Met4 Home Exam Fall 2020

Setup

```
rm(list = ls())
library(docstring)
library(tidyverse)
library(magrittr)
library(readxl)
library(ggplot2)
library(httr)
library(rjstat)
library(zoo)
library(rlang)

ssb_bnp_url <-
  "https://data.ssb.no/api/v0/no/table/09190/"

ssb_api_call <-
  '{
    "query":[{
      "code":"Makrost", "selection":{
        "filter":"item", "values":["bnpb.nr23_9fn"]
      }
    }, {
      "code":"ContentsCode", "selection":{
        "filter":"item", "values":["Volum"]
      }
    }], "response":{
      "format":"json - stat2"
    }
  }' %>%
  gsub("[[:space:]]", "", .)


df_latest_release <-
  POST(ssb_bnp_url, body = ssb_api_call, encode = "json", verbose()) %>%
  content("text") %>%
  fromJSONstat() %>%
  transmute(
    quarter = as.yearqtr(kvartal, format = "%YK%q"),
    Y = value) %>%
  mutate(
    lag_1_Y = lag(Y, n = 1, order_by = quarter),
    lag_2_Y = lag(Y, n = 2, order_by = quarter))
```

```
df_revisions <-
  read_excel("bnpfn.xlsx", sheet = "kvartalsvis") %>%
  rename(quarter = `...1`) %>%
  pivot_longer(
    cols = colnames(.)[-1],
    names_to = "pubdato",
    values_to = "Y")  %>%
  mutate(
    quarter = as.yearqtr(quarter, format = "%YK%q"),
    pubdato = as.yearqtr(pubdato, format = "%YK%q")
  ) %>%
  mutate(version = paste0("Y", (pubdato - quarter) * 4 + 1)) %>%
  filter(!is.na(Y)) %>%
  filter(quarter >= min(.$pubdato)) %>%
  arrange(quarter, pubdato) %>%
  mutate(
    main_revision =
      case_when(
        pubdato < as.yearqtr("2006K3", format = "%YK%q")~1,
        pubdato < as.yearqtr("2011K3", format = "%YK%q")~2,
        TRUE~3))

df_revisions_wide <-
  df_revisions %>%
  pivot_wider(
    id_cols = quarter,
    names_from = version,
    values_from = Y
  )
```

