Modeller

```
suppressPackageStartupMessages({
  library(tidyverse)
  library(lubridate)
  library(modelr)
  library(broom)
  library(lmtest)
  library(sandwich)
  library(viridis)
})
```

```
pm2 <- read_csv("data/pm2.csv", show_col_types = FALSE)
```

```
pm2 <- pm2 %>%
  mutate(
    fnr = str_sub(knr, 1,2),
    aar_f = str_sub(aar)
  )
```

```
head(pm2)
```

```
## # A tibble: 6 x 18
##   knr     aar knavn    pm2 Menn_ya_p Kvinner_ya_p Total_ya_p inc_k1 inc_k5
##   <chr> <dbl> <chr>  <dbl>     <dbl>        <dbl>      <dbl>  <dbl>  <dbl>
## 1 0101   2008 Halden 13427      59.7         56.8       58.3   24.5   13.6
## 2 0101   2009 Halden 13095      59.8         57.0       58.4   24.4   14.1
## 3 0101   2010 Halden 13832      59.6         57.1       58.3   23.9   13.7
## 4 0101   2011 Halden 14915      59.8         57.2       58.5   24     14
## 5 0101   2012 Halden 15473      59.5         57.0       58.2   23.9   14
## 6 0101   2013 Halden 15461      59.0         56.7       57.9   24.1   13.4
## # ... with 9 more variables: uni_k_mf <dbl>, uni_k_m <dbl>, uni_k_f <dbl>,
## #   uni_l_mf <dbl>, uni_l_m <dbl>, uni_l_f <dbl>, Trade_p <dbl>, fnr <chr>,
## #   aar_f <chr>
```

```
pm2 %>%
  mutate(
    fnr = parse_factor(fnr, levels = fnr),
    aar_f = parse_factor(aar_f, levels = aar_f)
  )
```

```
## # A tibble: 2,140 x 18
##    knr     aar knavn    pm2 Menn_ya_p Kvinner_ya_p Total_ya_p inc_k1 inc_k5
##    <chr> <dbl> <chr>  <dbl>     <dbl>        <dbl>      <dbl>  <dbl>  <dbl>
##  1 0101   2008 Halden 13427      59.7         56.8       58.3   24.5   13.6
##  2 0101   2009 Halden 13095      59.8         57.0       58.4   24.4   14.1
##  3 0101   2010 Halden 13832      59.6         57.1       58.3   23.9   13.7
##  4 0101   2011 Halden 14915      59.8         57.2       58.5   24     14
##  5 0101   2012 Halden 15473      59.5         57.0       58.2   23.9   14
##  6 0101   2013 Halden 15461      59.0         56.7       57.9   24.1   13.4
```

```
##  7 0101   2014 Halden 17164      58.8         56.7       57.7   23.9   13.5
##  8 0101   2015 Halden 17427      58.7         56.8       57.8   24     13.7
##  9 0101   2016 Halden 18941      58.7         56.6       57.7   24     13.8
## 10 0101   2017 Halden 20143      58.9         56.9       57.9   23.7   14
## # ... with 2,130 more rows, and 9 more variables: uni_k_mf <dbl>,
## #   uni_k_m <dbl>, uni_k_f <dbl>, uni_l_mf <dbl>, uni_l_m <dbl>, uni_l_f <dbl>,
## #   Trade_p <dbl>, fnr <fct>, aar_f <fct>
```

```
pm2 <- pm2 %>%
  mutate(
    Trade_pc_100K = Trade_p/100000
  )
```

```
head(pm2, n = 4)
```

```
## # A tibble: 4 x 19
##   knr     aar knavn    pm2 Menn_ya_p Kvinner_ya_p Total_ya_p inc_k1 inc_k5
##   <chr> <dbl> <chr>  <dbl>     <dbl>        <dbl>      <dbl>  <dbl>  <dbl>
## 1 0101   2008 Halden 13427      59.7         56.8       58.3   24.5   13.6
## 2 0101   2009 Halden 13095      59.8         57.0       58.4   24.4   14.1
## 3 0101   2010 Halden 13832      59.6         57.1       58.3   23.9   13.7
## 4 0101   2011 Halden 14915      59.8         57.2       58.5   24     14
## # ... with 10 more variables: uni_k_mf <dbl>, uni_k_m <dbl>, uni_k_f <dbl>,
## #   uni_l_mf <dbl>, uni_l_m <dbl>, uni_l_f <dbl>, Trade_p <dbl>, fnr <chr>,
## #   aar_f <chr>, Trade_pc_100K <dbl>
```

```
tibble("knr", "fnr", "aar_f", "Trade_pc_100k")
```

```
## # A tibble: 1 x 4
##   `"knr"` `"fnr"` `"aar_f"` `"Trade_pc_100k"`
##   <chr>   <chr>   <chr>     <chr>
## 1 knr     fnr     aar_f     Trade_pc_100k
```

#Modell

```
mod1 <- 'pm2 ~ aar_f + Total_ya_p + inc_k1 + inc_k5 + uni_k_mf + uni_l_mf + Trade_pc_100K'
```

```
lm1 <- lm(mod1, data = pm2, subset = complete.cases(pm2))
```

Vi legger inn residualene fra den lineære modellen

```
pm2 %>%
add_residuals(lm1)
```

```
## # A tibble: 2,140 x 20
##    knr     aar knavn    pm2 Menn_ya_p Kvinner_ya_p Total_ya_p inc_k1 inc_k5
##    <chr> <dbl> <chr>  <dbl>     <dbl>        <dbl>      <dbl>  <dbl>  <dbl>
##  1 0101   2008 Halden 13427      59.7         56.8       58.3   24.5   13.6
##  2 0101   2009 Halden 13095      59.8         57.0       58.4   24.4   14.1
##  3 0101   2010 Halden 13832      59.6         57.1       58.3   23.9   13.7
##  4 0101   2011 Halden 14915      59.8         57.2       58.5   24     14
##  5 0101   2012 Halden 15473      59.5         57.0       58.2   23.9   14
##  6 0101   2013 Halden 15461      59.0         56.7       57.9   24.1   13.4
##  7 0101   2014 Halden 17164      58.8         56.7       57.7   23.9   13.5
##  8 0101   2015 Halden 17427      58.7         56.8       57.8   24     13.7
##  9 0101   2016 Halden 18941      58.7         56.6       57.7   24     13.8
## 10 0101   2017 Halden 20143      58.9         56.9       57.9   23.7   14
## # ... with 2,130 more rows, and 11 more variables: uni_k_mf <dbl>,
```

```
## #   uni_k_m <dbl>, uni_k_f <dbl>, uni_l_mf <dbl>, uni_l_m <dbl>, uni_l_f <dbl>,
## #   Trade_p <dbl>, fnr <chr>, aar_f <chr>, Trade_pc_100K <dbl>, resid <dbl>
```

```
head(pm2, n = 4)
```

```
## # A tibble: 4 x 19
##   knr     aar knavn    pm2 Menn_ya_p Kvinner_ya_p Total_ya_p inc_k1 inc_k5
##   <chr> <dbl> <chr>  <dbl>     <dbl>        <dbl>      <dbl>  <dbl>  <dbl>
## 1 0101   2008 Halden 13427      59.7         56.8       58.3   24.5   13.6
## 2 0101   2009 Halden 13095      59.8         57.0       58.4   24.4   14.1
## 3 0101   2010 Halden 13832      59.6         57.1       58.3   23.9   13.7
## 4 0101   2011 Halden 14915      59.8         57.2       58.5   24     14
## # ... with 10 more variables: uni_k_mf <dbl>, uni_k_m <dbl>, uni_k_f <dbl>,
## #   uni_l_mf <dbl>, uni_l_m <dbl>, uni_l_f <dbl>, Trade_p <dbl>, fnr <chr>,
## #   aar_f <chr>, Trade_pc_100K <dbl>
```

```
summary(lm1)
```

```
##
## Call:
## lm(formula = mod1, data = pm2, subset = complete.cases(pm2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8516.6 -1472.1   -29.9  1467.3 15736.3
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -20400.74    2663.02  -7.661 2.79e-14 ***
## aar_f2009        104.15     244.77   0.426 0.670512
## aar_f2010        908.13     245.16   3.704 0.000217 ***
## aar_f2011       1663.93     245.86   6.768 1.68e-11 ***
## aar_f2012       2240.48     247.10   9.067  < 2e-16 ***
## aar_f2013       2869.30     248.31  11.555  < 2e-16 ***
## aar_f2014       2863.22     250.54  11.428  < 2e-16 ***
## aar_f2015       3525.22     253.08  13.929  < 2e-16 ***
## aar_f2016       4274.99     255.81  16.711  < 2e-16 ***
## aar_f2017       5146.33     258.50  19.909  < 2e-16 ***
## Total_ya_p       582.44      38.94  14.957  < 2e-16 ***
## inc_k1          -376.99      30.29 -12.445  < 2e-16 ***
## inc_k5           194.35      22.87   8.498  < 2e-16 ***
## uni_k_mf         -82.02      29.42  -2.788 0.005357 **
## uni_l_mf        1206.86      42.22  28.585  < 2e-16 ***
## Trade_pc_100K    871.99     218.42   3.992 6.77e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2531 on 2124 degrees of freedom
## Multiple R-squared:  0.8346, Adjusted R-squared:  0.8334
## F-statistic: 714.3 on 15 and 2124 DF,  p-value: < 2.2e-16
```

