

Assingment 3; Pendling

Arnstein Gjestland

Table of contents

Innledning	1
Kommuner i Sør-Rogaland	2
Hente data fra SSB	3
Kommunesammenslåingene	5
Bosted Nord-Jæren	5
Arbeidssted Nord-Jæren	7
Totalt antall arbeidstakere i hele landet per år	8
Andel pendlere	11
Bor og jobber i samme kommune på Nord-Jæren	13
Internpendling et utvalg kommuner	15
Definerer funksjoner for plot og tabell	15
Hvor jobber folk bosatt i Stavanger kommune?	16
Hvor jobber folk bosatt i Sandnes kommune?	16
Hvor jobber folk bosatt i Sola kommune?	18
Hvor jobber folk bosatt i Randaberg kommune?	18
Hvor jobber folk bosatt i Klepp kommune?	18
Hvor jobber folk bosatt i Time kommune?	18
Hvor jobber folk bosatt i Gjesdal kommune?	18
Hvor jobber folk bosatt i Hå kommune?	18
Hvor jobber folk bosatt i Bjerkreim kommune?	19
Hvor jobber folk bosatt i Strand kommune?	19
Hvor stor pendling er det fra resten av landet til Nord-Jæren?	19
Pendlematriser	19
2000	19
2010	19
2020	19
2023	19

Innledning

I denne oppgaven skal vi studere pendle-mønster for en norsk region. Vi vil studere et utvalg kommuner i Sør-Rogaland (Nord-Jæren) i perioden 2000 t.o.m. 2023. Dataene vil vi i hovedsak hente fra SSB sin statistikktabell 03321, «*Sysselsatte (15-74 år), etter arbeidssteds- og bostedskommune. Pendlingsstrømmer. 4. kvartal (K) 2000 - 2023*». Vi skal benytte dataene til å generere grafikk som viser utviklingen i jobb-

pendling over tid for de ulike kommunene. Grafikken vil vi generere vha `geom_line()` fra `ggplot2` pakken (del av `tidyverse`). For `geom_line()` trenger vi dataene i såkalt «long-format». Dette bør vi ta hensyn til alt fra starten av i vår behandling av pendle-dataene.

I tillegg skal vi generere såkalte pendle-matriser («origin-destination-matrix»). I en pendle-matrise vil hver **rekke** angi hvor de som **bor i en kommune** arbeider. Tilsvarende angir en **kolonne** hvor de som **jobber i en kommune** bor.

I Sør-Rogaland har det også vært kommunesammenslåinger som har gitt oss Nye-Stavanger og Nye-Sandnes fra 2020. Vi vil generere konsistente pendle-data for perioden 2000 t.o.m. 2023 som korresponderer med denne nye kommunestrukturen.

Dataene fra SSB vil vi hente direkte inn i R via pakken `PxWebApiData`, som gir oss tilgang til SSB via en api.

Vi vil hente kategorien «Alle kommuner» fra tabell 03321 og selv ta oss av aggregeringen av data til «Nye-Stavanger» og «Nye-Sandnes».

Oppgaven er laget for å gi trening i å hente data direkte fra SSBs servere vha. `PxWeb` api-en (pakken `PxWebApiData`, funksjonen `ApiData12()` anbefales), benytte funksjonene fra chp. 12 (`pivot_longer()`, `pivot_wider()` etc.) og lage et *tidy* datasett. Pendlematriser er ikke *tidy* så vi lager et stort *tidy* datasett for alle kommunene over hele tidsperioden 2000 til 2023 og henter så ut årsdata som vi lager pendlematriser av.

I dokumentet er lagt inn noe hjelp og flere hint. Jeg har også noen steder vist resultatet av min analyse slik at dere har noen «check-points» for å se at dere er på rett veg.

Kommuner i Sør-Rogaland

Vi definerer Nord-Jæren som kommunene listet nedenfor. Vi tar med både de «gamle» og «nye». Disse kommunene vil bare inneholde data for de årene de har eksistert.

Sandnes (-2019), Stavanger, Nye-Sandnes, Bjerkreim, Hå, Klepp, Time, Gjesdal, Sola, Randaberg, Forsand (-2019), Strand, Hjelmeland, Finnøy (-2019), Rennesøy (-2019), Kvitsøy.

De korresponderende kommune-numrene er:

1102, 1103, 1108, 1114, 1119, 1120, 1121, 1122, 1124, 1127, 1129,
1130, 1133, 1141, 1142, 1144

Det er verdt å merke seg at fra 1.1.2020 har vi 1108 Sandnes som består av 1102 Sandnes og 1129 Forsand. Vi har også 1103 Stavanger som består av 1103 Stavanger, 1141 Finnøy (-2019) og 1142 Rennesøy (-2019).

Vi ser at kommunesammenslåing blir behandlet ulikt for de to. Sandnes får nytt **knr**, mens Stavanger beholder sitt gamle.

I tillegg ble en mindre del av Forsand skilt ut og slått sammen med Strand kommune. Her vil vi se bort fra dette og legge alle arbeidstakere fra Forsand til «Nye Sandnes» (1108). Det er langt vanskeligere å håndtere en splitting av kommuner enn en sammenslåing.

Hente data fra SSB

Vi kunne trenge hele datasettet, dvs. for alle arbeidssted-kommuner, `akom`, og alle bosteds-kommuner, `bkom`, men det vil ikke SSB gi oss fra tabell 03321. Vi henter derfor ut data for alle `akom` (arbeidssted-kommuner) for dem som bor i en av kommunene på Nord-Jæren i et datasett, og et annet datasett der vi henter ut `bkom` (bostedskommune) for dem som arbeider i en kommune på Nord-Jæren. Vi vil da mangle informasjon om hvor mange som både bor og arbeider utenfor Nord-Jæren. Denne informasjonen vil vi nedenfor få tak i ved å hente data fra tabell 11616.

Kall datasettet med antall arbeidstakere som arbeider på Nord-Jæren for `pend_00_23_ssb_arbNJ`, og datasettet med arbeidstakere som bor på Nord-Jæren for `pend_00_23_ssb_boNJ`.

