Homework 4: Neo Lee

Introduction to Time Series, Fall 2023

Due Friday November 3 at 9pm

The total number of points possible for this homework is 34. The number of points for each question is written below, and questions marked as "bonus" are optional. Submit the **knitted html file** from this Rmd to Gradescope.

If you collaborated with anybody for this homework, put their names here:

Backshift commuting

1. (1 pt) Let B denote the backshift operator. Given any integers $k, \ell \geq 0$, explain why $B^k B^\ell = B^\ell B^k$.

SOLUTION GOES HERE

Consider arbitrary x_t. Then,

$$B^k B^\ell x_t = B^k x_{t-\ell} = x_{t-\ell-k} = B^\ell x_{t-k} = B^\ell B^k x_t.$$

2. (2 pts) Using Q1, if $\phi_1, \dots, \phi_k$ and $\varphi_1, \dots, \varphi_\ell$ are any coefficients, show that

$$(1 + \phi_1 B + \cdots + \phi_k B^k)(1 + \varphi_1 B + \cdots + \varphi_\ell B^\ell) = (1 + \varphi_1 B + \cdots + \varphi_\ell B^\ell)(1 + \phi_1 B + \cdots + \phi_k B^k)$$

SOLUTION GOES HERE

$$\begin{aligned}
(1 + \phi_1 B + \cdots + \phi_k B^k)(1 + \varphi_1 B + \cdots + \varphi_\ell B^\ell) &= \left(\sum_{i=0}^{k} \phi_i B^i\right)\left(\sum_{j=0}^{\ell} \varphi_j B^j\right) \qquad (\text{let } \phi_0 = \varphi_0 = 1) \\
&= \sum_{i=0}^{k} \sum_{j=0}^{\ell} \phi_i B^i \cdot \varphi_j B^j \\
&= \sum_{i=0}^{k} \sum_{j=0}^{\ell} \phi_i \varphi_j \cdot B^i B^j \\
&= \sum_{i=0}^{k} \sum_{j=0}^{\ell} \varphi_j \phi_i \cdot B^j B^i \\
&= \sum_{i=0}^{k} \sum_{j=0}^{\ell} \varphi_j B^j \cdot \phi_i B^i \\
&= \sum_{i=0}^{\ell} \sum_{j=0}^{k} \varphi_j B^j \cdot \phi_i B^i \\
&= (1 + \varphi_1 B + \cdots + \varphi_\ell B^\ell)(1 + \phi_1 B + \cdots + \phi_k B^k)
\end{aligned}$$

3. (2 pts) Verify the result in Q2 with a small code example.

```r
# CODE GOES HERE

k = 2
l = 3
x = rnorm(10)
phi = rnorm(k)
phi = c(1, phi)
varphi = rnorm(l)
varphi = c(1, varphi)

l_first = 0
indices = length(x):(length(x) - l)
for (i in 0:k) {
  l_first = l_first + phi[i+1] * sum(varphi * x[indices - i])
}

k_first = 0
indices = length(x):(length(x) - k)
for (i in 0:l) {
  k_first = k_first + varphi[i+1] * sum(phi * x[indices - i])
}

l_first - k_first < 1e-10 # floating point error
```

```
## [1] TRUE
```

4. (3 pts) Using Q2, show that we can write a SARIMA model equivalently as

$$\phi(B)\Phi(B^s)\nabla^d\nabla_s^D x_t = \theta(B)\Theta(B^s)w_t$$

and

$$\Phi(B^s)\phi(B)\nabla_s^D\nabla^d x_t = \Theta(B^s)\theta(B)w_t.$$

SOLUTION GOES HERE

Notice

$$\nabla^d\nabla_s^D = (1-B)^d(1-B^s)^D = (1-B^s)^D(1-B)^d$$

where we apply Q2 iteratively on $(1+\phi_1 B)$ where $\phi_1 = -1$ and $(1+\varphi_1 B + \cdots + \phi_s B^s)$ where $\varphi_s = -1$ and 0 else. Therefore,

$$\nabla^d\nabla_s^D = \nabla_s^D\nabla^d.$$

Similarly,

$$\begin{aligned}\phi(B)\Phi(B^s) &= (1+\phi_1 B + \cdots + \phi_p B^p)(1+\Phi_s B^s + \cdots + \Phi_{Ps}B^{Ps}) \\ &= (1+\Phi_s B^s + \cdots + \Phi_{Ps}B^{Ps})(1+\phi_1 B + \cdots + \phi_p B^p)\end{aligned}$$

