

LABORATION 1

732G52 - HT2025

Intro

Innehåll:

- Hantera tidseriedata i R
- Göra grafer över tidseriedata
- Komponentuppdelning

Denna laboration är till för er övning på kursmaterialet. Det finns ingen obligatorisk inlämning för den. Uppgifterna utgår från R och Rstudio. Om du behöver repetition i R:

- Titta på kurshemsidan för 732G33: [här](#).
- Cheat sheets [här](#)

Material:

- Time Series for Data Science: se R-kod, dataset mm: [länk](#)
- Forecasting: Principles and Practice (FPP), tredje upplagan: [länk](#)
- Repo för kursen: [länk](#)
- Om man behöver repetition på vanlig linjär regression, se kursen Linjära modeller 1, titta tex [här](#): [länk](#)

Se sektionen "The `lm()` function" på sidan 3 för detaljer kring funktionen `lm()`.

Uppgifter

1. **ts**-objekt i R: **ts**-objekt är objekt av klassen "ts" som är speciellt skapad för att hantera tidserier på olika sätt. Det finns många olika funktioner som fungerar för eller är anpassade för ts-objekt. Gå igenom Del 1 i koden som finns här: [länk](#). Se även här: fpp2: 2.1 ts objects för information om ts-objekt.
2. Nu ska ni kolla på några av de dataset med tidserier som finns i olika R-paket som man direkt kan läsa in med funktionen `data()`. Gå igenom Del 2 i koden som länkas ovan. Här får ni bekanta er med `acf()` (som beräknar sample autocorrelation) och med `lag()` som beräknar diffar eller laggar av en vektor/tidserie.
3. Komponentuppdelning: Gå igenom koden som finns här: [länk](#).
4. Ett annat sätt att hantera tidserier med paketet **tsibble**, där objekt av klassen **tbl_ts** används för representera tidserier. "Boken Forecasting: Principles and Practice" (FPP) använder **tsibble**. Gå igenom koden här: [Introduction to tsibble](#)
5. Se följande länkar för mer information:
 - [tsibble, Tidy tools for time series](#)
 - [R package: tsibble](#)
6. Utgå från boken FPP och gå igenom och återskapa koden i följande kaptiel:
 - 2, 3.1-3.4, 4.1-4.2
7. FPP Kap 2.10 Exercises: Gör uppgift 9.
8. Ni ska nu analysera data över pappersproduktion. Data finns i filen "**pappersproduktion.csv**" och en beskrivning finns i "**sw_prod_paper_90-04.txt**". Filerna ligger här: [länk](#). Gör en komponentuppdelning på data över pappersproduktion. Analysera och tolka resultaten.

The `lm()` function

The function `lm()` will be important during the course. Check out the documentation with `?lm()`.

Note that R is an object oriented language, and the `lm()` returns objects with class “`lm`”, which has the form of a list, so you can easily fetch different parts of the object when needed. These objects have several useful generic functions connected to it:

- `coef()`: Gives the regression coefficients
- `residuals()`: calculates the residuals of the model
- `fitted()`: Gives the fitted values of the model
- `summary()`: give detail summary and inference. It will return an object of class “`summary.lm`”. `coef()` will work on this object.
- `anova()`: Calculates anova table for the model
- `predict()`: make predictions with the model for (new) data
- `plot()`: output diagnostics plots for the model

In general, for documentation for these methods run commands of the type `?summary.lm()` in the terminal. Another useful function is to use `str()` on the `lm`-object, to get detailed information about it.