Ut i fra verdien på års koeffisientene kan vi lese at det er en økning på o,1% fra 2010 til 2017. Vi ser en økning fra år til år.

Vi antar at de øvrige koeffisientene er som forventet, da kvadrattmeter prisen har til vane å øke fra år til år.

#Heteroskedastisitet

```
bptest(lm1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  lm1
## BP = 352.89, df = 15, p-value < 2.2e-16
```

Ja vi har problemer med heteroskedasiteten her pga at verdien er for høy.

#Rapportere robuste standard feil og tilhørende robuste t-verdier

```
coeftest(lm1)
```

```
##
## t test of coefficients:
##
##                 Estimate Std. Error  t value  Pr(>|t|)
## (Intercept)   -20400.742   2663.022  -7.6607 2.790e-14 ***
## aar_f2009        104.150    244.767   0.4255 0.6705118
## aar_f2010        908.129    245.156   3.7043 0.0002174 ***
## aar_f2011       1663.926    245.857   6.7679 1.685e-11 ***
## aar_f2012       2240.475    247.095   9.0672 < 2.2e-16 ***
## aar_f2013       2869.297    248.315  11.5551 < 2.2e-16 ***
## aar_f2014       2863.224    250.537  11.4283 < 2.2e-16 ***
## aar_f2015       3525.223    253.083  13.9291 < 2.2e-16 ***
## aar_f2016       4274.990    255.812  16.7114 < 2.2e-16 ***
## aar_f2017       5146.326    258.498  19.9086 < 2.2e-16 ***
## Total_ya_p       582.436     38.941  14.9568 < 2.2e-16 ***
## inc_k1          -376.989     30.291 -12.4455 < 2.2e-16 ***
## inc_k5           194.354     22.871   8.4979 < 2.2e-16 ***
## uni_k_mf         -82.023     29.424  -2.7876 0.0053574 **
## uni_l_mf        1206.857     42.219  28.5853 < 2.2e-16 ***
## Trade_pc_100K    871.993    218.422   3.9922 6.768e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
vcovHC(lm1)
```

```
##               (Intercept)    aar_f2009   aar_f2010   aar_f2011   aar_f2012
## (Intercept)    9297989.37 -26519.17426 -34751.3931 -64358.9799 -88195.7750
## aar_f2009       -26519.17  42579.51052  22306.6988  22379.0191  22461.1963
## aar_f2010       -34751.39  22306.69876  41857.2132  22643.0594  22816.5776
## aar_f2011       -64358.98  22379.01911  22643.0594  45210.7304  23406.9880
## aar_f2012       -88195.78  22461.19628  22816.5776  23406.9880  47055.4187
## aar_f2013       -93332.22  22562.49160  23016.0483  23690.1311  24270.5328
## aar_f2014      -128032.51  22647.20878  23232.1454  24076.5421  24791.9383
## aar_f2015      -177893.27  22637.74268  23267.9132  24237.7165  25055.0255
## aar_f2016      -229170.12  22623.80635  23323.0788  24446.1520  25385.7301
## aar_f2017      -231919.09  22624.44448  23352.3686  24515.4258  25408.7607
## Total_ya_p     -134378.95     89.41919    277.8154    681.8928   1112.5721
## inc_k1          -48847.48    -46.78668   -117.7882    188.8338    193.4766
## inc_k5          -26724.41    110.78484    126.8286    397.1950    455.5137
## uni_k_mf        -23624.40   -129.42390   -212.3787   -468.5265   -572.7298
## uni_l_mf         79213.28    -45.36231   -237.3954   -324.3915   -491.9711
## Trade_pc_100K   145568.84    497.16540   1261.8579    987.3383    936.1196
```

```
##                  aar_f2013    aar_f2014    aar_f2015    aar_f2016    aar_f2017
## (Intercept)   -93332.21682 -128032.5143 -177893.2733 -229170.1243 -231919.0869
## aar_f2009      22562.49160   22647.2088   22637.7427   22623.8064   22624.4445
## aar_f2010      23016.04825   23232.1454   23267.9132   23323.0788   23352.3686
## aar_f2011      23690.13111   24076.5421   24237.7165   24446.1520   24515.4258
## aar_f2012      24270.53282   24791.9383   25055.0255   25385.7301   25408.7607
## aar_f2013      49220.90256   25428.8815   25755.4473   26135.5595   26169.5465
## aar_f2014      25428.88146   53475.4422   27156.8674   27482.0673   27045.3309
## aar_f2015      25755.44730   27156.8674   63394.1122   28309.5656   27655.2812
## aar_f2016      26135.55952   27482.0673   28309.5656   75087.4602   28071.1160
## aar_f2017      26169.54649   27045.3309   27655.2812   28071.1160   89424.5717
## Total_ya_p      1311.74280    1662.7240    2349.7551    3130.9906    3266.6554
## inc_k1           -23.25608     237.9932     438.1822     706.9105     723.9683
## inc_k5           419.80206     750.9501     927.6337    1166.2786    1178.1709
## uni_k_mf        -695.90501    -198.2867     136.4018    -110.1222    -816.2879
## uni_l_mf        -632.27758   -2195.0185   -3034.7846   -2540.7427   -1110.7783
## Trade_pc_100K   2510.69810    2684.4013    2764.2300     282.6406    1862.4720
##                  Total_ya_p       inc_k1      inc_k5     uni_k_mf    uni_l_mf
## (Intercept)   -134378.94615 -48847.47803 -26724.4053 -23624.40438 79213.27980
## aar_f2009          89.41919    -46.78668    110.7848   -129.42390   -45.36231
## aar_f2010         277.81538   -117.78822    126.8286   -212.37867  -237.39541
## aar_f2011         681.89276    188.83384    397.1950   -468.52650  -324.39148
## aar_f2012        1112.57212    193.47663    455.5137   -572.72977  -491.97106
## aar_f2013        1311.74280    -23.25608    419.8021   -695.90501  -632.27758
## aar_f2014        1662.72401    237.99318    750.9501   -198.28673 -2195.01848
## aar_f2015        2349.75511    438.18220    927.6337    136.40176 -3034.78456
## aar_f2016        3130.99055    706.91052   1166.2786   -110.12216 -2540.74265
## aar_f2017        3266.65535    723.96826   1178.1709   -816.28793 -1110.77830
## Total_ya_p       2167.75020    426.37025    133.2185     51.21924  -614.02732
## inc_k1            426.37025    801.89764    496.4444    158.26504  -500.25996
## inc_k5            133.21845    496.44438    547.3448    104.53767  -690.28424
## uni_k_mf           51.21924    158.26504    104.5377   1515.96690 -2398.54359
## uni_l_mf         -614.02732   -500.25996   -690.2842  -2398.54359  5463.68941
## Trade_pc_100K   -1619.34164  -2293.03278   -115.1786  -2608.77275   651.94105
##                Trade_pc_100K
## (Intercept)     145568.8365
## aar_f2009          497.1654
## aar_f2010         1261.8579
## aar_f2011          987.3383
## aar_f2012          936.1196
## aar_f2013         2510.6981
## aar_f2014         2684.4013
## aar_f2015         2764.2300
## aar_f2016          282.6406
## aar_f2017         1862.4720
## Total_ya_p       -1619.3416
## inc_k1           -2293.0328
## inc_k5            -115.1786
## uni_k_mf         -2608.7728
## uni_l_mf           651.9410
## Trade_pc_100K    60897.1826
```