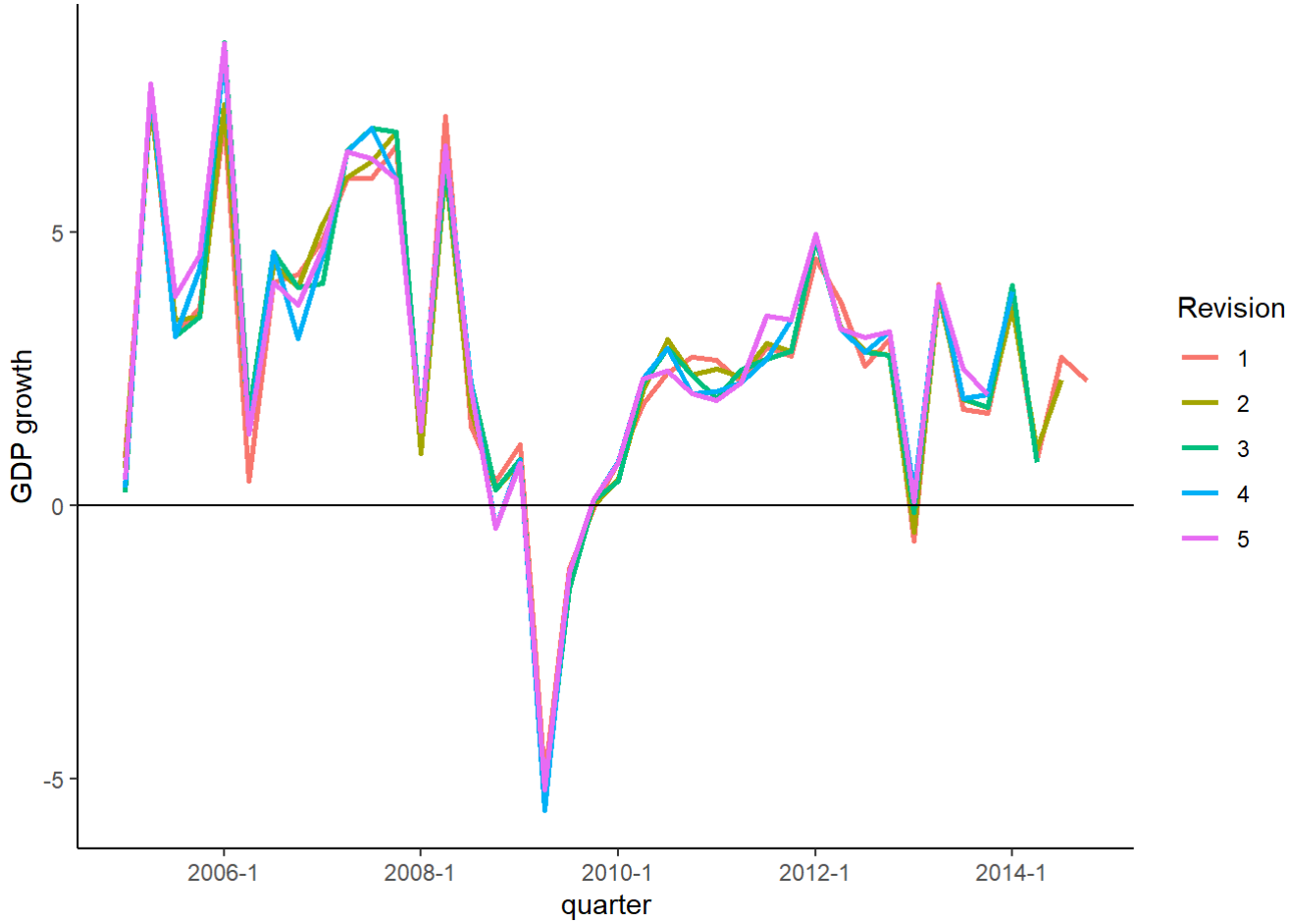
Q1

We start by creating a figure over the growth rates from the five first vintages. We can see that the series are quite similar, however there are some periods where there is greater disagreement between the vintages (e.g. before the financial crisis, and around 2011).

```
df_revisions_long <-
  df_revisions_wide %>%
  select(quarter, Y1, Y2, Y3, Y4, Y5) %>%
  pivot_longer(
    cols = starts_with("Y"),
    names_to = "Revision",
    values_to = "GDP growth") %>%
  mutate(Revision = as.factor(gsub("Y","",Revision)))


df_revisions_long %>%
  ggplot(aes(x=quarter, y = `GDP growth`, col=Revision)) +
  geom_line(lwd=1) +
  geom_hline(yintercept = 0) +
  theme_classic()
```

```
## Warning: Removed 10 row(s) containing missing values (geom_path).
```