Se denne [sliden](#) (parameteren Region) for tips om hvordan man velger *alle* kommuner. Se videre denne [sliden](#) for hvordan man kan finne hvilke data statistikktabell 03321 kan tilby. OBS! merk skrivemåten *Bokommuen*. Merk også at tid må angis som `character` vektor, dvs. `as.character(2000:2023)`.

```
#Get more info about table
ApiData(
  "http://data.ssb.no/api/v0/en/table/03321",
  returnApiQuery = TRUE
)
```

Hent arbeidstakere som bor på Nord-Jæren, jobber i kommuner over hele landet for perioden 2000-2023. Benytt funksjonen `PxWebApiData::ApiData12()` med argumentene:

- `urlToData`: nummer på tabellen som characters,
- `ArbstedKomm`: alle kommuner i landet ([Hint!](#))
- **`Bokommuen`**: character vektor med kommune-numrene (`knr`) til kommunene på Nord-Jæren.
- `Tid`: character vektor med årene 2000 til 2023

Når en er sikker på at en har fått dataene en ønsker fra SSB kan det være lurt å sette `#| cache: true` i chunk-en. Da blir dataene tatt vare på og en slipper å hente nye data hver gang en kjører koden i dokumentet.

```
pend_00_23_ssb_boNJ <- ApiData12(
  urlToData = "03321",
  ArbstedKomm = list('*'),
  Bokommuen = knr, # Merk skrivemåten
  Tid = as.character(2000:2023)
)
```

Finn på tilsvarende vis antall arbeidstakere som jobber på Nord-Jæren med bosteds-kommune fordelt over hele landet. Kall dette objektet for `pend_00_23_ssb_arbNJ`. Bruk funksjonen `ApiData12()` så får dere både `dataset` og tekst delen i en og samme dataframe.

Plukk så ut data fra `pend_00_23_ssb_boNJ` og plasser i `pend_00_23_boNJ`.

Vi trenger variablene:

- `akom_navn`: Navn på arbeidsstedskommune tatt fra `desc_boNJ`

- `bkom_navn`: Navn på bostedskommune tatt fra `desc_boNJ`
- `akom`: ArbstedKomm men vi legger til en k foran kommunenummer vha. `paste()` for å ha verdier som samsvarer med Rs regler for variabelnavn (kan ikke starte med et tall)
- `bkom`: Bokommuen, men vi legger til en k foran kommunenummer vha. `paste()` for å ha verdier som samsvarer med Rs regler for variabelnavn (kan ikke starte med et tall)
- `aar`: Endrer navn på variabelen Tid
- `pendlere`: Endrer navn på variabelen value
- Til slutt dropper vi variabler vi ikke trenger
- Endrer rekkefølgen på variablene til: aar, akom, akom_navn, bkom, bkom_navn, pendlere
- Til slutt konverterer vi fra dataframe til tibble.

```
pend_00_23_boNJ <- pend_00_23_ssb_boNJ %>%
  # Henter kommune navn fra desc df
  mutate(
    akom_navn = arbeidsstedskommune,
    bkom_navn = bostedskommune ,
    akom = paste("k", ArbstedKomm, sep = ""),
    bkom = paste("k", Bokommuen, sep = "")
  ) %>%
  # Endrer noen variabelnavn
  rename(
    aar = Tid,
    pendlere = value
  ) %>%
  select(aar, akom, akom_navn, bkom, bkom_navn, pendlere) %>%
  as_tibble()
```

Da skal vi ha noe som ser slik ut:

```
print(pend_00_23_boNJ, n = 5)
```

```
# A tibble: 283,392 x 6
  aar   akom akom_navn          bkom bkom_navn      pendlere
<chr> <chr> <chr>          <chr> <chr>          <int>
1 2000  k3001 Halden (2020-2023) k1102 Sandnes (-2019)      0
2 2001  k3001 Halden (2020-2023) k1102 Sandnes (-2019)      0
3 2002  k3001 Halden (2020-2023) k1102 Sandnes (-2019)      0
4 2003  k3001 Halden (2020-2023) k1102 Sandnes (-2019)      0
5 2004  k3001 Halden (2020-2023) k1102 Sandnes (-2019)      0
# i 283,387 more rows
```

Gjør tilsvarende for `pend_00_23_ssb_arbNJ` og plasser resultatet i `pend_00_23_arbNJ`.

Da skal vi ende opp med noe som ser slik ut:

```
# A tibble: 283,392 x 6
```

	aar	akom	akom_navn	bkom	bkom_navn	pendlere
	<chr>	<chr>	<chr>	<chr>	<chr>	<int>
1	2000	k1102	Sandnes (-2019)	k3001	Halden (2020-2023)	0
2	2001	k1102	Sandnes (-2019)	k3001	Halden (2020-2023)	0
3	2002	k1102	Sandnes (-2019)	k3001	Halden (2020-2023)	0
4	2003	k1102	Sandnes (-2019)	k3001	Halden (2020-2023)	0
5	2004	k1102	Sandnes (-2019)	k3001	Halden (2020-2023)	0

i 283,387 more rows

Kommunesammenslåingene

Vi må lage nye variabler `nye_akom` og `nye_bkom` som tar hensyn til kommunesammenslåingen («Nye Stavanger» og «Nye Sandnes» fra 01-01-2020). I tillegg innfører vi en ny «Super kommune» RAL (Resten Av Landet) som inneholder alle kommuner som *ikke* inngår i Nord-Jæren. I den forbindelse trenger vi en ny kommunenummerliste for Nord-Jæren som ikke inneholder kommunene i «Nye Stavanger» og «Nye Sandnes».

```
# knr Nord Jæren utenom kommunene som inngår i Nye Stavanger
# og Nye Sandnes
knr_u_SS <- paste(
  "k",
  c(
    1114, 1119, 1120, 1121, 1122, 1124, 1127, 1130, 1133, 1144
  ),
  sep = ""
)
```