by applying Q2 where $\varphi_i = \Phi_i$ for $i = s, 2s, \ldots, Ps$ and 0 else. Therefore,

$$\phi(B)\Phi(B^s) = \Phi(B^s)\phi(B).$$

Similarly,

$$\begin{aligned}\theta(B)\Theta(B^s) &= (1-\theta_1 B - \cdots - \theta_q B^q)(1-\Theta_s B^s - \cdots - \Theta_{Qs}B^{Qs}) \\ &= (1-\Theta_s B^s - \cdots - \Theta_{Qs}B^{Qs})(1-\theta_1 B - \cdots - \theta_q B^q),\end{aligned}$$

by applying Q2 where $\phi_i = -\theta_i$ and $\varphi_i = -\Theta_i$ for $i = s, \ldots, Qs$ and 0 else. Therefore,

$$\theta(B)\Theta(B^s) = \Theta(B^s)\theta(B).$$

Hence, we achieved the desired equivalence.

Long-range ARIMA

5. (1 pt) Let $\nabla = (1 - B)$ denote the difference operator. Suppose that $\nabla x_t = 0$ for all t. Prove that x_t must be a constant sequence.

SOLUTION GOES HERE

$$\nabla x_t = 0 \implies x_t - x_{t-1} = 0 \implies x_t = x_{t-1}.$$

This is true for all t. Therefore, $x_t = x_{t-1} = x_{t-2} = \cdots = x_0$, which means x_t is a constant sequence.

6. (2 pts) Suppose that $\nabla x_t = u$ for all t, where u is an arbitrary constant. Prove that x_t must be a linear function of t, of the form $x_t = a + bt$.

SOLUTION GOES HERE

$$\forall t, \nabla x_t = u \implies x_t = x_{t-1} + u = x_{t-2} + u + u = x_{t-3} + u + u + u = \cdots = x_0 + tu,$$

which is in the desired form of $x_t = a + bt$.

7. (3 pts) Suppose that $\nabla x_t = u + vt$ for all t, where u, v are again arbitrary constants. Prove that x_t must be a quadratic function of t, of the form $x_t = a + bt + ct^2$.

SOLUTION GOES HERE

$$\forall t, \nabla x_t = u + vt \implies \forall t, x_t = u + vt + x_{t-1}.$$

$$\begin{aligned}
x_t &= u + vt + x_{t-1} \\
&= u + vt + u + v(t-1) + x_{t-2} \\
&= u + vt + u + v(t-1) + u + v(t-2) + x_{t-3} \\
&= u + vt + u + v(t-1) + u + v(t-2) + \cdots + u + v(t-(t-1)) + x_0 \\
&= ut + v\left(t^2 - (1 + \cdots + (t-1))\right) + x_0 \\
&= ut + v\left(t^2 - \frac{t(t-1)}{2}\right) + x_0 \\
&= ut + v\left(\frac{t^2 + t}{2}\right) + x_0 \\
&= x_0 + \left(u + \frac{v}{2}\right)t + \frac{v}{2}t^2,
\end{aligned}$$

which is in the desired form of $x_t = a + bt + ct^2$.

8. (1 pt) Using Q7, prove that if $\nabla^2 x_t = u$ for all t, where u is a constant, then x_t must be a quadratic function of t.

SOLUTION GOES HERE

Notice $\nabla^2 x_t = \nabla(\nabla x_t)$, where (∇x_t) is itself another sequence. Then by Q6,

$$\nabla(\nabla x_t) = u \implies \nabla x_t = a + bt,$$

and by Q7,

$$\nabla x_t = u + vt \implies x_t = a + bt + ct^2,$$

which is a quadratic function of t. (abuse of symbols, u, a, b are not necessarily the same in equation 1 and 2)

9. (4 pts) Consider an ARMA(1,1) model:

$$(1 - \phi B)x_t = (1 + \theta B)w_t.$$

Suppose our estimates pass the "unit root test", $|\hat{\phi}|, |\hat{\theta}| < 1$, which will be assumed implicitly henceforth. Unravel the forecast iteration described in lecture to show that $\hat{x}_{t+h|t}$ approaches zero as $h \to \infty$.