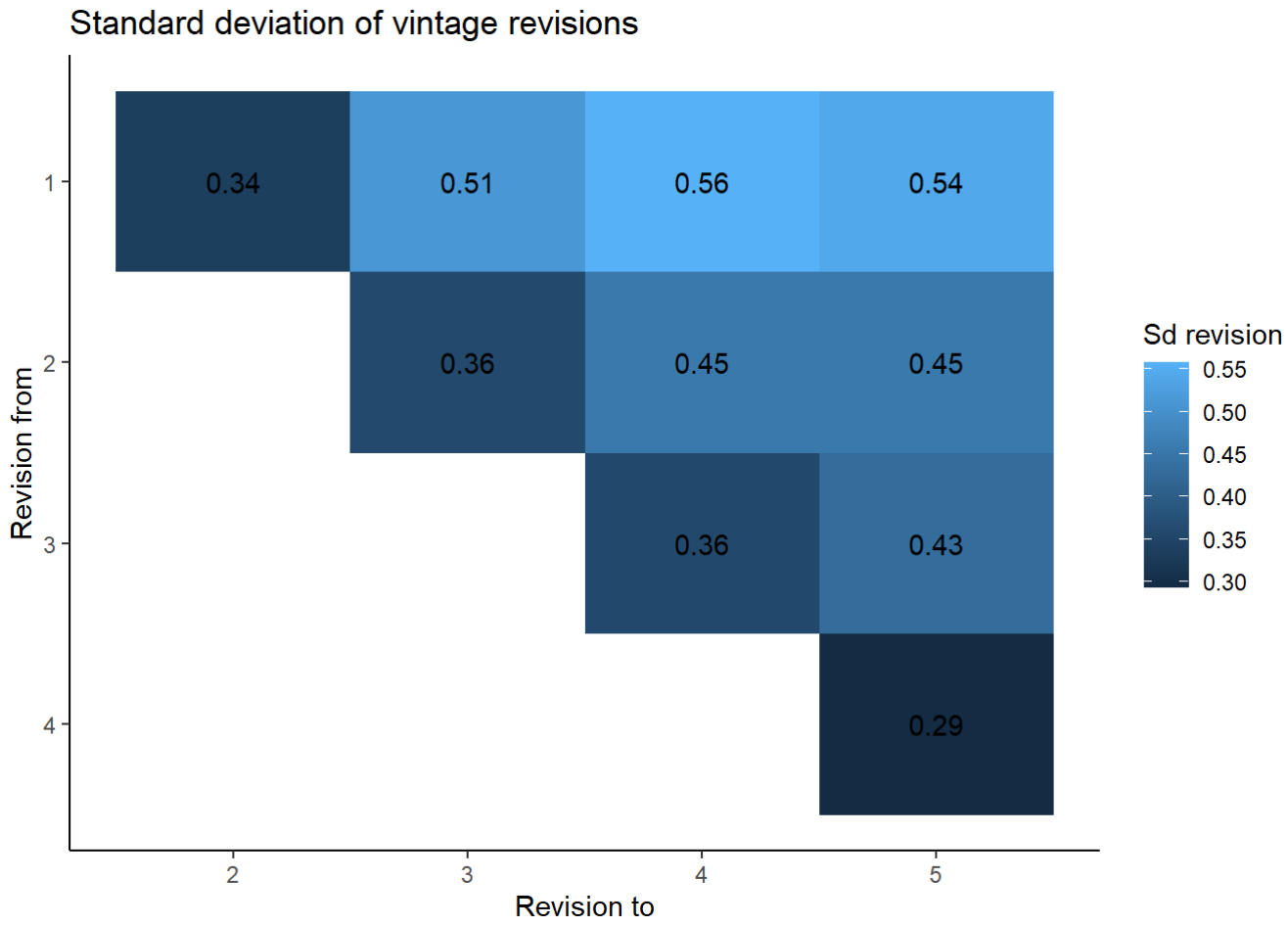


We can further obser that the mean, median and SD *all* increase with a higher revision number. This overall pattern is similar to MS, although the scales are different.

```
df_revisions_long %>%
  filter(complete.cases(.)) %>%
  group_by(Revision) %>%
  summarise(
    `Mean GDP growth` = mean(`GDP growth`),
    `Median GDP growth` = median(`GDP growth`),
    `Sd GDP growth` = sd(`GDP growth`)) %>%
  knitr::kable()
```

Revision	Mean GDP growth	Median GDP growth	Sd GDP growth
1	2.700000	2.685	2.473701
2	2.727436	2.760	2.517607
3	2.787105	2.720	2.660252
4	2.851892	2.810	2.673820
5	2.877222	2.795	2.673715

```
df_revisions_long %>%
  transmute(
    quarter=quarter,
    `Revision from` = Revision,
    `GDP growth from` = `GDP growth`
  ) %>%
  left_join(
    df_revisions_long %>%
      transmute(
        quarter=quarter,
        `Revision to` = Revision,
        `GDP growth to` = `GDP growth`
      ),
    by="quarter"
  ) %>%
  filter(as.numeric(`Revision from`) < as.numeric(`Revision to`)) %>%
  mutate(
    `Revision to estimate` = `GDP growth to` - `GDP growth from`
  ) %>%
  filter(complete.cases(.)) %>%
  mutate(
    `Revision from` = as.numeric(`Revision from`),
    `Revision to` = as.numeric(`Revision to`)
  ) %>%
  group_by(`Revision from`,`Revision to`) %>%
  summarise(`Sd revision` = sd(`Revision to estimate`)) %>%
  ggplot(aes(y=`Revision from`, x = `Revision to`)) +
  geom_tile(aes(fill =`Sd revision` )) +
  geom_text(aes(label = round(`Sd revision`, 2))) +
  scale_y_reverse() +
  ggtitle("Standard deviation of vintage revisions") +
  theme_classic()
```