Bosted Nord-Jæren

Vi tar utgangspunkt i `pend_00_21_boNJ` og lager `nye_bkom`, `nye_bkom_navn`, `nye_akom` og `nye_akom_navn` der verdiene avhenger av hvilket nummer/navn kommunene hadde i utgangspunktet. Vi koder resten av landet som k9999 med navn RAL.

```
#pend_00_21_boNJ
pend_00_23_boNJ <- pend_00_23_boNJ %>%
  mutate(
    nye_bkom = case_when(
      bkom %in% c("k1102", "k1108", "k1129") ~ "k1108",
      bkom %in% c("k1103", "k1141", "k1142") ~ "k1103",
      # Øvrige Nord- Jæren beholder sin bkom
      TRUE ~ bkom
    ),
    nye_bkom_navn = case_when(
      bkom %in% c("k1102", "k1108", "k1129") ~ "Sandnes",
      bkom %in% c("k1103", "k1141", "k1142") ~ "Stavanger",
      # Øvrige Nord- Jæren beholder sitt bkom_navn
      TRUE ~ bkom_navn
    ),
```

```

nye_akom = case_when(
  akom %in% c("k1102", "k1108", "k1129") ~ "k1108",
  akom %in% c("k1103", "k1141", "k1142") ~ "k1103",
  # Øvrige Nord- Jæren beholder sin akom
  akom %in% knr_u_SS ~ akom,
  # Resten av landet kodes som knr "9999"
  TRUE ~ "k9999"
),
nye_akom_navn = case_when(
  akom %in% c("k1102", "k1108", "k1129") ~ "Sandnes",
  akom %in% c("k1103", "k1141", "k1142") ~ "Stavanger",
  # Øvrige Nord- Jæren beholder sitt akom_navn
  akom %in% knr_u_SS ~ akom_navn,
  # Resten av landet kodes som knr "RAL"
  TRUE ~ "RAL"
)
)

```

```

# pend_00_23_boNJ_agg
pend_00_23_boNJ_agg <- pend_00_23_boNJ |>
  group_by(nye_bkom, nye_akom, aar,
           nye_bkom_navn, nye_akom_navn) |>
  summarise(pendlere = sum(pendlere), .groups = 'drop') |>
  ungroup() |>
  select(aar, nye_akom, nye_akom_navn, nye_bkom, nye_bkom_navn, pendlere)

```

```

# A tibble: 3,744 x 6
  aar   nye_akom nye_akom_navn nye_bkom nye_bkom_navn pendlere
<chr> <chr>      <chr>          <chr>    <chr>          <int>
1 2000 k1103      Stavanger    k1103    Stavanger    41755
2 2001 k1103      Stavanger    k1103    Stavanger    43136
3 2002 k1103      Stavanger    k1103    Stavanger    43142
4 2003 k1103      Stavanger    k1103    Stavanger    43186
5 2004 k1103      Stavanger    k1103    Stavanger    43472
# i 3,739 more rows

```

```

pend_00_23_boNJ_agg |>
  distinct(nye_bkom) |>
  pull(nye_bkom)

```

```

[1] "k1103" "k1108" "k1114" "k1119" "k1120" "k1121" "k1122" "k1124" "k1127"
[10] "k1130" "k1133" "k1144"

```

```

pend_00_23_boNJ_agg |>
  distinct(nye_akom) |>
  pull(nye_akom)

```

```

[1] "k1103" "k1108" "k1114" "k1119" "k1120" "k1121" "k1122" "k1124" "k1127"

```

```
[10] "k1130" "k1133" "k1144" "k9999"
```

Arbeidssted Nord-Jæren

Hvor bor arbeidstakere som jobber på Nord-Jæren? Gjør sammen aggregering for `pend_00_23_arbNJ` som vi gjorde for `pend_00_23_boNJ` ovenfor.

Da skal vi få noe som ser slik ut:

```
# A tibble: 3,744 x 6
  aar   nye_akom nye_akom_navn nye_bkom nye_bkom_navn pendlere
<chr> <chr>    <chr>          <chr>    <chr>          <int>
1 2000 k1103     Stavanger     k1103     Stavanger     41755
2 2001 k1103     Stavanger     k1103     Stavanger     43136
3 2002 k1103     Stavanger     k1103     Stavanger     43142
4 2003 k1103     Stavanger     k1103     Stavanger     43186
5 2004 k1103     Stavanger     k1103     Stavanger     43472
# i 3,739 more rows
```

Ting ser ut til å være riktig så vi kan endre variabelnavn fra `nye_akom` til `akom`, og tilsvarende for de andre, i `pend_00_23_arbNJ_agg` og `pend_00_23_boNJ_agg`.

Endrer variabelnavn `pend_00_23_arbNJ_agg`. Hint! `rename()`.

```
pend_00_23_arbNJ_agg <- pend_00_23_arbNJ_agg |>
  rename(
    akom = nye_akom,
    akom_navn = nye_akom_navn,
    bkom = nye_bkom,
    bkom_navn = nye_bkom_navn
  )
```

Gjør tilsvarende for `pend_00_23_boNJ_agg`.

Da skal vi ha følgende navn:

```
names(pend_00_23_arbNJ_agg)
```

```
[1] "aar"      "akom"      "akom_navn" "bkom"      "bkom_navn" "pendlere"
```

og

```
names(pend_00_23_boNJ_agg)
```

```
[1] "aar"      "akom"      "akom_navn" "bkom"      "bkom_navn" "pendlere"
```

Vi slår så sammen `pend_00_23_arbNJ_agg` og `pend_00_23_boNJ_agg` til et felles datasett vha. en `rbind()`. Det vi trenger fra `pend_00_23_boNJ_agg` er bare dem som bor NJ og arbeider i RAL.

```
boNJ_arb_RAL <- pend_00_23_boNJ_agg |>
  filter(akom == "k9999")
```

```
pend_00_23 <- bind_rows(
  pend_00_23_arbNJ_agg,
  boNJ_arb_RAL
)
```

