3. Laste inn og lagre data

September 2, 2024

0.1 Laste inn og lagre data

Det er mulig å laste inn data fra mange forskjellige kilder og formater i R. I dette kurset skal vi se på noen av de vanligste formatene som brukes i SSB: Excel-filer (.csv og .xlsx), SAS-filer (.sas7bdat) og Parquet-filer (.parquet).

> Når vi bruker GitHub til å lagre og dele systemene våre er det viktig å passe på at sensitive data ikke legges på GitHub. Generelt skal ikke data legges på GitHub, men for å kunne lese data i dette kurset er det lagt ut noen filer som ikke inneholder sensitive data.

Når man leser inn data i R lagres disse som objekter som gis et valgfritt navn. Disse objektene brukes videre i databehandling uten at den opprinnelige filen som ble lest inn blir endret. For å skrive de ferdig behandlede dataene tilbake til en fil gjøres dette eksplisitt med egne funksjoner.

[ ]:
```
getwd()
setwd("./presentasjoner")
```

0.1.1 CSV

CSV-filer (Comma Separated Values) er en type tekstfil som brukes til å lagre tabulære data, som oftest fra regneark eller databaser. Hver linje i filen representerer en rad i tabellen, og verdiene i hver rad er adskilt med et komma (eller noen ganger et annet skilletegn som semikolon eller tabulator). Den første linjen i en CSV-fil inneholder ofte kolonnenavnene. De viktigste egenskapene man må vite om filen man leser inn er hvilken separator som brukes (f.eks. `;`, `,`), hvilket desimaltegn som brukes (f.eks. `,`, `.`) og hvilket tegnsett (encoding) som brukes (f.eks. `UTF-8`, `latin1`).

- `read.csv()`: funksjon som leser inn CSV-filer som en data frame
- `write.csv2()`: brukes til å lagre en data frame til en semikolonseparart CSV-fil

[ ]:
```
sykemelding <- read.csv("../data/sykemelding.csv", sep = ";", dec = ",",␣
↪encoding = "UTF-8")
```

[ ]:
```
kommunedata <- read.csv("../data/kommunedata.csv", sep = ",")
```

[ ]:
```
# write.csv2(kommunedata, "../data/kommunedata_ny.csv", row.names = FALSE)
```

0.1.2 XLSX

- `read_excel()`: funksjon som leser inn XLSX-filer som en data frame

- `write.xlsx()`: funksjon for å lagre en data frame til som en XLSX-fil

[ ]:
```
fylkesinndeling <- readxl::read_excel("../data/fylkesinndeling.xlsx")
```

[ ]:
```
# openxlsx::write.xlsx(fylkesinndeling, file = "../data/fylkesinndeling_ny.
↪xlsx",
#                      rowNames = FALSE,
#                      showNA = FALSE,
#                      overwrite=T)
```

0.1.3 SAS

Den mest brukte pakken for å lese SAS-filer i R er haven. Denne pakken er en del av tidyverse-familien og gir en enkel måte å importere data fra SAS, Stata, og SPSS. Det anbefales ikke å lagre obejkter i R som SAS-filer, men dersom dette må gjøres anbefales det å lagre en CSV-fil som kan leses inn i SAS.

- `read_sas()`: funksjon som leser inn SAS-filer som en data frame

[ ]:
```
trygd <- haven::read_sas("../data/trygd.sas7bdat")
```

0.1.4 Parquet

Parquet-filer er en kolonneorientert lagringsfilformat som er optimalisert for rask lesing og lagring av store datamengder.

- `read_parquet()`: funksjon som leser inn parquet-filer som en data frame
- `write_parquet()`: funksjon for å lagre en data frame til som en parquet-fil

[ ]:
```
befolkning_per_fylke <- arrow::read_parquet("../data/befolkning_per_fylke.
↪parquet")
```

[ ]:
```
# arrow::write_parquet(befolkning_per_fylke, "../data_ny/
↪befolkning_per_fylke_ny.parquet")
```

0.1.5 Data fra API-er

I dette kurset vil det ikke gjennomgås hvordan å laste ned data fra API-er, men her er det lagt til eksempler på hvordan man laster ned data fra statistikkbanken og KLASS.

[ ]:
```
befolkning <- PxWebApiData::ApiData(07459,
                                    ContentsCode = T,
                                    Region = T,
                                    Kjonn = T,
                                    Alder = T,
                                    Tid = "2024")[[2]]
```

KLASS Klass er SSBs system for dokumentasjon av kodeverk (klassifikasjoner og kodelister). Klassifikasjoner er «offisielle» kodeverk, og alle klassifikasjoner i Klass har navn som begynner med «Standard for..», f.eks. Standard for sivilstand. I en klassifikasjon skal kategoriene være gjensidig utelukkende og uttømmende, dvs. at klassifikasjonen inneholder alle kategorier som tilhører området klassifikasjonen dekker.

Data fra KLASS API kan leses direkte inn i R med pakken `klassR`.

[ ]:
```
fylkesinndeling <- klassR::GetKlass(104, date = "2024-01-01")
```