

# 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_C5_H22.qmd` og lastes opp på deres GitHub-side. Hvis du har kandidatnummer 43, så vil filen hete `43_SOK1004_C5_H22.qmd`. Påse at koden kjører og at dere kan eksportere besvarelsen til pdf. Lever så lenken til GitHub-repositoriet i Canvas.

## Bakgrunn, læringsmål

I [seminar 3, oppgave 3](#) leste dere [kapittel 1 i NOU 2015:1](#) "Produktivitet - grunnlag for vekst og velferd." I denne oppgaven skal vi oppdatere tall brukt i analysen. Gjennom arbeidet vil dere repetere de grunnleggende ferdighetene i bruk av API, `tidyverse` og `ggplot` som vi har arbeidet med gjennom kurset. Dere vil også få anledning til å reflektere rundt sammenhengene i nasjonalregnskapet og å grave litt i statistikkbanken hos SSB.

## Last inn pakker

```
# output | false
rm(list=ls())
library(tidyverse)
```

```
-- Attaching packages ----- tidyverse 1.3.2 --
v ggplot2 3.3.6      v purrr   0.3.4
v tibble  3.1.8      v dplyr   1.0.10
v tidyr    1.2.1      v stringr 1.4.1
v readr    2.1.2      v forcats 0.5.2
-- Conflicts ----- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```
library(rjstat)
```

Attaching package: 'rjstat'

The following object is masked from 'package:dplyr':

`id`

```
library(httr)
```

## Oppgave I: Timeverksproduktivitet i markedsrettede fastlandsnæringer

Reproduser en oppdatert versjon av Figur 1.1 "Timeverksproduktivitet, markedsrettede fastlandsnæringer" i rapporten. Her er produktivitet målt som bruttoprodukt i markedsrettet produksjon i Fastlands-Norge per timeverk, utenom bolig.

## Oppgave Ia

Hvorfor trekker Produktivitetskommisjon fra produksjonen av boligjenester, offentlig etterspørsel, og inntekter fra olje og gass når de beregner produktivitet?

*[Svar på oppgave Ia her]*

Da boligjenester og offentlig etterspørsel hovedsaklig ikke omsettes på markeder, og vil derfor være vanskelig å vurdere korrekt, og potensielt bli undervurdert. Offentlig etterspørsel består også av mye det vil være vanskelig å ha produktivitetsvekst i, da det er arbeidsintensive oppgaver, det er vanskelig å få studenter kjappere gjennom skole og universitet, og barn trenger like mye oppmerksomhet og oppsyn i barnehager og liknende uavhengig av teknologisk fremgang.

Olje og gass består hovedsaklig av grunnrente, produksjonen rundt utvinning har lite å gjøre med verdien av naturressursene. Den volatile prisen på olje og gass, samt økte utvinningskostnader som følge av behovet for å benytte felt det er mindre effektivt å drive utvinning fra, vil kunne gi store utslag på tallene, uten å gi oss nyttig nok informasjon. Og det er mer hensiktsmessig å sammenlikne oss med andre land uten disse tallene, da vi vet disse ressursene tar slutt, og vi må være klare for det ved å tenke fremover i tid.

## Oppgave Ib

Skriv kode som bruker en JSON-spørring til å lage en oppdatert versjon av figuren, med observasjoner fra 1996 til 2021. Beregn gjennomsnittlig årlig endring for perioden 1996 - 2005 og, 2005-2013, og 2006 - 2021. Står konklusjonen i rapporten seg?

**Merknad.** Det er tidvis store revisjoner i nasjonalregnskapet. Du må regne med at individuelle tall på årlig endring kan avvike en god del, opp til et halvt prosentpoeng, mens de gjennomsnittlige veksttallene burde avvike noe mindre.