Datasettene `pend_00_23`, `pend_00_23_arbNJ_agg` og `pend_00_23_boNJ_agg` vil være de vi skal jobbe med videre. Vi kan derfor gjøre litt rydding og kvitte oss med `arb_RAL`, `pend_00_23_arbNJ`, `pend_00_23_boNJ`, `pend_00_23_ssb_arbNJ` og `pend_00_23_ssb_boNJ`.

```
rm(boNJ_arb_RAL, pend_00_23_arbNJ, pend_00_23_boNJ,
  pend_00_23_ssb_arbNJ, pend_00_23_ssb_boNJ
)
```

Totalt antall arbeidstakere i hele landet per år

For tabell 03321 får vi *ikke* lastet ned for *alle* bostedkommuner og *alle* jobbkommuner samtidig. Vi ønsker imidlertid å finne antallet arbeidstakere som *både* bor og jobber i resten av landet (RAL). Dette finner vi vha. (HL er Hele Landet):

Tot. arb. takere HL

- Bor NJ jobber NJ

- Bor NJ jobber Ral

- Bor RAL jobber NJ

= Bor og jobber i RAL

Dette kan forenkles til:

Tot. arb. takere HL

- Bor NJ jobber HL

- Bor RAL jobber NJ

= Bor og jobber i RAL

Vi henter data fra tabell 11616 fordi den gir oss data på fylkes-basis. Vi droppe «Svalbard og Jan Mayen» og «Kontinentalsokkelen» da det ikke finnes data for disse. Vi henter data for årene 2000 til 2023, begge kjønn, alder 15-74 år og ut fra hvor arbeidstakerne er bosatt. Vi er bare interessert i totalt antall arbeidstakere per år.

```
fnr <- c(
  "30", "01", "02", "06", "03", "34", "04", "05", "38",
  "07", "08", "42", "09", "10", "11", "46", "12", "14",
  "15", "50", "16", "17", "18", "54", "19", "20"
)
```


Hent vha. `ApiData12()` opplysninger fra tabell 11616 om antall arbeidstakere bosatt i hvert fylke årene 2000 til 2023.

Vi lager en tibble med to variabler, «Tid» (dvs. årstall) og «arbtak_HL».

Toppen av `tot_arb_HL` skal da se slik ut:

```
# A tibble: 24 x 2
  aar   arbtak_HL
<chr>   <int>
1 2000     2262000
2 2001     2275000
3 2002     2267000
4 2003     2260000
5 2004     2274000
# i 19 more rows
```

Bor NJ jobber HL

Definerer Nord-Jæren (NJ)

```
NJ <- c(knr_u_SS, "k1103", "k1108")
```

Legg antall som bor på Nord-Jæren og jobber utenfor region i `bNJjHL`.

```
bNJjHL <- pend_00_23 %>%
  filter(bkom %in% NJ) %>%
  group_by(aar) %>%
  summarise(bNJjHL = sum(pendlere))
```

Bor RAL jobber på NJ

Gjør tilsvarende for dem som bor på Nord-Jæren, men jobber utenfor regionen.

Sett sammen `tot_arb_HL`, `bNJjHL` og `bRALjNJ` til `tot_arb_HL` slik at vi kan regne ut antall som bor i RAL og jobber i RAL (RAL; resten av landet).

```
tot_arb_HL <- left_join(tot_arb_HL, bNJjHL, by = join_by(aar))
tot_arb_HL <- left_join(tot_arb_HL, bRALjNJ, by = join_by(aar))
```

Legger `p_bRALjRAL` til `pend_00_21`.

```
pend_00_23 <- bind_rows(pend_00_23, p_bRALjRAL)
```

Sjekker at ting ser greit ut ved å lage en foreløpig pendle-matrise for 2010.

```
# pendlematrise 2010, bo Nord-Jæren
pend_00_23 |>
  ungroup() |>
  filter(aar == "2010") |>
  select(bkom, akom, pendlere) |>
  pivot_wider(
    names_from = akom,
    values_from = pendlere
  ) |>
  as_flextable(max_row = 20, show_coltype = FALSE) |>
  autofit() |>
  line_spacing(space = 0.3, part = "all")
```

Table 1: Pendlematrise for 2010 for «Sør-fylket» i Rogaland. Kommunestrukturen fra 2018.

bkom	k1103	k1108	k1114	k1119	k1120	k1121	k1122	k1124
k1103	50,038	7,707	12	126	411	283	165	5,846
k1108	10,577	17,650	23	303	962	569	492	2,920
k1114	96	117	853	35	21	27	42	36
k1119	677	696	17	5,446	566	1,021	58	323
k1120	1,499	1,711	7	374	3,464	1,144	82	759
k1121	1,357	1,316	11	587	1,095	3,724	134	458
k1122	990	1,478	31	94	211	176	2,252	347
k1124	4,454	1,860	3	37	200	92	31	4,866
k1127	2,366	431	1	7	24	8	9	489
k1130	963	344	0	8	12	8	3	117
k1133	106	32	0	2	0	1	0	14
k1144	72	15	0	1	1	0	0	7
k9999	8,036	2,114	188	352	158	232	94	2,114
n: 13								

Pendlematrise for 2010 i Table 1 ser grei ut. Mangler nå bare totalt antall slik at vi kan regne ut andeler.

```
totalt_arb <- pend_00_23 |>
  group_by(aar, akom, akom_navn) |>
  summarise(pendlere = sum(pendlere), .groups = 'drop') |>
  mutate(
    bkom = "k0000",
    bkom_navn = "TotaltArb"
  )
```

```
totalt_bo <- pend_00_23 |>
  group_by(aar, bkom, bkom_navn) |>
  summarise(pendlere = sum(pendlere), .groups = 'drop') |>
  mutate(
```

```

    akom = "k0000",
    akom_navn = "TotaltBo"
  )

```

Bruk så `bind_rows()` for å legge til `totalt_arb` og `totalt_bo` til `pend_00_23` og legg resultatet i `pendle_data_00_23`. Her er det viktig at de tre har identiske variabelnavn og i samme rekkefølge.