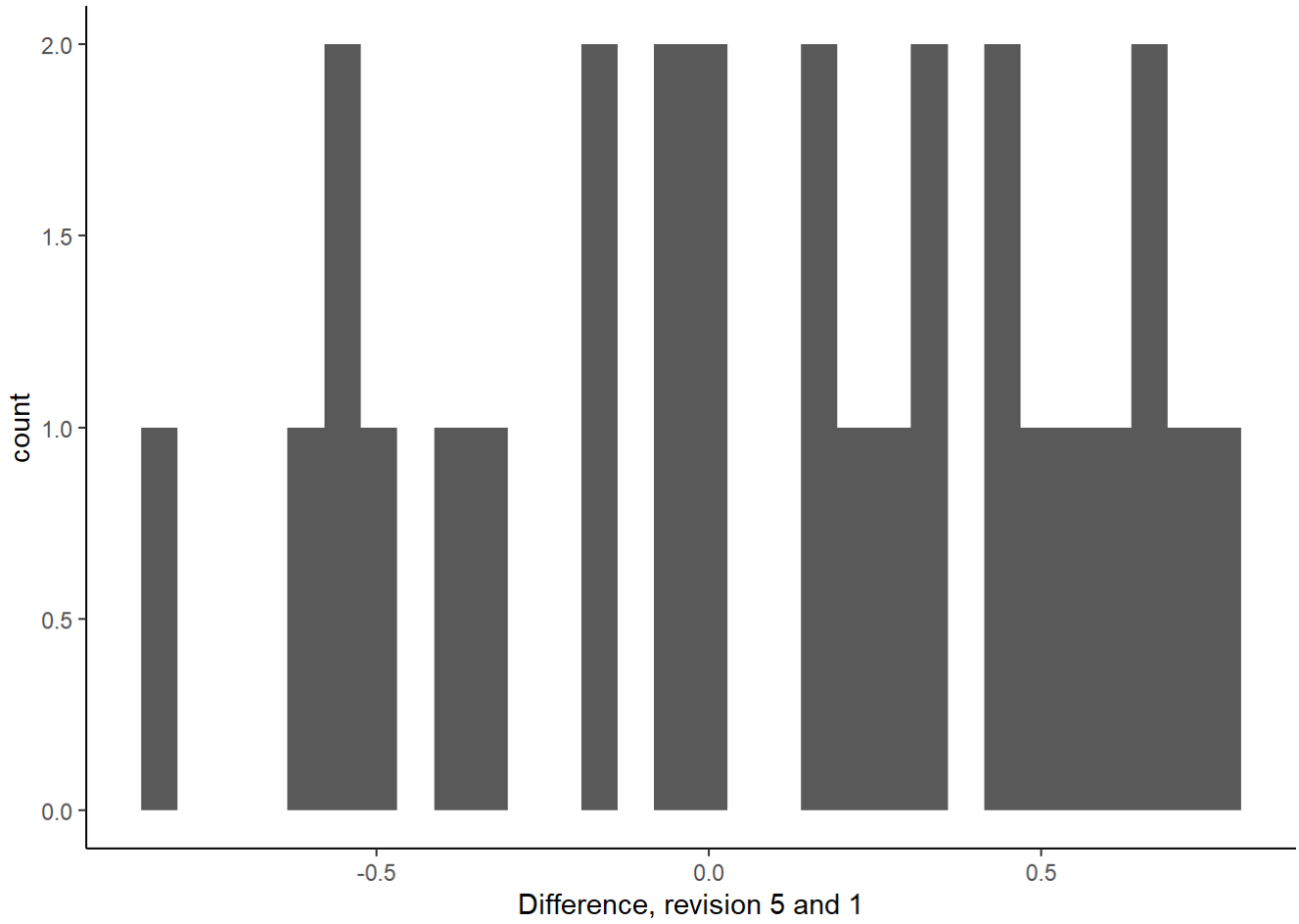


We can further inspect the difference between two revisions - here the first and fifth revision. There aren't many observations to work with. From the histogram we can see that the revision to the growth rate between the first and fifth revision is in the range a bit smaller than $\pm$ 1 percentage point.

```
diff_Y1_Y5 <-
  df_revisions_wide %>%
  select(quarter, Y1, Y5) %>%
  left_join(
    df_revisions %>%
      pivot_wider(
        id_cols = quarter,
        names_from = version,
        values_from = main_revision
      ) %>%
      select(quarter, Y1, Y5) %>%
      mutate(keep = case_when(Y1 == Y5 ~TRUE, TRUE~FALSE)) %>%
      select(quarter, keep)
  ) %>%
  filter(complete.cases(.)) %>%
  filter(keep) %>%
  transmute(diff_Y1_Y5 = Y5 - Y1)
```

```
## Joining, by = "quarter"
```

```
diff_Y1_Y5 %>%
  ggplot(aes(x = diff_Y1_Y5)) +
  geom_histogram(bins = 30) +
  xlab("Difference, revision 5 and 1") +
  theme_classic()
```



Q2

Q2A

MS finds that the variance of the GDP revisions increase with GDP-revision, and further argues that this is evidence against the “noise” hypothesis. On the Norwegian data there is a similar pattern where the variance of GDP-growth is increasing with the revision number. We could run a test of whether we from the data can conclude that the variances of GDP growth are different across vintages. Below, we conduct pair-wise tests of equality of variances using a two-sided test. Overall, there is little support in the data for rejecting the null of *equal* variances.

```
testVarTwoRevisons <-
  function(Rev1, Rev2){
    df_revisions_long %>%
    filter(Revision %in% c(Rev1, Rev2)) %$%
    var.test(`GDP growth` ~ Revision)$p.value
  }

returnSdRevision <-
  function(Rev){
    df_revisions_long %>%
      filter(Revision == Rev) %>%
      filter(complete.cases(.))  %$%
      sd(`GDP growth`)
  }

expand.grid(
    Revision1 = 1:5,
    Revision2 = 1:5) %>%
  as_tibble() %>%
  filter(Revision1 < Revision2) %>%
  arrange(Revision1, Revision2) %>%