**Tips:** Du henter data på timeverk og bruttoprodukt fra henholdsvis tabell 09174 og 09170.

```
rm(list = ls())

# Løs oppgave 1b her

suppressPackageStartupMessages(library(tidyverse))
library(lubridate)
```

Attaching package: 'lubridate'

The following objects are masked from 'package:base':

date, intersect, setdiff, union

```
library(rjstat)
library(janitor)
```

Attaching package: 'janitor'

The following objects are masked from 'package:stats':

chisq.test, fisher.test

```
library(gdata)
```

gdata: Unable to locate valid perl interpreter

gdata:

gdata: read.xls() will be unable to read Excel XLS and XLSX files

gdata: unless the 'perl=' argument is used to specify the location of a

gdata: valid perl intrpreter.

gdata:

gdata: (To avoid display of this message in the future, please ensure

gdata: perl is installed and available on the executable search path.)

gdata: Unable to load perl libraries needed by read.xls()

gdata: to support 'XLX' (Excel 97-2004) files.

gdata: Unable to load perl libraries needed by read.xls()

gdata: to support 'XLSX' (Excel 2007+) files.

gdata: Run the function 'installXLSXsupport()'

gdata: to automatically download and install the perl

gdata: libraries needed to support Excel XLS and XLSX formats.

Attaching package: 'gdata'

The following objects are masked from 'package:dplyr':

combine, first, last

The following object is masked from 'package:purrr':

keep

The following object is masked from 'package:stats':

nobs

The following object is masked from 'package:utils':

object.size

The following object is masked from 'package:base':

startsWith

```
library(httr)

url <- "https://data.ssb.no/api/v0/no/table/09174/"

query <- '{
  "query": [
    {
      "code": "NACE",
      "selection": {
        "filter": "item",
        "values": [

          "nr23fn"

        ]
      }
    },
    {
      "code": "ContentsCode",
      "selection": {
        "filter": "item",
        "values": [
          "BruttoprodTimev"
        ]
      }
    },
    {
      "code": "Tid",
      "selection": {
        "filter": "item",
        "values": [
          "1996",
          "1997",
          "1998",
          "1999",
          "2000",
          "2001",
          "2002",
          "2003",
          "2004",
          "2005",
          "2006",
          "2007",
          "2008",
          "2009",
          "2010",
          "2011",
          "2012",
          "2013",
```

```

        "2014",
        "2015",
        "2016",
        "2017",
        "2018",
        "2019",
        "2020",
        "2021"
      ]
    }
  }
],
"response": {
  "format": "json-stat2"
}
}'

```

```

hent_indeks.tmp <- url %>%
  POST(body = query, encode = "json")

```

```

df <- hent_indeks.tmp %>%
  content("text") %>%
  fromJSONstat() %>%
  as_tibble()

```

```
url <- "https://data.ssb.no/api/v0/no/table/09170/"
```

```

query <- '{
  "query": [
    {
      "code": "NACE",
      "selection": {
        "filter": "item",
        "values": [
          "nr23fn",
          "pub2X68B"
        ]
      }
    },
    {
      "code": "ContentsCode",
      "selection": {
        "filter": "item",
        "values": [
          "BNPB"
        ]
      }
    },
    {
      "code": "Tid",
      "selection": {
        "filter": "item",
        "values": [
          "1996",

```

```

        "1997",
        "1998",
        "1999",
        "2000",
        "2001",
        "2002",
        "2003",
        "2004",
        "2005",
        "2006",
        "2007",
        "2008",
        "2009",
        "2010",
        "2011",
        "2012",
        "2013",
        "2014",
        "2015",
        "2016",
        "2017",
        "2018",
        "2019",
        "2020",
        "2021"
      ]
    }
  }
],
"response": {
  "format": "json-stat2"
}
}'

```

```

hent_indeks.tmp <- url %>%
  POST(body = query, encode = "json")

```

```

df_2 <- hent_indeks.tmp %>%
  content("text") %>%
  fromJSONstat() %>%
  as_tibble()

```

```

#figur
df %>%
  ggplot(aes(x = år, y = value)) +
  geom_col(fill="blue", col="grey") +

  labs(y = " ", x = " ", title = "Timeverksproduktivitet, markedsrettede fastlandsnæringer",
  subtitle = " ") +

  scale_y_continuous(limits = c(-3, 5), expand = c(0, 0), breaks = c(-3,-2,-1,0,1,2,3,4,5), se

#Lage linje