Vi skal da ha:

```
dim(pendle_data_00_23)
```

```
[1] 4704    6
```

```
names(pendle_data_00_23)
```

```
[1] "aar"      "akom"      "akom_navn" "bkom"      "bkom_navn" "pendlere"
```

```
print(pendle_data_00_23, n = 5)
```

```

# A tibble: 4,704 x 6
  aar   akom akom_navn bkom  bkom_navn pendlere
<chr> <chr> <chr>      <chr> <chr>      <int>
1 2000  k1103 Stavanger k1103 Stavanger  41755
2 2001  k1103 Stavanger k1103 Stavanger  43136
3 2002  k1103 Stavanger k1103 Stavanger  43142
4 2003  k1103 Stavanger k1103 Stavanger  43186
5 2004  k1103 Stavanger k1103 Stavanger  43472
# i 4,699 more rows

```

Andel pendlere

Vi kan så regne ut andel av arbeidstakerne som bor i en kommune som jobber i de ulike kommunene. Dette kan gjøres på ulike måter, men det enkleste er kanskje å `pivot_wider()` og `across()` i en `mutate()`. Vi kan da benytte `mutate()` for å lage nye variabler med andelene for hver kommune. Legg resultatet i en tibble kalt `andel_pendle_data_00_23`. Det kan også være lurt å forandre variabelen `aar` til type `date` vha. `dmy()`. Dette vil senere gi oss finere plot av dataene.

Da har vi:

```
dim(andel_pendle_data_00_23)
```

```
[1] 336   17
```

```
names(andel_pendle_data_00_23)
```

```

[1] "aar"      "bkom"      "bkom_navn"      "k1103_Stavanger"
[5] "k1108_Sandnes" "k1114_Bjerkreim" "k1119_Hå"      "k1120_Klepp"
[9] "k1121_Time"    "k1122_Gjesdal"   "k1124_Sola"    "k1127_Randaberg"
[13] "k1130_Strand"  "k1133_Hjelmeland" "k1144_Kvitsøy"  "k9999_RAL"
[17] "k0000_TotaltBo"

```

```
andel_pendle_data_00_23 |>
  head(n = 5) |>
  as_flextable()
```

aar	bkom	bkom_navn	k1103_Stavanger	k1108_Sandnes	k1114_Bjerkreim
character	character	character	numeric	numeric	numeric
2000	k1103	Stavanger	72.3	8.0	0.0
2001	k1103	Stavanger	73.4	8.1	0.0
2002	k1103	Stavanger	73.1	8.0	0.0
2003	k1103	Stavanger	72.9	9.1	0.0
2004	k1103	Stavanger	72.1	9.7	0.0
n: 5					

Så går vi long før plot.

Vi skal nå ha følgende «kommuner»:

akom	akom_navn
k1103	Stavanger
k1108	Sandnes
k1114	Bjerkreim
k1119	Hå
k1120	Klepp
k1121	Time
k1122	Gjesdal
k1124	Sola
k1127	Randaberg
k1130	Strand
k1133	Hjelmeland
k1144	Kvitsøy
k9999	RAL
n: 13	

Og for datasettet vårt:

```
dim(andel_pendle_data_00_23_long)

[1] 4704      6

names(andel_pendle_data_00_23_long)

[1] "aar"      "akom"      "akom_navn" "bkom"      "bkom_navn" "andel"

print(andel_pendle_data_00_23_long, n = 5, width = 76)

# A tibble: 4,704 x 6
  aar   akom akom_navn bkom  bkom_navn  andel
<chr> <chr> <chr>      <chr> <chr>      <dbl>
1 2000 k1103 Stavanger k1103 Stavanger 72.3
2 2000 k1108 Sandnes  k1103 Stavanger 7.99
3 2000 k1114 Bjerkreim k1103 Stavanger 0.0069
4 2000 k1119 Hå       k1103 Stavanger 0.154
5 2000 k1120 Klepp    k1103 Stavanger 0.346
# i 4,699 more rows
```

Når vi nå endelig har fått dataene på rett form i `andel_pendle_data_00_23_long` er det relativt enkelt å lage fine line-plot av pendle-andeler og også å genere såkalte pendle-matriser for de årene vi måtte ønske.

En siste ting vi bør gjøre er å konvertere `aar` til et `date` objekt. `ggplot()` vil gi oss en langt finere x-akse hvis vi har `aar` som `date` enn som nå `integer`.

Bor og jobber i samme kommune på Nord-Jæren

Da har vi datasettet vårt, `andel_pendle_data_00_23_long`, og vil bruke det i det resterende.

Lag et linjeplot (`geom_line()`) som viser utviklingen fra 2000 til 2023 i andelen arbeidstakere som bor og jobber i samme kommune. Hint! Sett «aesthetics» `group` og `colour` lik `bkom` (eller `akom`).

```
andel_pendle_data_00_23_long |>
  filter(akom == bkom) |>
  filter(!(akom == "k9999" & bkom == "k9999")) |>
  rename(
    Bosted = bkom_navn
  ) |>
  ggplot(
    mapping = aes(
      x = aar,
      y = andel,
      group = bkom,
      colour = bkom
    )
  ) +
  geom_line(lwd = 0.75) +
  theme(legend.position = 'bottom') +
```

```
ggtitle("Intern pendling")
```