#Flytter residualene fra lm1 til datasettet pm2

```
pm2 <- pm2 %>%
  add_residuals(lm1)
```

lager ny variabel

```
pm2 <- pm2 %>%
mutate(aar_d = make_date(aar))
```

Vi filterer ut fylkene Østfold, Akershus, Oslo, Rogaland og Hordaland

```
pm2 <- pm2 %>%
mutate(fylke = substr(knr, start = 1, stop = 2))
```

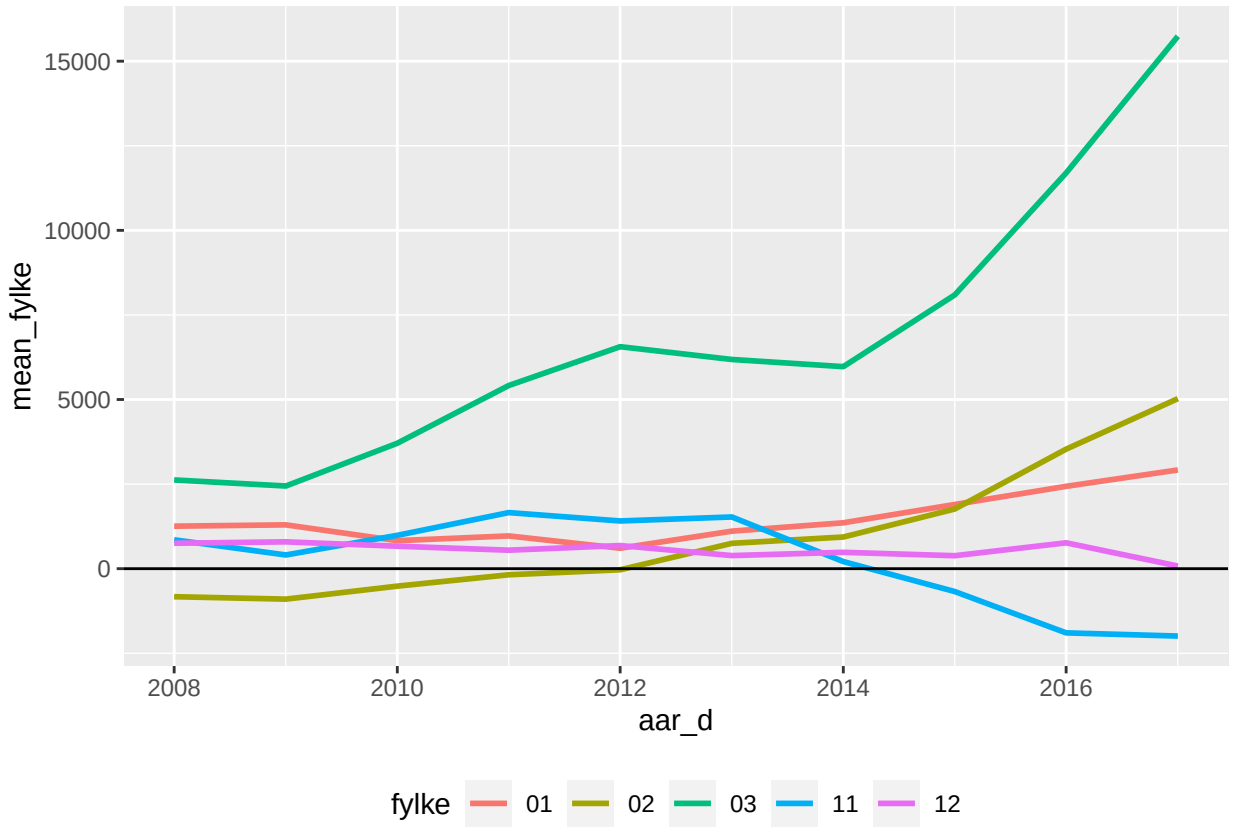
```
pm2_red <- pm2 %>%
  filter(fylke %in% c("01", "02", "03", "11", "12"))
```

Oppgave 7 til 10

```
# pm2_red %>%
#   unnest(c(fylke)) %>%
#   group_by(fylke, aar_d) %>%
#   summarise(mean_fylke = mean(resid)) %>%
#   ggplot(mapping = aes(x = aar_d, y = mean_fylke, colour = fylke)) +
#   geom_line(lwd = 1) +
#   geom_hline(yintercept = 0, colour = "white") +
#   theme(legend.position = "bottom")
```

```
pm2_red %>%
  filter(fylke %in% c("01", "02", "03", "11", "12")) %>%
  unnest(c(fylke)) %>%
  group_by (fylke, aar_d) %>%
  summarise(mean_fylke = mean(resid)) %>%
  ggplot(aes(x=aar_d, y=mean_fylke, colour = fylke)) +
  geom_line(lwd=1) +
  geom_hline(yintercept = 0, colour = "black") +
  theme(legend.position = "bottom")
```

```
## `summarise()` has grouped output by 'fylke'. You can override using the `.groups` argument.
```



#Dummy fylke og år

Vi innfører en dummy for hvert fylke

```
# Nå har vi hel pm2 å bruke
mod2 <- 'pm2 ~ fnr*aar_f + Total_ya_p + inc_k1 + inc_k5 + uni_k_mf + uni_l_mf + Trade_pc_100K'
```

