

# Case 1

Phu Henry Luz Nguyen

Denne oppgaven er laget av Even S. Hvinden og oppdatert av Derek J. Clark. Sistnevnte er ansvarlig for eventuelle feil og mangler.

## Instruksjoner

Denne oppgaven skal løses interaktivt i RStudio ved å legge inn egen kode og kommentarer. Det ferdige dokumentet lagres med kandidatnummeret som navn `[kandidatnummer]_SOK1004_C1_H23.qmd` og lastes opp på deres GitHub-side. Hvis du har kandidatnummer 43, så vil filen hete `43_SOK1004_C1_H23.qmd`. Påse at koden kjører og at dere kan eksportere besvarelsen til pdf. Dere leverer lenken til GitHub-repositoriet i Canvas (under oppgaver).

## Bakgrunn

Vi skal analysere utviklingen i bruttonasjonalprodukt (BNP) per person i Norge. Vi bruker data Statistisk Sentralbyrå (SSB), tabell “09842: BNP og andre hovedstørrelser (kr per innbygger), etter statistikkvariabel og år”. Tabellen inneholder årlige data på BNP per innbygger, fra 1970 til 2022.

## I. API, visualisering

SSB gir oss tilgang til sine data via en [API](#) (*Application Programming Interface*), programvare som lar to applikasjoner kommunisere med hverandre. SSB tilbyr en API med [ferdige datasett](#). Her er det om lag 250 kontinuerlig oppdaterte datasett med en fast URL over de mest brukte tabellene i Statistikkbanken.

For å få tilgang til tabellen med bruttonasjonalprodukt må vi benytte tjenesten [PxWebApi](#). Her finner du en [API konsoll](#) med en søkefunksjon. Prøv å søk på “**bnp**” og merk forslaget: tabell 09842. Søk på denne, og noter URL-en. Den vil vi bruke etterpå.

Til å laste ned dataene skal vi bruke en R-pakke, [PxWebApiData](#), som SSB har laget. I første omgang skal vi bruke funksjonen `ApiData()`. Syntaksen er ikke den samme som i `tidyverse`,

og har noen litt uvante egenskaper, herunder lagring i tegnformat og en kombinasjon av norsk og engelsk.

**Tips:** Det er typisk instruktivt å se på [eksempel på bruk](#). Da har man et intuitivt utgangspunkt for hvordan koden kan brukes.

Jeg vil nå vise dere trinnvis hvordan å laste ned dataene. Formålet er å gi dere en idé på hvordan man kan lære seg å bruke en ny pakke eller funksjon. Vi begynner med å laste inn nødvendige pakker:

**NB!** Du må installere `PxWebApiData` først. Kjør kommandoen `install.packages("PxWebApiData")` i konsollen. Det må kun gjøres én gang.

Vi bruker funksjonen `ApiData()` til å hente tabell 09842. Som notert ovenfor fant vi URL-en ved hjelp av søkefunksjonen til SSB. Først prøver vi å laste ned dataene direkte, uten ytterligere tilvalg, og tar en titt på hva vi får.

```
$`09842: BNP og andre hovedstørrelser (kr per innbygger), etter statistikkvariabel og år`
      statistikkvariabel  år  value
1      Bruttonasjonalprodukt 1970   23616
2      Bruttonasjonalprodukt 2021   778730
3      Bruttonasjonalprodukt 2022  1020805
4 Konsum i husholdninger og ideelle organisasjoner 1970   12283
5 Konsum i husholdninger og ideelle organisasjoner 2021  298958
6 Konsum i husholdninger og ideelle organisasjoner 2022  331008
7      MEMO: Bruttonasjonalprodukt. Faste 2015-priser 1970   215602
8      MEMO: Bruttonasjonalprodukt. Faste 2015-priser 2021   627433
9      MEMO: Bruttonasjonalprodukt. Faste 2015-priser 2022   642200
```

```
$dataset
  ContentsCode  Tid  value
1      BNP 1970   23616
2      BNP 2021   778730
3      BNP 2022  1020805
4 KonsumHIO 1970   12283
5 KonsumHIO 2021  298958
6 KonsumHIO 2022  331008
7      MEMOBNP 1970   215602
8      MEMOBNP 2021   627433
9      MEMOBNP 2022   642200
```