  mutate(
    `Sd Revision 1` = purrr::map(Revision1, returnSdRevision),
    `Sd Revision 2` = purrr::map(Revision2, returnSdRevision),
    `P-val, test of equal variances`= purrr::map2(Revision1, Revision2, testVarTwoRevisons))
 %>%
  knitr::kable()
```

Revision1	Revision2	Sd Revision 1	Sd Revision 2	P-val, test of equal variances
1	2	2.47370064312456	2.51760717427096	0.912523384497416
1	3	2.47370064312456	2.66025187579565	0.654124585041081
1	4	2.47370064312456	2.67382044381551	0.633031116540818
1	5	2.47370064312456	2.67371503034487	0.634459800647036
2	3	2.51760717427096	2.66025187579565	0.736363579689763
2	4	2.51760717427096	2.67382044381551	0.713886077049888
2	5	2.51760717427096	2.67371503034487	0.714801076724225
3	4	2.66025187579565	2.67382044381551	0.974438478756583
3	5	2.66025187579565	2.67371503034487	0.973540074467494
4	5	2.67382044381551	2.67371503034487	0.998923430836917

Q2B

Comparing revision from the first to the fifth estimate, we can note that the standard deviation of the revision is $\approx .46$. If we assume revisions are normally iid, then this implies that a 95% prediction interval around a 5% first release growth rate is $[4.1, 5.9]$. This is quite a bit smaller than MS, who found that a similar interval was $[-.4, 10.6]$. A possible reason we find so much tighter intervals could be that we are using a more recent time series, and GDP measurement technology might have improved significantly.

```
sd(diff_Y1_Y5$diff_Y1_Y5) * qnorm(.975)
```

```
## [1] 0.8963958
```

Q3

There are many ways this question can be answered, so this part of the solution proposal only contains some general pointers to the issues in solving this.

First, there are many possible pairs of vintages that can be used for estimating the “news vs noise” hypotheses. Further, there is the consideration of whether to include seasonal dummies, and also whether we should account for main revisions. Below, we simply estimate all of these possible models, and collect all the regression models in a dataframe.