Vi genererer lm 2 fra modell 2 og datasettet pm2

```
lm2 <- lm(mod2, data = pm2)
```

```
summary(lm2)
```

```
##
## Call:
## lm(formula = mod2, data = pm2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -8546  -1191     32   1198   8328
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -21200.688   2521.645  -8.407  < 2e-16 ***
## fnr02            -1482.789    702.970  -2.109 0.035045 *
## fnr03             3248.234   2190.443   1.483 0.138260
## fnr04            -1049.219    774.264  -1.355 0.175537
## fnr05            -1937.388    758.293  -2.555 0.010696 *
## fnr06            -2172.731    772.094  -2.814 0.004941 **
## fnr07             -737.995   1080.348  -0.683 0.494620
```

```
## fnr08           -3213.279    878.620  -3.657 0.000262 ***
## fnr09           -1219.813    913.691  -1.335 0.182020
## fnr10            -281.375    852.265  -0.330 0.741323
## fnr11            -565.360    771.927  -0.732 0.464012
## fnr12            -903.071    742.464  -1.216 0.224012
## fnr14           -3339.829   1182.013  -2.826 0.004768 **
## fnr15           -3619.198    715.832  -5.056 4.69e-07 ***
## fnr16           -1093.217    759.677  -1.439 0.150296
## fnr17           -2005.965    917.216  -2.187 0.028860 *
## fnr18           -1567.503    774.530  -2.024 0.043126 *
## fnr19           -2856.881   1326.142  -2.154 0.031341 *
## fnr20           -2656.315   1180.088  -2.251 0.024500 *
## aar_f2009          94.009    744.240   0.126 0.899496
## aar_f2010         417.129    744.379   0.560 0.575290
## aar_f2011        1280.914    744.731   1.720 0.085597 .
## aar_f2012        1455.525    745.679   1.952 0.051088 .
## aar_f2013        2479.533    746.367   3.322 0.000910 ***
## aar_f2014        2795.831    747.254   3.741 0.000188 ***
## aar_f2015        3987.973    748.109   5.331 1.09e-07 ***
## aar_f2016        5264.965    749.169   7.028 2.89e-12 ***
## aar_f2017        6618.572    749.430   8.831  < 2e-16 ***
## Total_ya_p        511.787     36.100  14.177  < 2e-16 ***
## inc_k1           -243.050     27.007  -9.000  < 2e-16 ***
## inc_k5            251.645     22.916  10.981  < 2e-16 ***
## uni_k_mf          178.253     28.157   6.331 3.02e-10 ***
## uni_l_mf          732.442     42.235  17.342  < 2e-16 ***
## Trade_pc_100K    1067.760    190.885   5.594 2.54e-08 ***
## fnr02:aar_f2009   -40.505    978.026  -0.041 0.966969
## fnr03:aar_f2009    84.133   3068.211   0.027 0.978127
## fnr04:aar_f2009  -330.219   1089.318  -0.303 0.761813
## fnr05:aar_f2009   416.862   1069.758   0.390 0.696816
## fnr06:aar_f2009  -163.759   1089.292  -0.150 0.880516
## fnr07:aar_f2009   134.353   1525.051   0.088 0.929808
## fnr08:aar_f2009   329.317   1240.237   0.266 0.790631
## fnr09:aar_f2009   686.715   1288.922   0.533 0.594245
## fnr10:aar_f2009   231.288   1199.909   0.193 0.847172
## fnr11:aar_f2009  -414.412   1069.772  -0.387 0.698515
## fnr12:aar_f2009    21.853   1036.805   0.021 0.983186
## fnr14:aar_f2009  -220.698   1663.985  -0.133 0.894498
## fnr15:aar_f2009   205.720    998.429   0.206 0.836779
## fnr16:aar_f2009  -346.631   1069.772  -0.324 0.745955
## fnr17:aar_f2009  -288.412   1288.940  -0.224 0.822969
## fnr18:aar_f2009  -148.285   1089.412  -0.136 0.891744
## fnr19:aar_f2009   453.061   1872.733   0.242 0.808864
## fnr20:aar_f2009  -927.061   1664.164  -0.557 0.577542
## fnr02:aar_f2010   792.694    978.020   0.811 0.417747
## fnr03:aar_f2010  2004.378   3068.354   0.653 0.513677
## fnr04:aar_f2010  -191.813   1089.355  -0.176 0.860250
## fnr05:aar_f2010   655.342   1069.794   0.613 0.540221
## fnr06:aar_f2010   189.332   1089.409   0.174 0.862046
## fnr07:aar_f2010   728.914   1525.112   0.478 0.632745
## fnr08:aar_f2010  1281.636   1240.345   1.033 0.301597
## fnr09:aar_f2010   986.486   1288.914   0.765 0.444149
## fnr10:aar_f2010   924.121   1199.916   0.770 0.441302
```

```
## fnr11:aar_f2010    642.468   1069.866   0.601 0.548235
## fnr12:aar_f2010    381.898   1036.801   0.368 0.712658
## fnr14:aar_f2010    536.844   1663.957   0.323 0.747009
## fnr15:aar_f2010    548.008    998.671   0.549 0.583249
## fnr16:aar_f2010   -237.962   1069.934  -0.222 0.824020
## fnr17:aar_f2010   -422.338   1289.001  -0.328 0.743214
## fnr18:aar_f2010    402.939   1089.510   0.370 0.711545
## fnr19:aar_f2010    982.125   1872.779   0.524 0.600045
## fnr20:aar_f2010   -547.207   1664.063  -0.329 0.742313
## fnr02:aar_f2011    992.480    978.070   1.015 0.310359
## fnr03:aar_f2011   3891.025   3068.768   1.268 0.204970
## fnr04:aar_f2011   -775.700   1089.399  -0.712 0.476523
## fnr05:aar_f2011    183.865   1069.834   0.172 0.863563
## fnr06:aar_f2011     33.963   1089.394   0.031 0.975132
## fnr07:aar_f2011    275.017   1525.266   0.180 0.856930
## fnr08:aar_f2011    646.495   1240.336   0.521 0.602269
## fnr09:aar_f2011    599.582   1288.944   0.465 0.641860
## fnr10:aar_f2011    168.648   1199.944   0.141 0.888243
## fnr11:aar_f2011   1243.418   1070.024   1.162 0.245359
## fnr12:aar_f2011    165.379   1036.901   0.159 0.873297
## fnr14:aar_f2011   1984.847   1664.012   1.193 0.233090
## fnr15:aar_f2011    463.880    998.884   0.464 0.642414
## fnr16:aar_f2011   -497.945   1069.952  -0.465 0.641705
## fnr17:aar_f2011    257.671   1289.086   0.200 0.841590
## fnr18:aar_f2011    252.454   1089.674   0.232 0.816812
## fnr19:aar_f2011   -669.729   1872.850  -0.358 0.720682
## fnr20:aar_f2011   -542.321   1664.293  -0.326 0.744568
## fnr02:aar_f2012   1565.161    978.102   1.600 0.109716
## fnr03:aar_f2012   5674.403   3069.281   1.849 0.064642 .
## fnr04:aar_f2012   -808.528   1089.510  -0.742 0.458115
## fnr05:aar_f2012    820.104   1070.017   0.766 0.443507
## fnr06:aar_f2012    800.976   1089.455   0.735 0.462302
## fnr07:aar_f2012   1047.940   1525.235   0.687 0.492122
## fnr08:aar_f2012   1090.416   1240.413   0.879 0.379470
## fnr09:aar_f2012   1071.846   1289.011   0.832 0.405779
## fnr10:aar_f2012    321.458   1200.216   0.268 0.788856
## fnr11:aar_f2012   1467.212   1070.665   1.370 0.170728
## fnr12:aar_f2012    669.171   1037.128   0.645 0.518864
## fnr14:aar_f2012   1739.551   1664.177   1.045 0.296018
## fnr15:aar_f2012    463.860    999.265   0.464 0.642556
## fnr16:aar_f2012    380.682   1070.437   0.356 0.722154
## fnr17:aar_f2012    637.493   1289.624   0.494 0.621133
## fnr18:aar_f2012    482.679   1089.761   0.443 0.657871
## fnr19:aar_f2012    727.671   1872.902   0.389 0.697670
## fnr20:aar_f2012   -378.342   1664.741  -0.227 0.820240
## fnr02:aar_f2013   1953.373    978.298   1.997 0.045996 *
## fnr03:aar_f2013   5108.375   3070.149   1.664 0.096297 .
## fnr04:aar_f2013  -1206.685   1089.615  -1.107 0.268240
## fnr05:aar_f2013   -198.536   1070.094  -0.186 0.852832
## fnr06:aar_f2013    410.281   1089.375   0.377 0.706497
## fnr07:aar_f2013    890.998   1525.236   0.584 0.559173
## fnr08:aar_f2013    575.599   1240.249   0.464 0.642628
## fnr09:aar_f2013     64.585   1289.204   0.050 0.960050
## fnr10:aar_f2013   -515.180   1200.200  -0.429 0.667793
```

```
## fnr11:aar_f2013   1179.371   1071.062   1.101 0.270979
## fnr12:aar_f2013    -69.430   1037.183  -0.067 0.946636
## fnr14:aar_f2013    208.353   1664.208   0.125 0.900381
## fnr15:aar_f2013      7.994    999.213   0.008 0.993617
## fnr16:aar_f2013   -347.235   1070.757  -0.324 0.745754
## fnr17:aar_f2013    203.405   1289.762   0.158 0.874704
## fnr18:aar_f2013    201.272   1090.026   0.185 0.853524
## fnr19:aar_f2013    278.261   1873.128   0.149 0.881921
## fnr20:aar_f2013  -1110.163   1664.836  -0.667 0.504960
## fnr02:aar_f2014   2019.269    978.649   2.063 0.039214 *
## fnr03:aar_f2014   4938.603   3071.105   1.608 0.107979
## fnr04:aar_f2014  -1456.367   1089.708  -1.336 0.181550
## fnr05:aar_f2014   -254.055   1070.253  -0.237 0.812388
## fnr06:aar_f2014    571.152   1089.474   0.524 0.600167
## fnr07:aar_f2014    582.123   1525.332   0.382 0.702772
## fnr08:aar_f2014    689.084   1240.251   0.556 0.578548
## fnr09:aar_f2014   -186.541   1289.179  -0.145 0.884965
## fnr10:aar_f2014   -674.319   1200.339  -0.562 0.574335
## fnr11:aar_f2014   -183.391   1071.523  -0.171 0.864124
## fnr12:aar_f2014   -147.825   1037.277  -0.143 0.886690
## fnr14:aar_f2014    253.302   1664.812   0.152 0.879084
## fnr15:aar_f2014   -481.056    999.093  -0.481 0.630220
## fnr16:aar_f2014   -229.362   1070.812  -0.214 0.830418
## fnr17:aar_f2014    -61.073   1289.824  -0.047 0.962239
## fnr18:aar_f2014   -393.115   1090.258  -0.361 0.718459
## fnr19:aar_f2014   1688.165   1873.121   0.901 0.367563
## fnr20:aar_f2014  -1563.827   1665.176  -0.939 0.347778
## fnr02:aar_f2015   2401.120    979.036   2.453 0.014273 *
## fnr03:aar_f2015   6985.367   3073.112   2.273 0.023131 *
## fnr04:aar_f2015  -1912.336   1089.754  -1.755 0.079446 .
## fnr05:aar_f2015  -1326.089   1070.254  -1.239 0.215480
## fnr06:aar_f2015     22.631   1089.626   0.021 0.983431
## fnr07:aar_f2015    990.944   1525.354   0.650 0.515996
## fnr08:aar_f2015   -776.910   1240.290  -0.626 0.531130
## fnr09:aar_f2015  -1242.730   1289.232  -0.964 0.335201
## fnr10:aar_f2015  -1492.749   1200.502  -1.243 0.213856
## fnr11:aar_f2015  -1489.385   1072.451  -1.389 0.165063
## fnr12:aar_f2015   -711.755   1037.476  -0.686 0.492767
## fnr14:aar_f2015  -1695.187   1665.139  -1.018 0.308783
## fnr15:aar_f2015   -587.449    999.385  -0.588 0.556727
## fnr16:aar_f2015   -139.973   1070.880  -0.131 0.896019
## fnr17:aar_f2015   -867.834   1289.740  -0.673 0.501107
## fnr18:aar_f2015   -439.127   1090.372  -0.403 0.687190
## fnr19:aar_f2015    369.085   1873.412   0.197 0.843839
## fnr20:aar_f2015  -3266.760   1665.444  -1.961 0.049964 *
## fnr02:aar_f2016   3656.344    979.067   3.735 0.000193 ***
## fnr03:aar_f2016  10264.572   3074.072   3.339 0.000856 ***
## fnr04:aar_f2016  -2459.017   1089.893  -2.256 0.024169 *
## fnr05:aar_f2016  -2117.228   1070.338  -1.978 0.048059 *
## fnr06:aar_f2016   -598.671   1089.701  -0.549 0.582801
## fnr07:aar_f2016    447.813   1525.278   0.294 0.769099
## fnr08:aar_f2016  -1716.491   1240.468  -1.384 0.166595
## fnr09:aar_f2016  -1987.219   1289.181  -1.541 0.123368
## fnr10:aar_f2016  -3090.918   1200.777  -2.574 0.010124 *
```