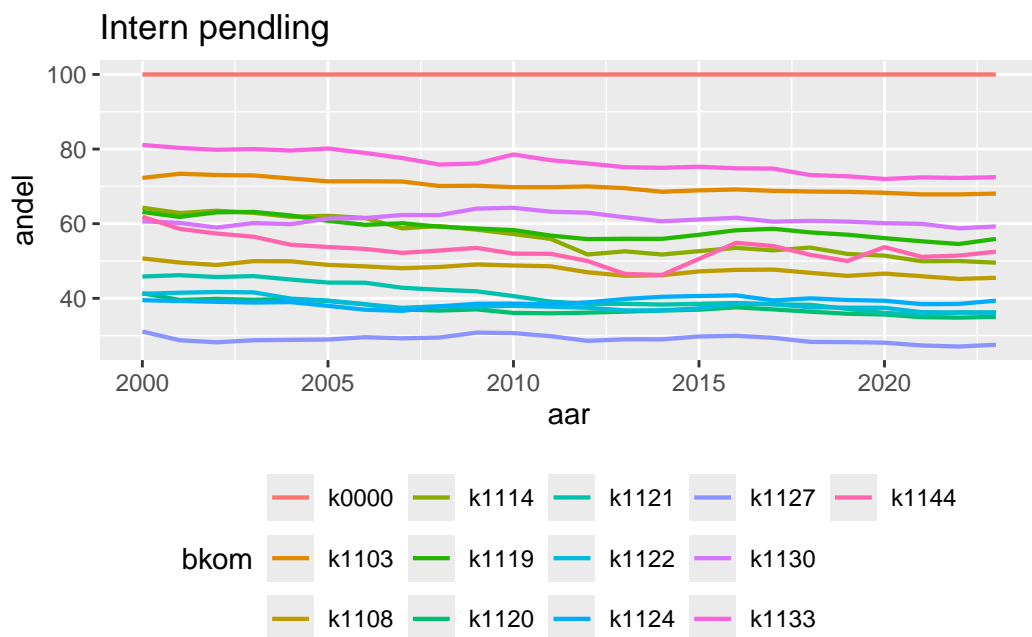
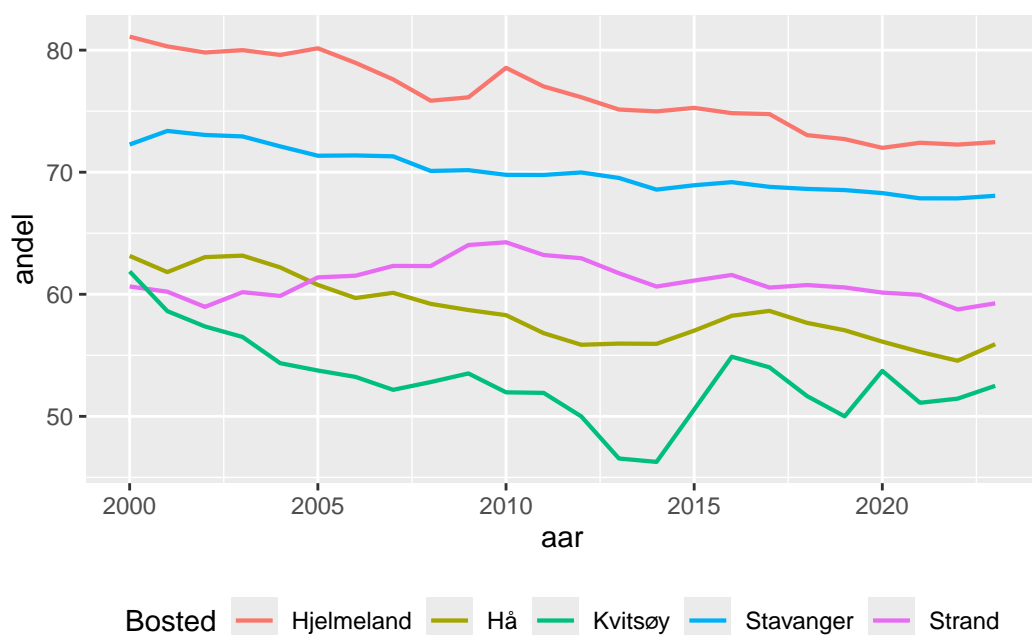


Table 4: Tabell som viser andelen av arbeidstakere som jobber i samme kommune som hvor de bor i 2023.

aar	akom	akom_navn	bkom	Bosted	andel
2023-01-01	k0000	TotaltBo	k0000	TotaltArb	100.0
2023-01-01	k1133	Hjelmeland	k1133	Hjelmeland	72.5
2023-01-01	k1103	Stavanger	k1103	Stavanger	68.1
2023-01-01	k1130	Strand	k1130	Strand	59.3
2023-01-01	k1119	Hå	k1119	Hå	55.9
2023-01-01	k1144	Kvitsøy	k1144	Kvitsøy	52.5
2023-01-01	k1114	Bjerkreim	k1114	Bjerkreim	49.6
2023-01-01	k1108	Sandnes	k1108	Sandnes	45.5
2023-01-01	k1124	Sola	k1124	Sola	39.4
2023-01-01	k1121	Time	k1121	Time	36.2
2023-01-01	k1122	Gjesdal	k1122	Gjesdal	36.1
2023-01-01	k1120	Klepp	k1120	Klepp	35.0

aar akom	akom_navn	bkom	Bosted	andel
2023-01-01 k1127	Randaberg	k1127	Randaberg	27.6
n: 13				

Interpendling et utvalg kommuner



Definerer funksjoner for plot og tabell

Viser koden for en funksjon som lager tabeller i det følgende. Det kan være hensiktsmessig å lage en tilsvarende funksjon for plot-ene som skal lages.

```
tab_pendlere <- function(knr)
andel_pendle_data_00_23_long |>
  filter(bkom == knr) |>
  filter(!akom %in% c(knr, "k0000")) |>
  filter(year(aar) == "2023") |>
  arrange(desc(andel)) |>
  mutate(andel = round(andel, digits = 2)) |>
  select(`Place of work` = akom_navn, `Prop. in %` = andel) |>
  as_flextable(show_coltype = FALSE) |>
  delete_part("footer") |>
  theme_booktabs()
```

Hvor jobber folk bosatt i Stavanger kommune?

Vi så ovenfor at 68.1% både bor og jobber i Stavanger kommune i 2023, men hvor jobber de som pendler ut av kommunen? Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Stavanger. Hvordan har andelen arbeidstakere som både bor og jobber i Stavanger utviklet seg over tid?

```
plot_pendlere("k1103")
```

```
```{r}
#| label: tbl-1103pend
#| tbl-cap: Andelene som pendler ut av Stavanger i 2023.
#| output: true
tab_pendlere("k1103")
```
```

Table 5: Andelene som pendler ut av Stavanger i 2023.