Note that a code as given below is *not* expected for students from Met4, as it uses programming techinques that go well beyond Met4.

```r
estimateModel <-
  function(Rev1, Rev2, include_seasons = FALSE, same_main_revision=FALSE){
    #' EstimateModel
    #'
    #' Estimates and returns a linear regression model to test news vs noise.
    #' L is the latest release, and P is the preliminary release. R = L-P.
    #'
    #' The function will determine which one of Rev1 and Rev2 is L and P.
    #'
    #' The function assumes the data frames df_revisions_wide and
    #' df_main_revisions are defined. These will be used in the regressions.
    #'
    #' @param Rev1 The benchmark revision to use in regression
    #' @param Rev2 Then revision to use as dependent variable in the regression.
    #' @param include_seasons Flags whether regression should include quarterly dummies.
    #' @param same_main_revision Flags whether the data frame should be filtered such
    #' that only observations from same main revision are used in the regression.

    form <-
        formula(
          paste0("R~", "Y",Rev2,
            ifelse(include_seasons,"+factor(lubridate::quarter(quarter))","")
            )
          )

    df_tmp <-
      df_revisions_wide %>%
      .[,c("quarter", paste0("Y",Rev1),paste0("Y",Rev2))]

    df_tmp[,"R"] <- (
      df_tmp[,paste0("Y",max(c(Rev1, Rev2)))] -
        df_tmp[,paste0("Y",min(c(Rev1, Rev2)))]
    )

    if (same_main_revision) {
      df_main_rev <-
        df_revisions %>%
        pivot_wider(
          id_cols = quarter,
          names_from = version,
          values_from = main_revision
        )
      df_main_rev <-
        df_main_rev[,c("quarter", paste0("Y",Rev1),paste0("Y",Rev2))]

      same_main_revisions <-
        as.vector(df_main_rev[,2] == df_main_rev[,3]) %>%
        replace_na(FALSE)

      df_main_rev <- df_main_rev[same_main_revisions,1]

      df_tmp %<>%
      inner_join(df_main_rev, by = "quarter")
    }

  lm(form, data = df_tmp)
  }
```

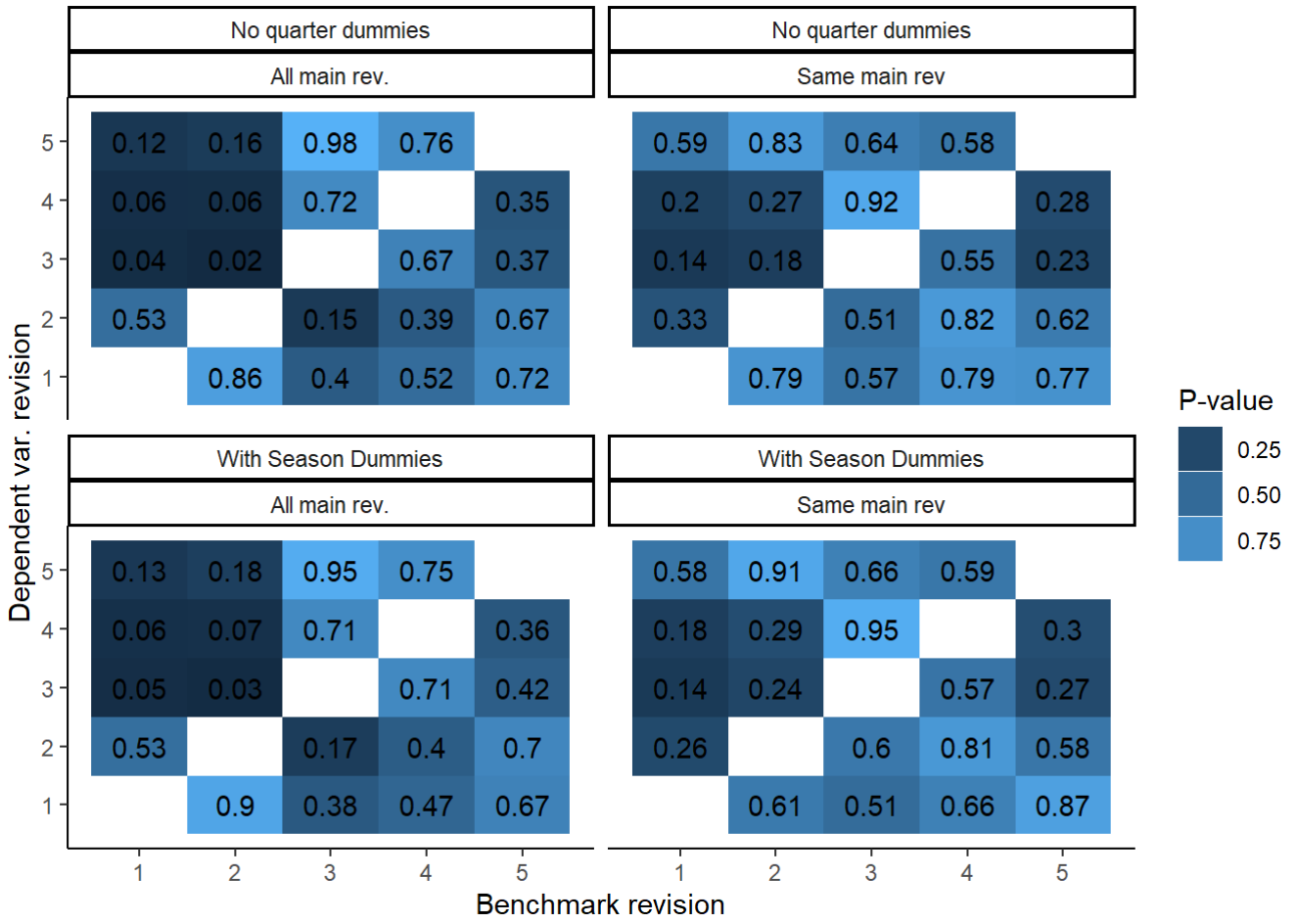
```
regressions <-
  expand.grid(
    Rev1 = 1:5,
    Rev2 = 1:5,
    include_seasons = c(TRUE,FALSE),
    same_main_revision = c(TRUE,FALSE)
  ) %>%
  as_tibble() %>%
  filter(Rev1 != Rev2) %>%
  mutate(
    regression = purrr::pmap(.,estimateModel),
    reg_summary = purrr::map(regression,summary),
    coefficients = purrr::map(reg_summary, "coefficients"),
    sigma = purrr::map_dbl(reg_summary, "sigma"),
    R2 = purrr::map_dbl(reg_summary, "r.squared"),
    Intercept = purrr::map_dbl(coefficients, ~.x[1,1]),
    Intercept_pval = purrr::map_dbl(coefficients, ~.x[1,4]),
    Y_coefficient = purrr::map_dbl(coefficients, ~.x[2,1]),
    Y_p_val = purrr::map_dbl(coefficients, ~.x[2,4])
    )
```