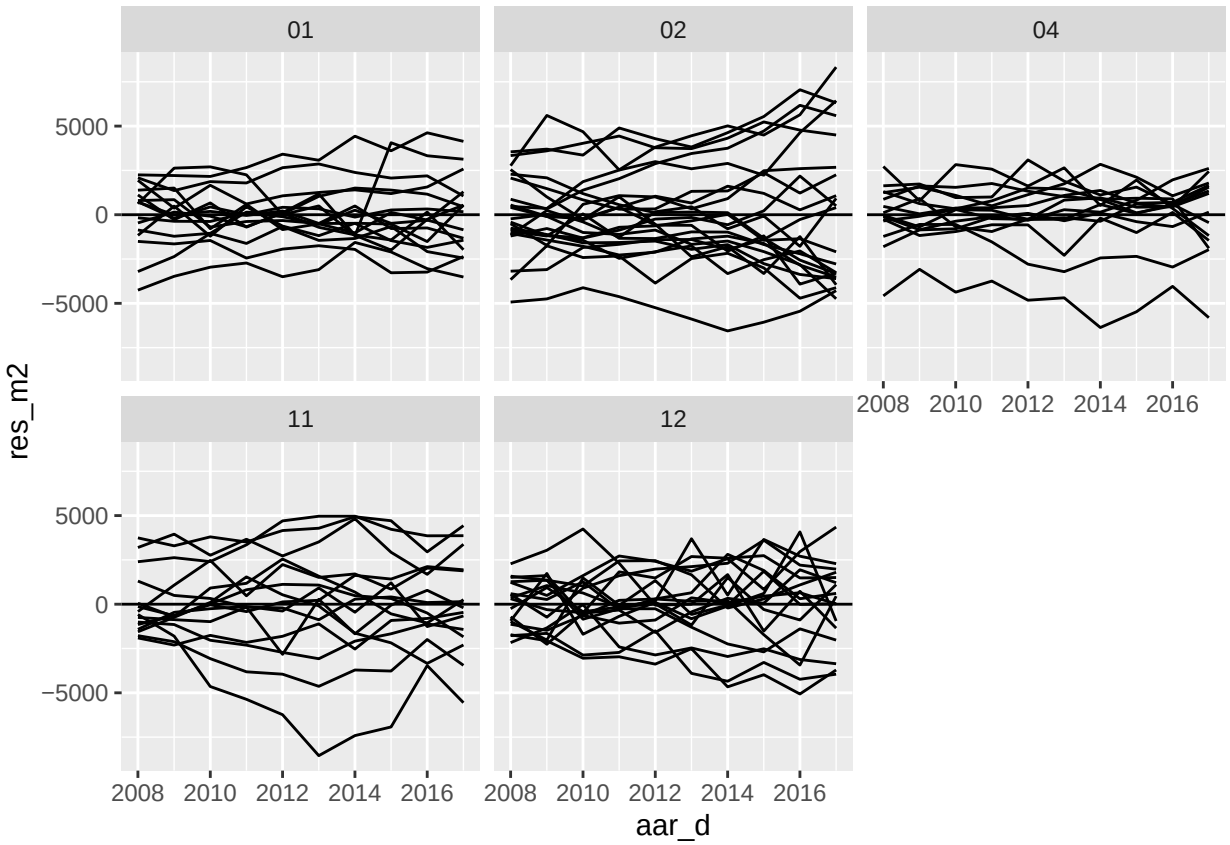
```
## fnr11:aar_f2016  -3274.743   1072.946  -3.052 0.002303 **
## fnr12:aar_f2016   -901.775   1037.688  -0.869 0.384941
## fnr14:aar_f2016  -1552.417   1665.259  -0.932 0.351330
## fnr15:aar_f2016  -1872.887    999.582  -1.874 0.061126 .
## fnr16:aar_f2016  -1074.143   1070.970  -1.003 0.316004
## fnr17:aar_f2016  -1612.215   1290.487  -1.249 0.211703
## fnr18:aar_f2016  -1361.291   1090.771  -1.248 0.212178
## fnr19:aar_f2016    906.286   1873.612   0.484 0.628646
## fnr20:aar_f2016  -3169.910   1665.821  -1.903 0.057200 .
## fnr02:aar_f2017   4707.776    979.374   4.807 1.65e-06 ***
## fnr03:aar_f2017  13986.613   3075.071   4.548 5.74e-06 ***
## fnr04:aar_f2017  -3549.658   1089.920  -3.257 0.001146 **
## fnr05:aar_f2017  -2397.820   1070.176  -2.241 0.025165 *
## fnr06:aar_f2017     60.036   1089.704   0.055 0.956069
## fnr07:aar_f2017    960.018   1525.236   0.629 0.529146
## fnr08:aar_f2017  -2045.538   1240.415  -1.649 0.099294 .
## fnr09:aar_f2017  -3223.036   1289.344  -2.500 0.012510 *
## fnr10:aar_f2017  -3807.142   1200.767  -3.171 0.001545 **
## fnr11:aar_f2017  -3863.610   1073.185  -3.600 0.000326 ***
## fnr12:aar_f2017  -2046.447   1038.104  -1.971 0.048828 *
## fnr14:aar_f2017  -2074.192   1665.271  -1.246 0.213077
## fnr15:aar_f2017  -2799.827    999.681  -2.801 0.005149 **
## fnr16:aar_f2017  -2278.453   1070.923  -2.128 0.033499 *
## fnr17:aar_f2017  -2761.733   1290.527  -2.140 0.032479 *
## fnr18:aar_f2017  -2661.041   1090.689  -2.440 0.014785 *
## fnr19:aar_f2017   -716.410   1873.886  -0.382 0.702272
## fnr20:aar_f2017  -3922.387   1665.464  -2.355 0.018615 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2105 on 1944 degrees of freedom
## Multiple R-squared:  0.8953, Adjusted R-squared:  0.8848
## F-statistic: 85.21 on 195 and 1944 DF,  p-value: < 2.2e-16
```