```

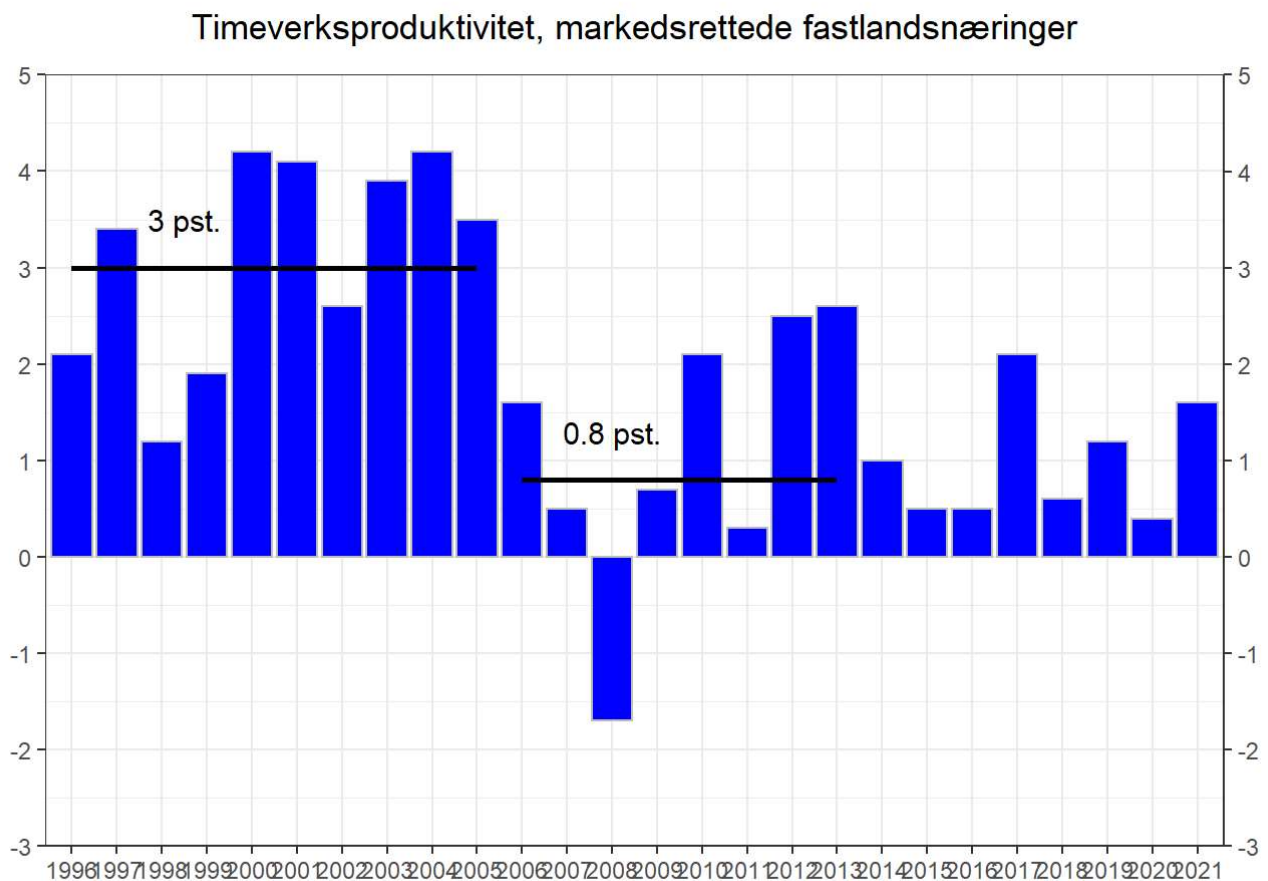
```

annotate("segment", x = 1, xend = 10, y = 3, yend = 3, colour = "black", size = 1) +
annotate("segment", x = 11, xend = 18, y = 0.8, yend = 0.8, colour = "black", size = 1) +

#Skriv tekst
annotate("text", x = 3.5, y = 3.5, label = "3 pst.", size = 4) +
annotate("text", x = 13, y = 1.3, label = "0.8 pst.", size = 4) +

theme_bw() +
#Bestemme posisjon på tittel
theme(plot.title = element_text(hjust = 0.5, vjust = -3))