With all of these regressions we can summarise the results in the following broad points:

- There are surprisingly few violations of the OLS assumptions. The residuals show generally low levels of autocorrelation. There might be some influential observations.
- The seasonal dummies are generally not significant, and further does not change the coefficient on L or P in the regression.
- However, whether we include *all* observations *or* filter the dataset first such that we only include pairs from the same main revision does have an impact. Using all the data we find support for the *news* hypothesis between revision 1 and 2 and later revisions. However, this effect disappears when we filter on having the same main revision.

Overall, the findings are similar to the discussion in MTF. They found a strong news signal between the first and fifth estimate of GDP growth in Norway, howver, they also note in the discussion that several statistics bureaus objected to the comparison of data assembled with very different methodologies.

```
regressions %>%
  mutate(
    include_seasons =
      case_when(
        include_seasons~"With Season Dummies",
        TRUE~"No quarter dummies"),
    same_main_revision =
      case_when(
        same_main_revision~"Same main rev",
        TRUE~"All main rev.")) %>%
  ggplot(aes(x=Rev1, y=Rev2)) +
  geom_tile(aes(fill = Y_p_val)) +
  geom_text(aes(label = round(Y_p_val, 2))) +
  xlab("Benchmark revision") +
  ylab("Dependent var. revision") +
  facet_wrap(~factor(include_seasons) + factor(same_main_revision)) +
  guides(fill=guide_legend(title="P-value")) +
  theme_classic()
```



Q4

We first estimate the AR(2) model as requested. Note, here we are using data straight from SSB, and we omit the last four observations.