Vi legger inn residualene fra lm2 til pm2 og kaller dem res_m2

```
pm2 <- pm2 %>%
  mutate(res_m2 = resid(lm2))
```

Vi filtrer fylkene

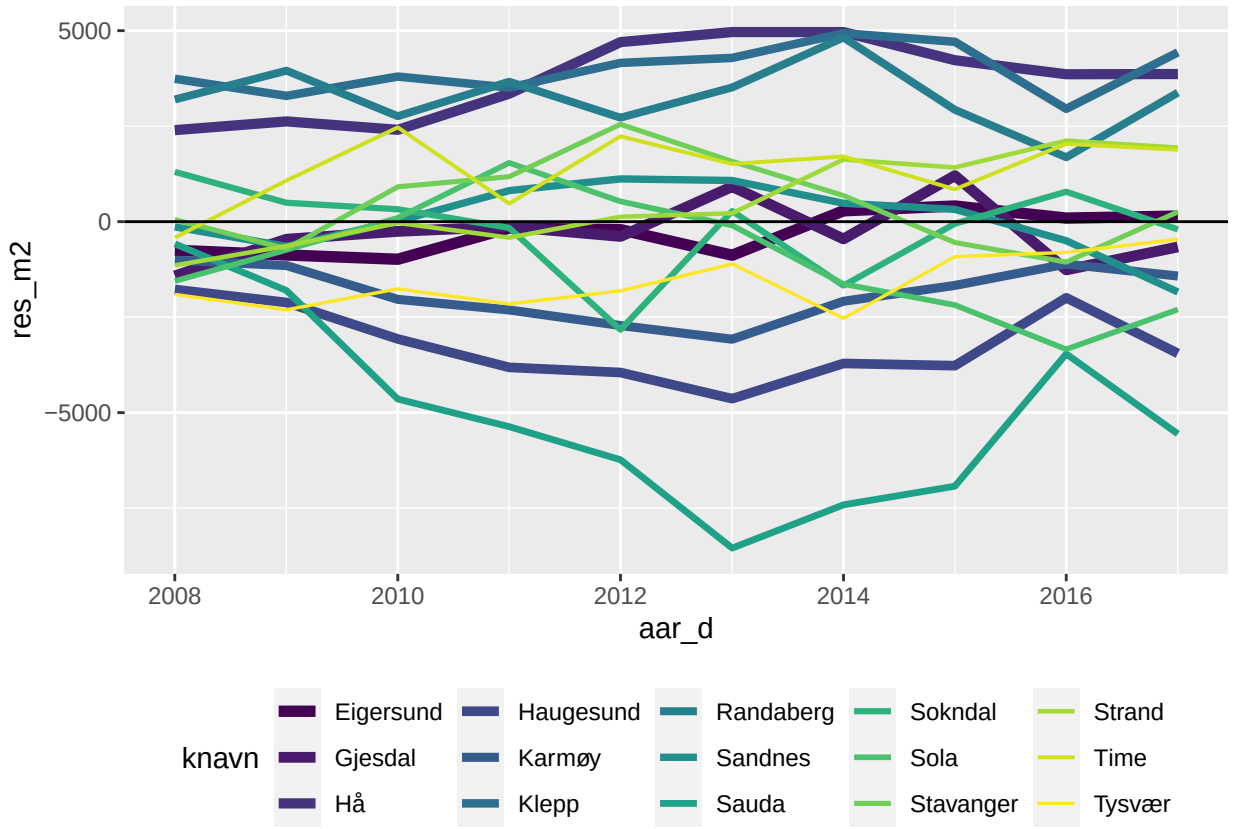
```
pm2 %>% filter(fnr %in% c("01", "02", "04", "11", "12")) %>%
ggplot(mapping = aes(x = aar_d, y = res_m2)) +
geom_line(aes(group = knavn)) +
scale_size_manual(values = c(seq(2.0, 0.5, by = -0.1))) +
geom_hline(yintercept = 0) +
theme(legend.position = 'bottom') +
  facet_wrap(~fylke)
```



Ut fra modell 2 kan de være vanskelig å eksakte svar. med flere variabler ville nok bilde blitt et annet.

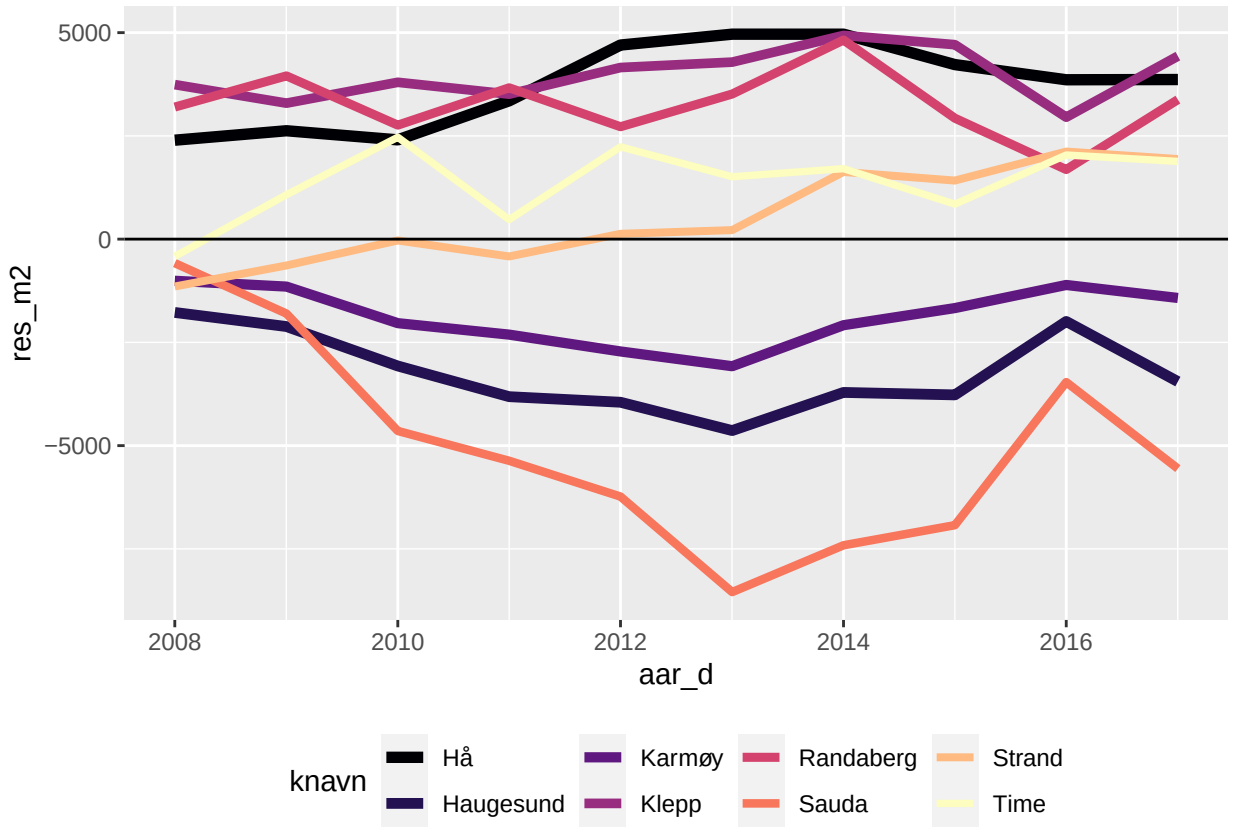
#Vi filtrerer med hensyn på fylke "11"

```
pm2 %>% filter(fnr %in% c("11")) %>%
ggplot(mapping = aes(x = aar_d, y = res_m2)) +
scale_color_viridis(discrete = TRUE, option = "D") +
geom_line(aes(group = knavn, colour = knavn, size = knavn)) +
scale_size_manual(values = c(seq(2.0, 0.5, by = -0.1))) +
geom_hline(yintercept = 0) +
theme(legend.position = 'bottom')
```