SOLUTION GOES HERE

$$(1-\phi B)x_t = (1+\theta B)w_t \implies x_t - \phi x_{t-1} = w_t + \theta w_{t-1} \implies x_t = \phi x_{t-1} + w_t + \theta w_{t-1}.$$

Then,

$$\begin{aligned}
\hat{x}_{t+1|t} &= \hat{\phi} x_t + w_{t+1} + \hat{\theta}\hat{w}_t = \hat{\phi} x_t + \hat{\theta}\hat{w}_t \qquad (w_{t+1}=0) \\
\hat{x}_{t+2|t} &= \hat{\phi} \hat{x}_{t+1|t} + w_{t+2} + \hat{\theta} w_{t+1} = \hat{\phi}\left(\hat{\phi} x_t + \hat{\theta}\hat{w}_t\right) \qquad (w_{t+2}=w_{t+1}=0) \\
\hat{x}_{t+3|t} &= \hat{\phi} \hat{x}_{t+2|t} + w_{t+3} + \hat{\theta} w_{t+2} = \hat{\phi}\left(\hat{\phi}\left(\hat{\phi} x_t + \hat{\theta}\hat{w}_t\right)\right) \qquad (w_{t+3}=w_{t+2}=0) \\
&\vdots \\
\lim_{h\to\infty} \hat{x}_{t+h|t} &= \lim_{h\to\infty} \hat{\phi}^{h-1}\left(\hat{\phi} x_t + \hat{\theta}\hat{w}_t\right) = 0 \qquad (\because |\hat{\phi}|<1 \text{ and } \hat{\phi}x_t + \hat{\theta}w_t \in \mathbb{R})
\end{aligned}$$

10. (2 pts) Consider an ARMA(1,1) model, with intercept:

$$(1-\phi B)x_t = c + (1+\theta B)w_t.$$

Unravel the forecast iteration to show that $\hat{x}_{t+h|t}$ approaches a nonzero constant as $h \to \infty$.

SOLUTION GOES HERE

$$(1-\phi B)x_t = c + (1+\theta B)w_t \implies x_t = \phi x_{t-1} + w_t + \theta w_{t-1} + c.$$

Then,

$$\begin{aligned}
\hat{x}_{t+1|t} &= \hat{\phi} x_t + w_{t+1} + \hat{\theta}\hat{w}_t + c = \hat{\phi} x_t + \hat{\theta}\hat{w}_t + c \qquad (w_{t+1}=0) \\
\hat{x}_{t+2|t} &= \hat{\phi} \hat{x}_{t+1|t} + w_{t+2} + \hat{\theta} w_{t+1} + c = \hat{\phi}\left(\hat{\phi} x_t + \hat{\theta}\hat{w}_t + c\right) + c \qquad (w_{t+2}=w_{t+1}=0) \\
\hat{x}_{t+3|t} &= \hat{\phi} \hat{x}_{t+2|t} + w_{t+3} + \hat{\theta} w_{t+2} + c = \hat{\phi}\left(\hat{\phi}\left(\hat{\phi} x_t + \hat{\theta}\hat{w}_t + c\right) + c\right) + c \qquad (w_{t+3}=w_{t+2}=0) \\
&\vdots \\
\hat{x}_{t+h|t} &= \hat{\phi}^h x_t + \hat{\phi}^{h-1}\hat{\theta}\hat{w}_t + c \cdot \sum_{k=0}^{h-1} \hat{\phi}^k \\
\lim_{h\to\infty} \hat{x}_{t+h|t} &= \lim_{h\to\infty} \hat{\phi}^h x_t + \hat{\phi}^{h-1}\hat{\theta}\hat{w}_t + c \cdot \sum_{k=0}^{h-1} \hat{\phi}^k = c \cdot \lim_{h\to\infty} \sum_{k=0}^{h-1} \hat{\phi}^k = C,
\end{aligned}$$

because $|\hat{\phi}| < 1$ and $x_t, \hat{\theta}w_t \in \mathbb{R}$, and $\lim_{h\to\infty} \sum_{k=0}^{h-1} \hat{\phi}^k$ converges to some non-zero real number due to $|\hat{\phi}| < 1$ and $\hat{\phi}^0 = 1$.

11. (Bonus) Now consider the extension to ARIMA$(1, d, 1)$:

$$(1-\phi B)\nabla^d x_t = c + (1+\theta B)w_t.$$

Use Q5–Q10 to argue the following:

- If $c = 0$ and $d = 1$, then $\hat{x}_{t+h|t}$ approaches a constant as $h \to \infty$.
- If $c = 0$ and $d = 2$, then $\hat{x}_{t+h|t}$ approaches a linear trend as $h \to \infty$.
- If $c \neq 0$ and $d = 1$, then $\hat{x}_{t+h|t}$ approaches a linear trend as $h \to \infty$.
- If $c \neq 0$ and $d = 2$, then $\hat{x}_{t+h|t}$ approaches a quadratic trend as $h \to \infty$.