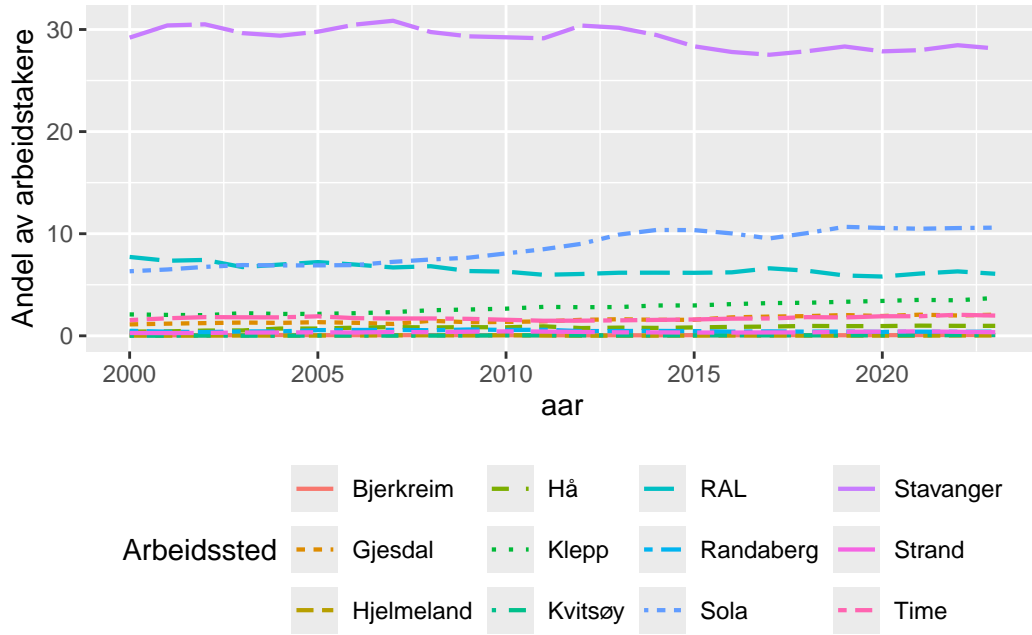
| Place of work | Prop. in % |
|---------------|------------|
| Sandnes | 10.6 |
| Sola | 10.5 |
| RAL | 6.8 |
| Randaberg | 1.8 |
| Klepp | 0.8 |
| Time | 0.5 |
| Hå | 0.3 |
| Strand | 0.3 |
| Gjesdal | 0.3 |
| Kvitsøy | 0.1 |

Hvor jobber folk bosatt i Sandnes kommune?

Vi så ovenfor at bare 45,5% av arbeidstakere bosatt i Sandnes jobbet i samme kommune i 2023. Hvor jobber de som jobber utenfor kommunen? Hvordan har andelen som både bor og jobber i Sandnes kommune utviklet seg over tid? Diskuter hva som kan være årsaken til at vi har denne forskjellen mellom Stavanger og Sandnes når det gjelder intern (innen samme kommune) pendling.

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Sandnes. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

```
plot_pendlere("k1108")
```

```
tab_pendlere("k1108")
```

Table 6: Andelene som pendler ut av Sandnes i 2023.

| Place of work | Prop. in % |
|---------------|------------|
| Stavanger | 28.1 |
| Sola | 10.6 |
| RAL | 6.1 |
| Klepp | 3.7 |
| Gjesdal | 2.1 |
| Time | 2.0 |
| Hå | 1.0 |
| Randaberg | 0.4 |
| Strand | 0.4 |
| Kvitsøy | 0.1 |

Table 7: Andelene som pendler ut av Sola kommune i 2023.

Table 8: Andelene som pendler ut av Randaberg kommune i 2023.

Hvor jobber folk bosatt i Sola kommune?

Vi så ovenfor at 39.4% både bor og jobber i Sola kommune i 2023. Hvor jobber de som pendler ut av kommunen?

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Sola. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

Hvor jobber folk bosatt i Randaberg kommune?

Bare 27.6% av dem som bodde i Randaberg kommune i 2023 jobber i denne kommunen. Hvor jobber de som pendler ut av kommunen?

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Randaberg. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

Hvor jobber folk bosatt i Klepp kommune?

Klepp kommune ligger plassert mellom Sandnes og Bryne. Vi så ovenfor at 35,0% både bor og jobber i denne kommunen. Hvor jobber de som pendler ut av kommunen?

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Klepp. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

Hvor jobber folk bosatt i Time kommune?

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Time. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

Hvor jobber folk bosatt i Gjesdal kommune?

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Gjesdal. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

Hvor jobber folk bosatt i Hå kommune?

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Hå. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

Table 9: Andelene som pendler ut av Klepp kommune i 2023.

Table 10: Pendling ut av Time kommune i 2023.

Table 11: Pendling ut av Gjesdal kommune i 2023.

Hvor jobber folk bosatt i Bjerkreim kommune?

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Bjerkreim. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

Hvor jobber folk bosatt i Strand kommune?

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i Strand. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

Ryfast er et vegprosjekt som forbinder Strand kommune med Stavanger vha. ferjefri forbindelse. Prosjektet ble vedtatt av Stortinget i 2012, byggestart var våren 2013 og forbindelsen åpnet 30. des. 2019. Total kostnad på prosjektet er 6-7 milliarder kroner.

Stavanger Aftenblad skrev 12. oktober 2015:

Fra Strand, Forsand og Hjelmeland pendler over 1800 personer daglig til Forus og omegn. Når reisetiden reduseres til 20 minutter kan Ryfast bidra til at Strand kommune utvikler seg til en forstadskommune til Stavanger ...

Diskuter utfra plot-et du genererte ovenfor om denne spådommen har slått til så langt.

Hvor stor pendling er det fra resten av landet til Nord-Jæren?

Lag et linje-plot lik ovenfor, men som nå viser andelen som jobber i de ulike kommunene for folk som bor i resten av landet. Lag også en tabell som ovenfor som viser de fem med størst andel i 2023.