#Vi gjentar plottet ovenfor med utvalgte kommuner

```
pm2 %>% filter(knr %in% c("1119", "1120", "1127", "1121", "1130", "1135", "1106", "1149")) %>%
ggplot(mapping = aes(x = aar_d, y = res_m2)) +
scale_color_viridis(discrete = TRUE, option = "A") +
geom_line(aes(group = knavn, colour = knavn, size =knavn)) +
scale_size_manual(values = c(seq(2.0, 0.5, by = -0.1))) +
geom_hline(yintercept = 0) +
theme(legend.position = 'bottom')
```



Det som kjennetegner de utvalgte kommunene i Rogaland stor spredning fra kommunene. vi kan se at Sauda blir undervurdert i forhold til gjennomsnittet mens Hå og Klepp blir overvurdert. som vil si de har dyrere kvadratmeter pris i forhold til gjennomsnittet.

#Modell for hvert år

i.

```
pm2_n <- pm2 %>%
# velger først variablene. Tar med aar_d
   select(pm2, fnr, knr, aar, aar_f, aar_d, Menn_ya_p,
          Kvinner_ya_p, Total_ya_p, inc_k1, inc_k5, uni_k_mf,
          uni_l_mf, Trade_pc_100K) %>%
  # grupperer mht. aar_d som er date object
  group_by(aar_d) %>%
  nest()
```

```
pm2_n
```

```
## # A tibble: 10 x 2
## # Groups:   aar_d [10]
##    aar_d      data
##    <date>     <list>
##  1 2008-01-01 <tibble [214 x 13]>
##  2 2009-01-01 <tibble [214 x 13]>
##  3 2010-01-01 <tibble [214 x 13]>
##  4 2011-01-01 <tibble [214 x 13]>
##  5 2012-01-01 <tibble [214 x 13]>
##  6 2013-01-01 <tibble [214 x 13]>
##  7 2014-01-01 <tibble [214 x 13]>
```

```
##  8 2015-01-01 <tibble [214 x 13]>
##  9 2016-01-01 <tibble [214 x 13]>
## 10 2017-01-01 <tibble [214 x 13]>
```

```
# Ser bra ut ;-)
pm2_n$data[[1]] %>%
 head(n = 5)
```

```
## # A tibble: 5 x 13
##     pm2 fnr   knr     aar aar_f Menn_ya_p Kvinner_ya_p Total_ya_p inc_k1 inc_k5
##   <dbl> <chr> <chr> <dbl> <chr>     <dbl>        <dbl>      <dbl>  <dbl>  <dbl>
## 1 13427 01    0101   2008 2008       59.7         56.8       58.3   24.5   13.6
## 2 18299 01    0104   2008 2008       60.7         58.7       59.7   22.8   16.2
## 3 14981 01    0105   2008 2008       60.9         58.1       59.5   22.2   13.6
## 4 15671 01    0106   2008 2008       59.8         57.8       58.8   21.8   16.2
## 5 18844 01    0111   2008 2008       61.7         61.3       61.5   17.8   19
## # ... with 3 more variables: uni_k_mf <dbl>, uni_l_mf <dbl>,
## #   Trade_pc_100K <dbl>
```

```
dim(pm2_n)
```

```
## [1] 10  2
```

```
# må bruke a_df i funksjonen ellers vil den alltid bruke pm2
kom_model <- function(a_df) {
   lm(pm2 ~ fnr + Total_ya_p + inc_k1 + inc_k5 + uni_k_mf + uni_l_mf + Trade_pc_100K, data = a_df)
}
```

```
pm2_n <- pm2_n %>%
   mutate(model = map(data, .f = kom_model))
```

```
# kom_model(pm2_n$aar) %>%
# summary()
```

```
mod_sum <- pm2_n %>%
   mutate(mod_summary = map(.x = model, .f = glance)) %>%
   unnest(mod_summary) %>%
   print()
```

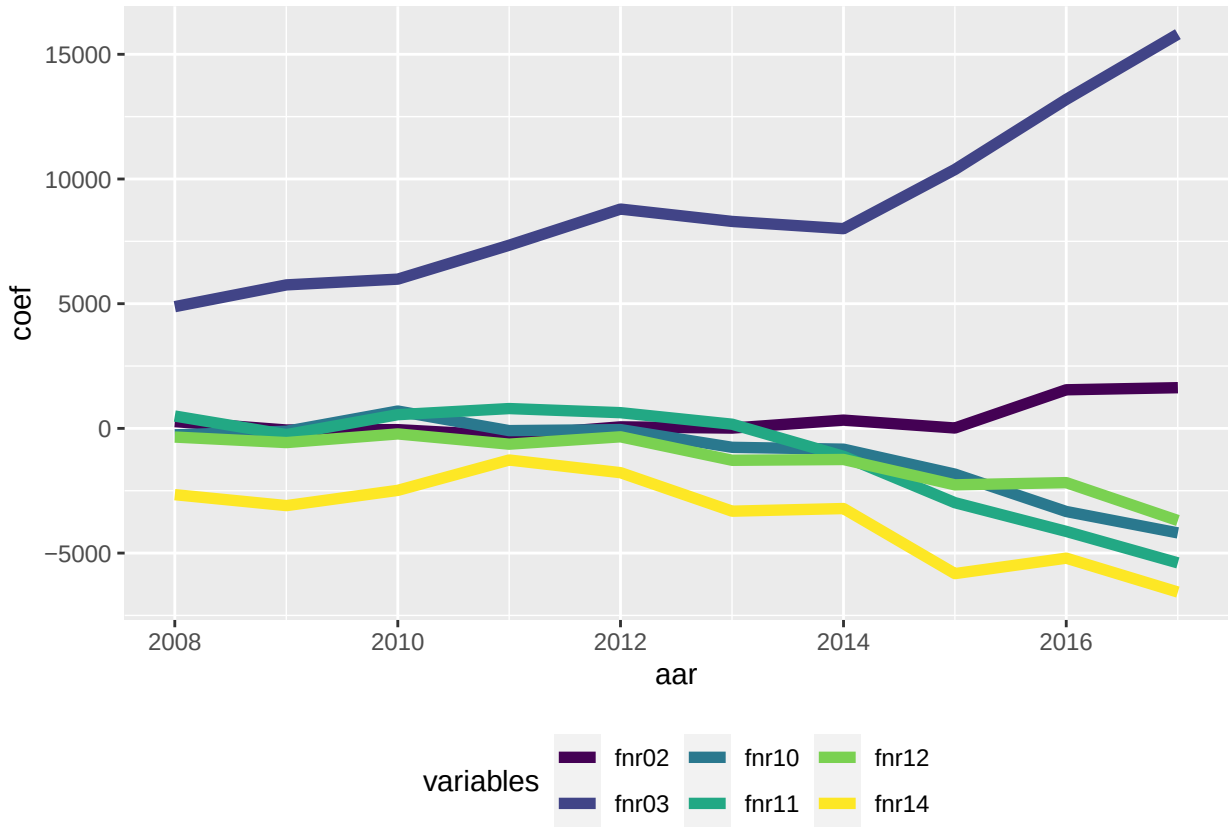
```
## # A tibble: 10 x 15
## # Groups:   aar_d [10]
##    aar_d      data  model r.squared adj.r.squared sigma statistic  p.value    df
##    <date>     <lis> <lis>     <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>
##  1 2008-01-01 <tib~ <lm>      0.873         0.857 1701.      54.2 1.19e-71    24
##  2 2009-01-01 <tib~ <lm>      0.886         0.871 1614.      61.2 5.63e-76    24
##  3 2010-01-01 <tib~ <lm>      0.888         0.874 1743.      62.4 1.13e-76    24
##  4 2011-01-01 <tib~ <lm>      0.883         0.868 1925.      59.4 6.50e-75    24
##  5 2012-01-01 <tib~ <lm>      0.891         0.877 1953.      64.2 1.06e-77    24
##  6 2013-01-01 <tib~ <lm>      0.895         0.881 2026.      67.0 3.03e-79    24
##  7 2014-01-01 <tib~ <lm>      0.884         0.869 2149.      60.1 2.30e-75    24
##  8 2015-01-01 <tib~ <lm>      0.879         0.863 2361.      57.1 1.57e-73    24
##  9 2016-01-01 <tib~ <lm>      0.883         0.869 2467.      59.7 4.19e-75    24
## 10 2017-01-01 <tib~ <lm>      0.895         0.882 2614.      67.0 2.84e-79    24
## # ... with 6 more variables: logLik <dbl>, AIC <dbl>, BIC <dbl>,
## #   deviance <dbl>, df.residual <int>, nobs <int>
```

```r
coef_df <- mod_sum$model %>%
  # 1 plukker ut koeffisientene
   map_df(1) %>%
  # legges i en tibble
   tibble()
```

```r
# Lager en års variabel i coef_df
coef_df <- coef_df %>%
  mutate(
    aar = ymd(paste(2008:2017, "-01-01", sep = ""))
  ) %>%
  select(aar, everything())
```