SOLUTION GOES HERE

$\underline{c=0, d=1}$: Let $y_t = \nabla x_t$, then

$$(1-\phi B)\nabla x_t = (1+\theta B)w_t \Leftrightarrow (1-\phi B)y_t = (1+\theta B)w_t.$$

By Q9, $\hat{y}_{t+h|t} = 0$ as $h \to \infty$, thus $\nabla \hat{x}_{t+h|t} = 0$ as $h \to \infty$. By Q5, $\hat{x}_{t+h|t}$ approaches a constant sequence as $h \to \infty$.

$\underline{c=0, d=2}$: Let $y_t = \nabla^2 x_t$, then

$$(1-\phi B)\nabla^2 x_t = (1+\theta B)w_t \Leftrightarrow (1-\phi B)y_t = (1+\theta B)w_t.$$

By Q9, $\hat{y}_{t+h|t} = 0$ as $h \to \infty$, thus $\nabla^2 \hat{x}_{t+h|t} = 0$ as $h \to \infty$. Notice $\nabla^2 \hat{x}_{t+h|t} = \nabla(\nabla \hat{x}_{t+h|t})$, then by Q5, $\nabla \hat{x}_{t+h|t}$ approaches a constant sequence as $h \to \infty$. Further, by Q6, $\hat{x}_{t+h|t}$ approaches a linear function of t as $h \to \infty$.

$\underline{c \neq 0, d=1}$: Let $y_t = \nabla x_t$, then

$$(1-\phi B)\nabla x_t = c + (1+\theta B)w_t \Leftrightarrow (1-\phi B)y_t = c + (1+\theta B)w_t.$$

By Q10, $\hat{y}_{t+h|t}$ approaches a non-zero constant as $h \to \infty$, thus $\nabla \hat{x}_{t+h|t}$ approaches a none-zero constant as $h \to \infty$. By Q6, $\hat{x}_{t+h|t}$ approaches a linear function of t as $h \to \infty$.

$\underline{c \neq 0, d=2}$: Let $y_t = \nabla^2 x_t$, then

$$(1-\phi B)\nabla x_t = c + (1+\theta B)w_t \Leftrightarrow (1-\phi B)y_t = c + (1+\theta B)w_t.$$

By Q10, $\hat{y}_{t+h|t}$ approaches a non-zero constant as $h \to \infty$, thus $\nabla^2 \hat{x}_{t+h|t}$ approaches a none-zero constant as $h \to \infty$. By Q8, $\hat{x}_{t+h|t}$ approaches a quadratic function of t as $h \to \infty$.

Time series CV

When we learned time series cross-validation in lecture (weeks 3-4, "Linear regression and prediction"), we implemented it "manually", by writing a loop in R to iterate over time, rebuild models, and so on. The `fable` package in R does it differently. It relies on data being stored in a class that is known as a `tsibble`, which is like a special data frame for time series. You can then use a function called `stretch_tsibble()` in order to "prepare it" for time series cross-validation. Take a look at what it does with this example:

```
library(tidyverse)
library(fpp3)

dat = tsibble(date = as.Date("2023-10-01") + 0:9,
              value = 1:10 + rnorm(10, sd = 0.25),
              index = date)
dat
```

```
## # A tsibble: 10 x 2 [1D]
##    date       value
##    <date>     <dbl>
##  1 2023-10-01  1.31
##  2 2023-10-02  1.61
##  3 2023-10-03  2.68
##  4 2023-10-04  3.73
##  5 2023-10-05  4.85
##  6 2023-10-06  5.35
##  7 2023-10-07  7.03
##  8 2023-10-08  8.08
##  9 2023-10-09  8.74
## 10 2023-10-10 10.0
```

```
dat_stretched = dat|> stretch_tsibble(.init = 3)
dat_stretched
```

```
## # A tsibble: 52 x 3 [1D]
## # Key:       .id [8]
##    date       value   .id
##    <date>     <dbl> <int>
##  1 2023-10-01  1.31     1
##  2 2023-10-02  1.61     1
##  3 2023-10-03  2.68     1
##  4 2023-10-01  1.31     2
##  5 2023-10-02  1.61     2
##  6 2023-10-03  2.68     2
##  7 2023-10-04  3.73     2
##  8 2023-10-01  1.31     3
##  9 2023-10-02  1.61     3
## 10 2023-10-03  2.68     3
## # i 42 more rows
```

