots are useful for the purpose of answering the research question
- All plots are explained and interpreted (not just described passively)
- Report quantities with names meaningful to the application and not with generic ones (e.g., State1, State2, ecc)

General

- Spacing is used efficiently: no excessive white spaces
- Borders are normal, line spacing is standard, no other weirdness to fit everything within the page limit
- English: spelling mistakes? Too verbose? Concise enough? We're not the British Council but you don't want to be sloppy.
- Report does not look hastily made or sloppy
- Report looks professional
- Text is concise and to the point

Code: we will randomly pick some groups for a code check. Or, we may check the code when figures or values look funny (as it happens).

- Submitted code can be compiled/run without error generating all figures and tables in the report, with the same numbers
- Code is easily readable and it is possible for anybody to understand what is going on
- Variables are named to improve readability (i.e. avoid calling things "a1" "x9534", "asdfa", but rather use names such as "user_speed", "daily_price", "log_returns".
- The code would work with minor modification on different data