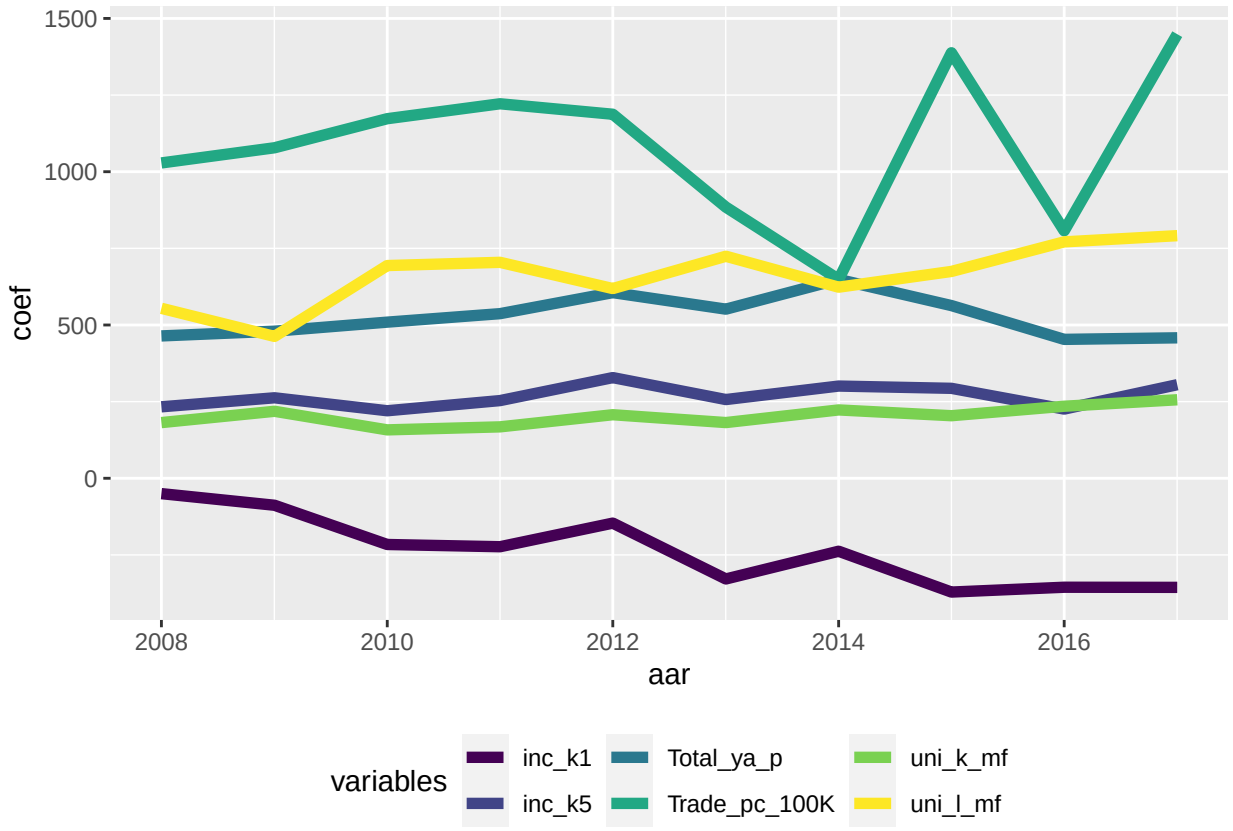
```r
# Gjør variabelnavn om til en variabel kalt "variables"
# Verdien på koeffisientene for de ulike variablene i ulike år
# legges i variabelen coef
coef_df_long <- coef_df %>%
   pivot_longer(
     # Tar IKKE med aar
     cols = `(Intercept)`:`Trade_pc_100K`,
     names_to = "variables",
     values_to = "coef")
```

```r
coef_df_long %>%
  # Plukker ut de relevante fylkene
  filter(
    variables %in% c("fnr02", "fnr03", "fnr10", "fnr11", "fnr12", "fnr14")) %>%
 ggplot(mapping = aes(x = aar, y = coef, colour = variables)) +
 scale_color_viridis(discrete = TRUE, option = "D") +
  geom_line(aes(group = variables), lwd = 2) +
  theme(legend.position = 'bottom')
```



I 2014 hadde vi et kraftig fall i råolje-prisen. Det ser ut til å ha gitt et fall i prisen per kvadratmeter på Sør og Vestlandet, mens effekten har vært motsatt for Akershus og særlig Oslo. I modellen er det justert for effekten av befolkning i yrkesaktiv alder, andel av husholdninger i laveste inntekts-kvintil, andel av husholdninger i høyeste inntekts-kvintil, andel av befolkning med kort universitets/høgskole-utdanning, andel av befolkning med lang universitets/høgskole-utdanning og omsetning i detaljhandelen per innbygger. Med unntak av variabelen "Trade_pc_100K" ser koeffisientene til disse variablene ut til å være rimelig stabile over tid.

```
coef_df_long %>%
  select(aar, variables, coef) %>%
  # rettet vaiabelnavnene til inc_k1 og inc_k2
  filter(
    variables %in% c("Total_ya_p", "inc_k1", "inc_k5", "uni_k_mf", "uni_l_mf", "Trade_pc_100K")) %>%
  ggplot(mapping = aes(x = aar, y = coef, colour = variables)) +
  scale_color_viridis(discrete = TRUE, option = "D") +
  geom_line(aes(group = variables), lwd = 2) +
  theme(legend.position = 'bottom')
```



Alle er stabile med unntak av Trade_pc_100K.