```



```

df_3 <- df #>%
  #group_by(næring, gruppe) %>%

#endre navn
df_3["år"][df_3["år"] == "1996"] <- "1996 - 2005"
df_3["år"][df_3["år"] == "1997"] <- "1996 - 2005"
df_3["år"][df_3["år"] == "1998"] <- "1996 - 2005"
df_3["år"][df_3["år"] == "1999"] <- "1996 - 2005"
df_3["år"][df_3["år"] == "2000"] <- "1996 - 2005"
df_3["år"][df_3["år"] == "2001"] <- "1996 - 2005"
df_3["år"][df_3["år"] == "2002"] <- "1996 - 2005"
df_3["år"][df_3["år"] == "2003"] <- "1996 - 2005"
df_3["år"][df_3["år"] == "2004"] <- "1996 - 2005"
df_3["år"][df_3["år"] == "2005"] <- "1996 - 2005"

```

```
df_3["år"][df_3["år"] == "2006"] <- "2006 - 2013"
df_3["år"][df_3["år"] == "2007"] <- "2006 - 2013"
df_3["år"][df_3["år"] == "2008"] <- "2006 - 2013"
df_3["år"][df_3["år"] == "2009"] <- "2006 - 2013"
df_3["år"][df_3["år"] == "2010"] <- "2006 - 2013"
df_3["år"][df_3["år"] == "2011"] <- "2006 - 2013"
df_3["år"][df_3["år"] == "2012"] <- "2006 - 2013"
df_3["år"][df_3["år"] == "2013"] <- "2006 - 2013"

df_3["år"][df_3["år"] == "2014"] <- "2006 - 2021"
df_3["år"][df_3["år"] == "2015"] <- "2006 - 2021"
df_3["år"][df_3["år"] == "2016"] <- "2006 - 2021"
df_3["år"][df_3["år"] == "2017"] <- "2006 - 2021"
df_3["år"][df_3["år"] == "2018"] <- "2006 - 2021"
df_3["år"][df_3["år"] == "2019"] <- "2006 - 2021"
df_3["år"][df_3["år"] == "2020"] <- "2006 - 2021"
df_3["år"][df_3["år"] == "2021"] <- "2006 - 2021"
```

*#tungvin måte å regne ut gjennomsnittet for årsgruppene*

```
df_4 <- df_3 %>%
  filter(år == "1996 - 2005") %>%
  group_by(næring, statistikkvariabel, år) %>%
  summarise(value = sum(value, na.rm=TRUE)/10)
```

`summarise()` has grouped output by 'næring', 'statistikkvariabel'. You can override using the `.groups` argument.

```
df_5 <- df_3 %>%
  filter(år == "2006 - 2013") %>%
  group_by(næring, statistikkvariabel, år) %>%
  summarise(value = sum(value, na.rm=TRUE)/8)
```

`summarise()` has grouped output by 'næring', 'statistikkvariabel'. You can override using the `.groups` argument.

```
df_6 <- df_3 %>%
  filter(år == "2006 - 2021") %>%
  group_by(næring, statistikkvariabel, år) %>%
  summarise(value = sum(value, na.rm=TRUE)/8)
```

`summarise()` has grouped output by 'næring', 'statistikkvariabel'. You can override using the `.groups` argument.

```
df_7 <- rbind(df_5, df_6)

df_7["år"][df_7["år"] == "2006 - 2013"] <- "2006 - 2021"

df_7 <- df_7 %>%
  group_by(næring, statistikkvariabel, år) %>%
  summarise(value = sum(value, na.rm=TRUE)/2)
```