**Merk følgende:** `df` inneholder to datasett i formatet `data.frame`. Datasettene heter "09842: BNP og andre hovedstørrelser (kr per innbygger), etter statistikkvariabel og år" og `dataset`. Datasettene inneholder 9 verdier av 3 variabler. Variabelen `value` er

identisk. Variablene `år` og `Tid` inneholder de identiske verdiene "1970", "2021" og "2022". Merk at disse er i tegnformat `<chr>` (derav anførselstegnene) og ikke en numerisk verdi, for eksempel `<dbl>`. Variabelen `statistikkvariabel` og `ContentsCode` inneholder henholdsvis verdiene BNP, KonsumHIO MEMOBNP og Bruttonasjonalprodukt, Konsum i husholdninger og ideelle organisasjoner og MEMO: Bruttonasjonalprodukt. Faste 2015-priser.

Vi har altså ikke fått hele tabell 09842, men verdiene for tre statistikkvariabler over tre tidsperioder, lagret med forskjellige variabelnavn og verdier.

Det vi trenger er **metadata**: Informasjon som beskriver innholdet i dataene, slik at vi kan filtrere API-spørringen. Kjør følgende kode.

Åpner vi listen `metadata` fra minnet så kan vi se nærmere på den i øvre venstre vindu i Rstudio. Her ser vi to lister kalt `[[1]]` og `[[2]]`. Listene beskriver variablene vi kan filtrere på. Liste `[[1]]` har fire variable: `code`, `text`, `values`, og `valueTexts`. Alle variablene er `<chr>`. Liste `[[2]]` har de samme foregående fire variablene samt en variabel `time`.

- `code` viser navnene på variablene vi bruker i funksjonen `ApiData()` for å filtrere. Den tar verdiene `ContentsCode` og `Tid`. Legg merke til at utviklerne i SSB her blander norsk og engelsk.
- `text` er en unik tekstverdi tilknyttet verdien på `code` som forklarer hva vi ser på. Den tar verdien `statistikkvariabel` og `år`. Vi kan altså filtrere på `statistikkvariabel` og `år`.
- `values` viser hvilke verdier av `statistikkvariabel` og `år` vi kan velge, med henholdsvis 6 og 53 forskjellige verdier. Du vil kjenne igjen tre av hver fra den første spørringen ovenfor.
- `valueTexts` gir en unik tekstverdi tilknyttet verdien på `values` som forklarer oss hva vi ser på. For `Tid` og `år` er de identiske, men for `ContentsCode` og `statistikkvariabel` får vi en mer fullstendig forklaring.
- `time` er en logisk variabel, og tar derfor to verdier: `TRUE` og `FALSE`. I dette tilfellet indikerer den at variabelen `Tid` måler tid, hvilket gjør at funksjonene i pakken vil behandle `Tid` på en annen måte enn en `statistikkvariabel`.

Vi har nå informasjonen vi trenger til å laste ned BNP-tall mellom 1970 og 2022. Jeg velger å ta BNP med både løpende og faste priser.

På venstre side av likhetstegnet bruker vi `code` fra `metadata`. På høyre side velger vi verdier fra `values`. Merk at jeg bruker funksjonen `paste()` for å konvertere numeriske verdier, for eksempel `<dbl>` til tegn `<chr>`.

La oss rydde i data. Det er tre ting å ta tak i:

1. `df` lagrer informasjonen i to tabeller med samme informasjon, som vist over. Det er unødvendig.
2. Årstallene er lagret som tegn, `<chr>`. Disse skulle heller være heltall, `<int>`.