Further, we calculate the variance of the revision from $Y^4 - Y^1$ and $Y^4 - Y^2$.

```
reg_ar2 <-
  df_latest_release %>%
  head(-4) %>%
  lm(Y ~ lag_1_Y + lag_2_Y, data = .)


diff_Y1_Y4 <-
  df_revisions_wide %>%
  select(quarter, Y1, Y4) %>%
  left_join(
    df_revisions %>%
      pivot_wider(
        id_cols = quarter,
        names_from = version,
        values_from = main_revision
      ) %>%
      select(quarter, Y1, Y4) %>%
      mutate(keep = case_when(Y1 == Y4 ~TRUE, TRUE~FALSE)) %>%
      select(quarter, keep)
  ) %>%
  filter(complete.cases(.)) %>%
  filter(keep) %>%
  transmute(diff_Y1_Y4 = Y4 - Y1)
```

```
## Joining, by = "quarter"
```

```
diff_Y2_Y4 <-
  df_revisions_wide %>%
  select(quarter, Y2, Y4) %>%
  left_join(
    df_revisions %>%
      pivot_wider(
        id_cols = quarter,
        names_from = version,
        values_from = main_revision
      ) %>%
      select(quarter, Y2, Y4) %>%
      mutate(keep = case_when(Y2 == Y4 ~TRUE, TRUE~FALSE)) %>%
      select(quarter, keep)
  ) %>%
  filter(complete.cases(.)) %>%
  filter(keep) %>%
  transmute(diff_Y2_Y4 = Y4 - Y2)
```

```
## Joining, by = "quarter"
```

```
summary_ar2 <- summary(reg_ar2)
var_Y1_Y4 <- var(diff_Y1_Y4$diff_Y1_Y4)
var_Y2_Y4 <- var(diff_Y2_Y4$diff_Y2_Y4)
```

Further, we create a prediction using the latest data:

```
prediction <-
  predict(
    reg_ar2,
    newdata =  tail(df_latest_release,1),
    interval = 'prediction')
prediction %>%
  knitr::kable()
```

	fit	lwr	upr
170	1.739986	-2.838812	6.318784

We can now find the variance of the prediction - where we only account for the uncertainty of the “residual” in the next period:

```
var_pred <- ((prediction[3] - prediction[2]) / (2*qnorm(.975)))^2
var_pred
```

```
## [1] 5.457664
```

The regression model we estimated is $\alpha + \rho_1 Y_{t-1} + \rho_2 Y_{t-2} + \epsilon$. Let’s ignore the uncertainty associated with the parameters we have estimated. For the prediction, we do not observe Y_{t-1}, Y_{t-2}, but instead $Y^1_{t-1} + \rho_2 Y^4_{t-2}$. Let’s further assume that $Y_{t-1} = Y^1_{t-1} + \nu_{1,4}$ and $Y_{t-2} = Y^2_{t-2} + \nu_{2,4}$ - i.e. that the two newest data points are equal to the true values + some noise with a *known* variance. Assuming the noise is normally IID, the variance of the prediction is

$$
\begin{aligned}
&Var[\hat{Y}] \\
&= Var\left[\alpha + \rho_2 Y^1 + \rho_1 Y^2 + \epsilon\right] \\
&= Var\left[\rho_1 Y^1\right] + Var\left[\rho_2 Y^2\right] + Var\left[\epsilon\right] \\
&= \rho_1^2 Var\left[Y^1\right] + \rho_2^2 Var\left[Y^2\right] + Var\left[\epsilon\right] \\
&= \rho_1^2 Var[\nu_{1,4}] + \rho_2^2 Var[\nu_{2,4}] + Var\left[\epsilon\right]
\end{aligned}
$$

We can find the variance contribution from the two terms related to the uncertainty of the first and second revision of GDP:

```
var_me <-
  (
    var_Y1_Y4 * summary_ar2$coefficients[2,1]^2 +
    var_Y2_Y4 * summary_ar2$coefficients[3,1]^2
  )
print(var_me)
```

```
## [1] 0.03034717
```

This is actually a fairly small number! We can then finally print out the variance of the prediciton, with and without the accounting for the uncertainty of the latest two data points:

```
print(var_pred)
```

```
## [1] 5.457664
```

```
print(var_pred + var_me)
```

```
## [1] 5.488011
```

As we can see, the contribution from accounting for the uncertainty of the last two data points is small. This is partly caused by the coefficients of the model reducing the contribution from these terms. Further, the AR(2)-model is generally not very good, and hence the uncertainty of the predictions is large. Therefore, relative to the existing uncertainty, the measurement error does not change the variance of the prediction very much (If we had a better time series model, this point might change!).