Pendlematriser

Vi kan nå enkelt generere pendle-matriser for ulike år. Vi nøyer oss med årene 2000, 2005, 2010, 2015 og 2021.

2000

Generer en pendle-matrise for 2000 lik den vist i Figure 1 nedenfor.

2010

Generer en pendle-matrise for 2010.

2020

Generer en pendle-matrise for 2020.

2023

Generer en pendle-matrise for 2023 se Figure 2.

Table 12: Pendling ut av Hå kommune i 2023.

Table 13: Pendling ut av Bjerkreim kommune i 2023.

Table 14: Pendling ut av Strand kommune i 2023.

Table 15: Pendling inn til Nord-Jæren fra resten av landet i 2023.

Table 16Pendlematrise for Nord-Jæren for år 2000.

| bkom | k1103 | k1108 | k1114 | k1119 | k1120 | k1121 | k1122 | k1124 | k1127 | k1130 | k1133 | k1144 | k9999 | k0000 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---------|---------|
| k1103 | 41755 | 4618 | 4 | 89 | 200 | 193 | 83 | 3864 | 1025 | 98 | 19 | 25 | 5803 | 57776 |
| k1108 | 8183 | 14206 | 19 | 116 | 588 | 431 | 314 | 1772 | 138 | 78 | 7 | 4 | 2165 | 28021 |
| k1114 | 50 | 79 | 819 | 15 | 12 | 14 | 51 | 14 | 4 | 3 | 0 | 0 | 213 | 1274 |
| k1119 | 610 | 477 | 2 | 4686 | 334 | 652 | 33 | 170 | 6 | 2 | 0 | 0 | 449 | 7421 |
| k1120 | 1178 | 1178 | 11 | 193 | 3060 | 865 | 42 | 427 | 21 | 6 | 0 | 0 | 416 | 7397 |
| k1121 | 1030 | 937 | 3 | 335 | 822 | 3298 | 106 | 238 | 12 | 5 | 1 | 0 | 406 | 7193 |
| k1122 | 911 | 1017 | 24 | 30 | 127 | 104 | 1921 | 208 | 10 | 8 | 1 | 0 | 300 | 4661 |
| k1124 | 3651 | 1052 | 0 | 24 | 112 | 60 | 18 | 3857 | 117 | 11 | 1 | 9 | 853 | 9765 |
| k1127 | 2190 | 269 | 0 | 2 | 16 | 14 | 3 | 301 | 1438 | 3 | 0 | 3 | 385 | 4624 |
| k1130 | 893 | 215 | 1 | 19 | 12 | 9 | 5 | 120 | 14 | 2955 | 116 | 0 | 514 | 4873 |
| k1133 | 81 | 19 | 0 | 5 | 2 | 4 | 1 | 12 | 2 | 46 | 1159 | 0 | 98 | 1429 |
| k1144 | 51 | 9 | 0 | 1 | 0 | 2 | 0 | 5 | 11 | 1 | 0 | 159 | 18 | 257 |
| k9999 | 5542 | 2190 | 125 | 190 | 93 | 124 | 42 | 1585 | 217 | 41 | 43 | 46 | 2117071 | 2127309 |
| k0000 | 66125 | 26266 | 1008 | 5705 | 5378 | 5770 | 2619 | 12573 | 3015 | 3257 | 1347 | 246 | 2128691 | 2262000 |
| n: 14 | | | | | | | | | | | | | | |

Figure 1

| bkom | k1103 | k1108 | k1114 | k1119 | k1120 | k1121 | k1122 | k1124 | k1127 | k1130 | k1133 | k1144 | k9999 | k0000 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---------|---------|
| k1103 | 53630 | 8380 | 26 | 210 | 646 | 352 | 208 | 8288 | 1379 | 209 | 50 | 68 | 5343 | 78789 |
| k1108 | 12316 | 19912 | 33 | 428 | 1625 | 868 | 906 | 4638 | 177 | 156 | 15 | 35 | 2660 | 43769 |
| k1114 | 108 | 122 | 792 | 72 | 45 | 28 | 84 | 48 | 3 | 1 | 0 | 0 | 295 | 1598 |
| k1119 | 732 | 758 | 30 | 5707 | 770 | 1168 | 87 | 346 | 16 | 9 | 2 | 1 | 581 | 10207 |
| k1120 | 1691 | 1933 | 15 | 558 | 3968 | 1253 | 128 | 1094 | 38 | 15 | 2 | 12 | 618 | 11325 |
| k1121 | 1570 | 1441 | 20 | 843 | 1487 | 3882 | 172 | 696 | 11 | 10 | 0 | 6 | 580 | 10718 |
| k1122 | 1058 | 1433 | 92 | 146 | 355 | 223 | 2366 | 506 | 18 | 11 | 1 | 5 | 337 | 6551 |
| k1124 | 5384 | 2221 | 4 | 69 | 308 | 129 | 65 | 6009 | 149 | 16 | 5 | 17 | 888 | 15264 |
| k1127 | 2619 | 532 | 1 | 18 | 41 | 29 | 11 | 683 | 1668 | 17 | 0 | 20 | 415 | 6054 |
| k1130 | 1222 | 536 | 0 | 14 | 39 | 11 | 8 | 300 | 27 | 4023 | 201 | 0 | 408 | 6789 |
| k1133 | 83 | 29 | 0 | 1 | 3 | 1 | 1 | 40 | 3 | 113 | 992 | 0 | 103 | 1369 |
| k1144 | 69 | 19 | 0 | 1 | 4 | 1 | 0 | 14 | 6 | 0 | 1 | 147 | 18 | 280 |
| k9999 | 9031 | 4477 | 170 | 533 | 306 | 308 | 141 | 4792 | 148 | 101 | 59 | 38 | 2632490 | 2652594 |
| k0000 | 89513 | 41793 | 1183 | 8600 | 9597 | 8253 | 4177 | 27454 | 3643 | 4681 | 1328 | 349 | 2644736 | 2845307 |
| n: 14 | | | | | | | | | | | | | | |

Figure 2: Pendlematrise for år 2023 for Nord-Jæren.