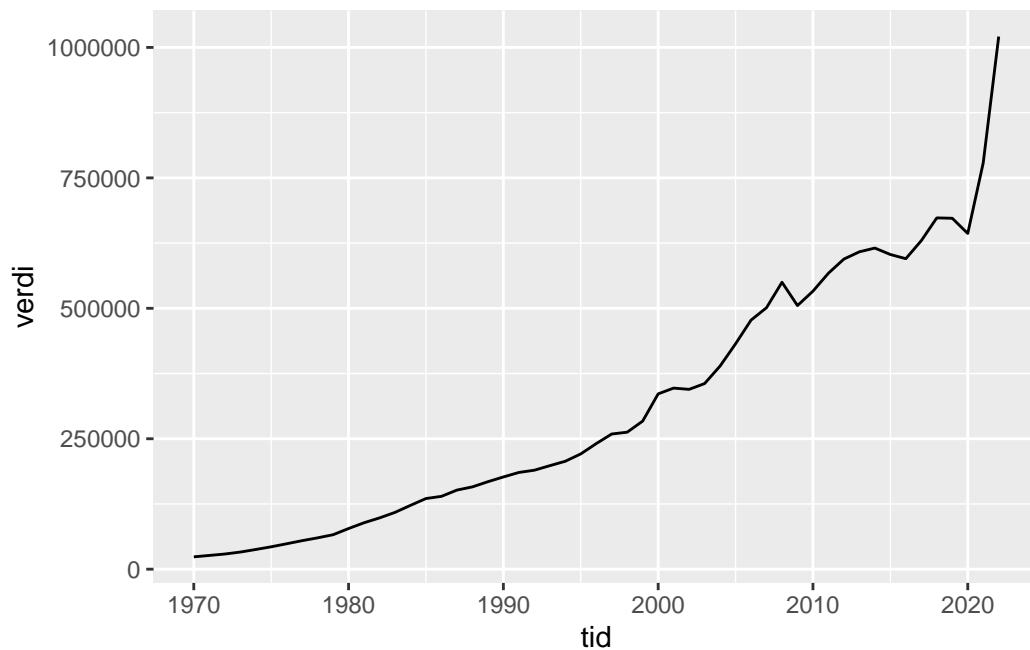
3. Formatet `data.frame` er underlegent `tibble`.

### Oppgave Ia: Rydd i data

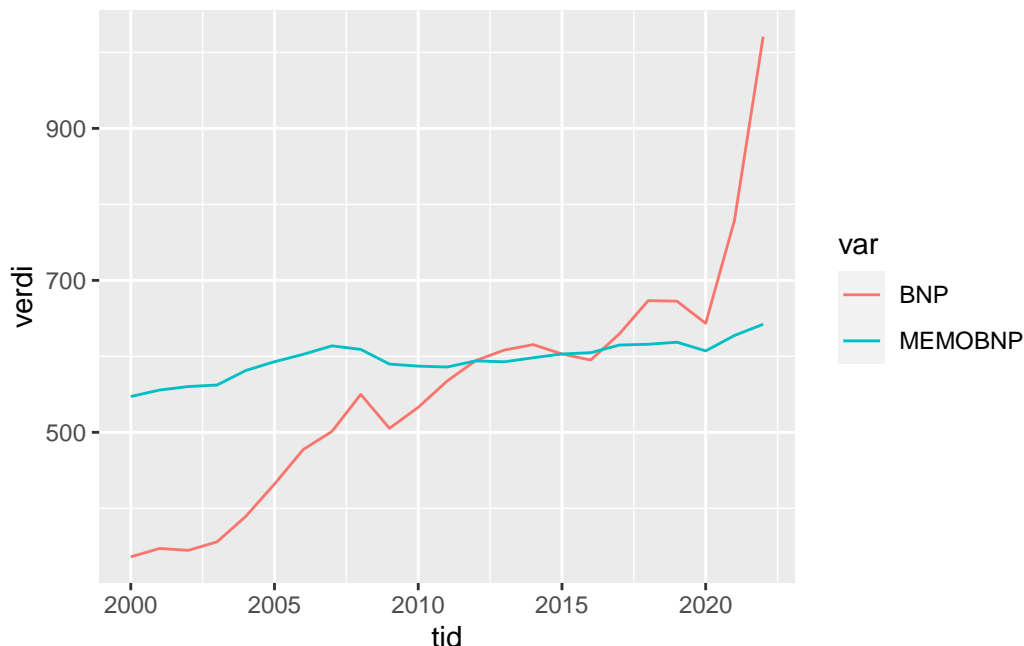
Skriv kode som lagrer dataene som én tibble med anstendige variabelnavn og årstall som heltall. Fremover bruker jeg “var”, “tid”, og “verdi” for “statistikkvariabel”, “Tid”, og “value”.

### Oppgave Ib: Lag en figur

Følgende kode skaper en enkel figur.



Lag en pen figur som viser BNP i tusener av kroner per person, i både løpende og faste priser, mellom 2000 og 2022. Skriv en tydelig forklaring og tolkning av figuren. Hvordan har inntektene utviklet seg? Forklar forskjellen mellom BNP i løpende og faste priser. Til hvilke formål er de mest relevante?