`summarise()` has grouped output by 'næring', 'statistikkvariabel'. You can override using the `.groups` argument.

```
df_8 <- rbind(df_4, df_5, df_7)

#####

#figur
df_8 %>%
  ggplot(aes(x = år, y = value)) +
  geom_col(fill="blue", col="grey") +

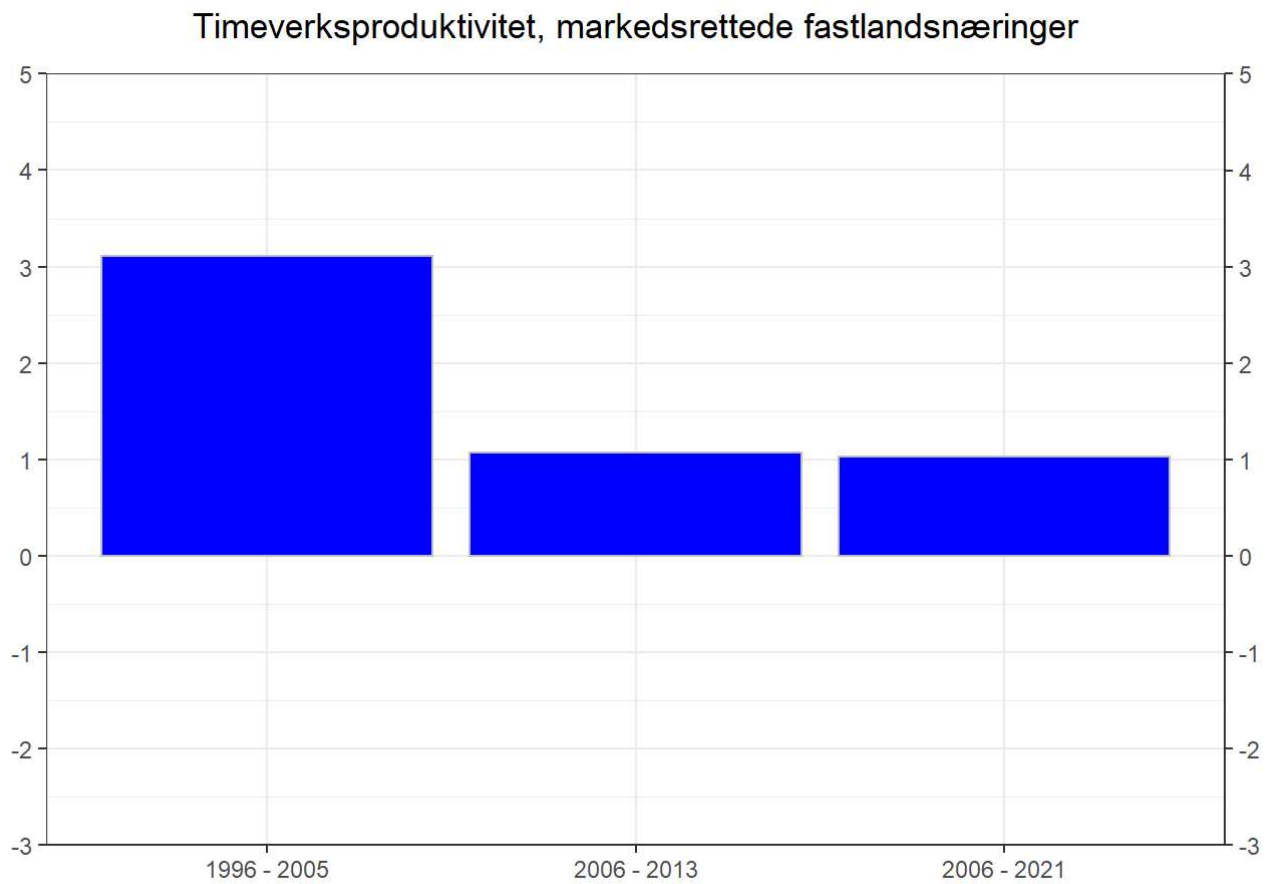
  labs(y = " ", x = " ", title = "Timeverksproduktivitet, markedsrettede fastlandsnæringer",
        subtitle = " ") +

  scale_y_continuous(limits = c(-3, 5), expand = c(0, 0), breaks = c(-3,-2,-1,0,1,2,3,4,5), se

#Lage linje
#annotate("segment", x = 1, xend = 10, y = 3, yend = 3, colour = "black", size = 1) +
#annotate("segment", x = 11, xend = 18, y = 0.8, yend = 0.8, colour = "black", size = 1) +

#Skriv tekst
#annotate("text", x = 3.5, y = 3.5, label = "3 pst.", size = 4) +
#annotate("text", x = 13, y = 1.3, label = "0.8 pst.", size = 4) +

theme_bw() +
#Bestemme posisjon på tittel
theme(plot.title = element_text(hjust = 0.5, vjust = -3))
```



Jeg klarte ikke å få ekstrapolert/transformert riktige tall. Jeg synes ikke det kom frem godt nok av NOU hvilke tall som var blitt brukt. Når jeg prøvde å trekke bolig fra bruttoprodukt, og gjøre det om til en prosent jeg kunne gange bruttoprodukt per utførte timesverk i prosent med, fikk jeg bare feil tall.

Men i følge de tallene jeg har her, er jo veksten fra 2006-2021 rimelig lik veksten fra 2006-2013, så konklusjonen i rapporten holder.