[Besvar oppgave 1b her]

BNP viser markedsverdien av alle varer og tjenester.

MEMOBNP er liknende BNP men måles med priser fra et bestemt år. Dette vil hjelpe oss å finne volumendringer mellom tidsperioder.

Grafen viser en liten vekst i produksjonsmengde og en stor endring i nominal BNP. Det vil si at volumet av produksjon har ikke endret seg særlig men at den økonomiske verdien til produktet har økt stadig.

## II. Transformasjon, visualisering

Våre data er en tidsserie, hvilket betyr at rekkefølgen i observasjonene er ordnet etter tid. Vi skal nå regne prosentvis, årlig endring. La  $x_t$  være BNP i år  $t$ . For eksempel vil  $x_{1970}$  være 23616.

Den årlige endringen i BNP fra år  $t - 1$  til  $t$  er gitt ved  $x_t - x_{t-1}$ . I samfunnsøkonomi er det vanlig å betegne dette som  $\Delta x_t := x_t - x_{t-1}$ . Tegnet  $\Delta$  er den greske bokstaven delta og betegner differanse. For eksempel vil  $\Delta x_{1971} = 26363 - 23616 = 2747$  kroner.

I mange tilfeller er vi interesserte i relativ vekst: Hvor mye økte BNP, relativt til hva den var i utgangspunkt? Den mest brukte enheten er hundredeler eller prosentvis endring, gitt ved  $100 \times \Delta x_t / x_{t-1}$ . For eksempel var den prosentvise endringen i BNP i 1971  $100 \times \Delta x_{1971} / x_{1970} = 100 \times (2747 / 23616) \approx 11.6$ , hvor  $\approx$  betegner "omtrent lik" da jeg viser svaret med kun én

desimal. Tilsvarende kan man skrive at  $\Delta x_{1971}/x_{1970} = 2747/23616 \approx 0.116 = 11.6\%$ , hvor tegnet % betegner at beløpet oppgis i hundredeler eller prosent.

## Oppgave IIa: Omorganisere datasett med `pivot_wider()`

Vi skal lage to variable `dBNP` og `dMEMOBNP` som viser relativ endring i BNP og MEMOBNP. Til dette formålet skal vi bruke kommandoene `pivot_wide()` og `pivot_long()` til å omorganisere dataene. Jeg anbefaler dere først å lese [kapittel 12.3](#) i pensum. Betrakt følgende kode.

Beskriv konkret hva koden gjorde. Sammenlign `df` og `df_wide`.

Koden gjorde som at BNP og MEMOBNP står ved siden hverandre i sine egne kolonner istedenfor etter hverandre i samme kolonne.

*[Besvar oppgave IIa her]*

## Oppgave IIb: Beregn vekst

Til å beregne endring er funksjonen `lag()` meget nyttig. I denne konteksten er begrepet *lag* et engelsk verb som beskriver foregående observasjon. Bruker vi funksjonen `lag()` på en variabel (kolonne) så returnerer den en ny kolonne hvor verdien er lik foregående observasjon. Betrakt følgende kode:

```
# A tibble: 53 x 5
  tid    BNP  LBNP MEMOBNP LMEMOBNP
<int> <int> <int>   <int>   <int>
1  1970 23616    NA  215602     NA
2  1971 26363 23616  226242  215602
3  1972 29078 26363  236487  226242
4  1973 32805 29078  245483  236487
5  1974 37734 32805  253534  245483
6  1975 42884 37734  264625  253534
7  1976 48711 42884  278730  264625
8  1977 54652 48711  289103  278730
9  1978 60090 54652  299145  289103
10 1979 66069 60090  311164  299145
# i 43 more rows
```

Hvis vi bruker den matematiske notasjonen diskutert tidligere så har vi nå kolonner med  $x_t$  (BNP, MEMOBNP) og  $x_{t-1}$  (LBNP, LMEMOBNP).

Bruk funksjonen `mutate()` til å lage en ny variabel med relativ endring i BNP og MEMOBNP i `df_wide` og lagre de som `DBNP` og `DMEMOBNP`.

```
# A tibble: 53 x 9
  tid   BNP  LBNP  DBNP prosDifBNP MEMOBNP LMEMOBNP DMEMOBNP prosDifMEMOBNP
  <int> <int> <int> <int>      <dbl>   <int>   <int>   <int>      <dbl>
1  1970 23616    NA    NA      NA     215602    NA     NA      NA
2  1971 26363 23616 2747    11.6    226242  215602  10640    4.94
3  1972 29078 26363 2715    10.3    236487  226242  10245    4.53
4  1973 32805 29078 3727    12.8    245483  236487   8996    3.80
5  1974 37734 32805 4929    15.0    253534  245483   8051    3.28
6  1975 42884 37734 5150    13.6    264625  253534  11091    4.37
7  1976 48711 42884 5827    13.6    278730  264625  14105    5.33
8  1977 54652 48711 5941    12.2    289103  278730  10373    3.72
9  1978 60090 54652 5438     9.95   299145  289103  10042    3.47
10 1979 66069 60090 5979     9.95   311164  299145  12019    4.02
# i 43 more rows
```

### Oppgave IIc: Omorganisere datasett med pivot\_longer()

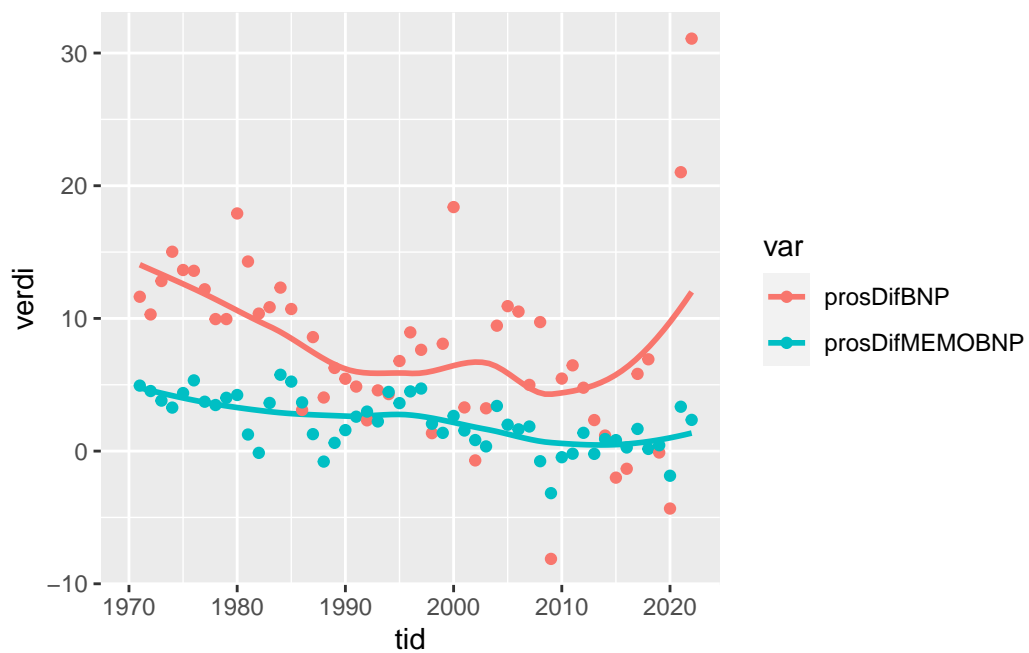
Bruk nå funksjonen `pivot_longer()` til å transformere `df_wide` til det opprinnelige formatet, altså med variablene `var` og `verdi`. Kall den transformerte tabellen for `df_long`.

**NB!** Husk å bruk anførselstegn ("`[variabelnavn]`") når du definerer nye variable i `pivot_longer()`.

### Oppgave IId: Figur med vekst

Lag en pen figur med prosentvis vekst i nominelt og reelt BNP per person fra 1970 til 2022. Finnes det observasjoner med negativ vekst i reell BNP? Hva skyldes dette?

**Merknad:** Det er en del støy i data. Prøv å kombinere `geom_point()` og `geom_smooth()` for å få et bedre inntrykk av den langsiktige utviklingen.



```
# A tibble: 424 x 3
  tid var      verdi
  <int> <chr>    <dbl>
1  1970 BNP      23616
2  1970 LBNP      NA
3  1970 DBNP      NA
4  1970 prosDifBNP  NA
5  1970 MEMOBNP  215602
6  1970 LMEMOBNP   NA
7  1970 DMEMOBNP   NA
8  1970 prosDifMEMOBNP NA
9  1971 BNP      26363
10 1971 LBNP      23616
# i 414 more rows
```