What this does is it takes the first 3 entries of the time series and assigns them `.id = 1`. Then it appends the first 4 entries of the time series and assigns them `.id = 2`. Then it appends the first 5 entries of the time series and assigns them `.id = 3`, and so on. Downstream, when we go to fit a forecast model with fable, it (by default) will fit a separate model to the data in each level of the `.id` column. And by making forecasts at a (say) horizon `h = 1`, these are actually precisely the 1-step ahead forecasts that we would generate in time series CV:

```
dat_fc = dat_stretched |>
  model(RW = RW(value ~ drift())) |>
  forecast(h = 1)
dat_fc
```

```
## # A fable: 8 x 5 [1D]
## # Key:     .id, .model [8]
##     .id .model date              value .mean
##   <int> <chr>  <date>           <dist> <dbl>
## 1     1 RW     2023-10-04 N(3.4, 0.44)  3.36
## 2     2 RW     2023-10-05 N(4.5, 0.26)  4.53
## 3     3 RW     2023-10-06 N(5.7, 0.19)  5.73
## 4     4 RW     2023-10-07 N(6.2, 0.17)  6.16
## 5     5 RW     2023-10-08   N(8, 0.28)  7.99
## 6     6 RW     2023-10-09   N(9, 0.23)  9.04
## 7     7 RW     2023-10-10 N(9.7, 0.21)  9.66
## 8     8 RW     2023-10-11   N(11, 0.2) 11.0
```

The `.mean` column give us the point forecast. To evaluate these, we could join the original data `dat` to the point forecasts in `dat_fc`, and then align by the `date` column, and compute whatever metrics we wanted. However, there is also a handy function to do all of this for us, called `accuracy()`. This computes a bunch of common metrics, and here we just pull out the `MAE` column:

```
accuracy(dat_fc, dat) |> select(.model, MAE)
```

```
## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 1 observation is missing at 2023-10-11
```

```
## # A tibble: 1 x 2
##   .model   MAE
##   <chr>  <dbl>
```

```
## 1 RW      0.389
```

Now for the questions.

12. (3 pts) A clear advantage to the above workflow is convenience: we have to write less code. A disadvantage is that it can be inefficient, and in particular, memory inefficient. To see this, consider using this to do time series CV on a sequence with n observations and burn-in time t_0. We store this as a `tsibble`, call it `x`, with `n` rows, and then we run `stretch_tsibble(x, .init = t0)`. How many rows does the output have? Derive the answer mathematically (as an explicit formula involving n, t_0), and then verify it with a couple of code examples.

SOLUTION GOES HERE

$$\text{rows} = t_0 + (t_0 + 1) + (t_0 + 2) + \cdots + n = \frac{(t_0 + n)(n - t_0 + 1)}{2}.$$

```r
# CODE GOES HERE

library(tidyverse)
library(fpp3)

n = c(100, 249, 320, 408)
t0 = c(29, 40, 269, 397)

for (i in 1:4) {
  q12 = tsibble(
    date = as.Date("2023-10-01") + 0:(n[i]-1),
    value = 1:n[i] + rnorm(n[i], sd = 0.25),
    index = date
  )

  q12_stretched = q12 |> stretch_tsibble(.init = t0[i])
  rows = nrow(q12_stretched)
  print(rows == (t0[i] + n[i]) * (n[i] - t0[i] + 1) / 2)
}
```

```
## [1] TRUE
## [1] TRUE
## [1] TRUE
## [1] TRUE
```

13. (4 pts) Show that the MAE result for the random walk forecasts produced above, on the data in `dat`, matches the MAE from a manual implementation of time series CV with the same forecaster. (Your manual implementation can build off the code from the regression lecture, and/or from previous homeworks.)

```r
# CODE GOES HERE

n = nrow(dat)
t0 = 3

manual_forecast = rep(NA, n - t0)
for (t in (t0 + 1):n) {
  train = dat[1:(t - 1), ]
  model = model(train, RW = RW(value ~ drift()))
  manual_forecast[t - t0] = forecast(model, h = 1)$.mean
}
```

```
manual_mae = mean(abs(manual_forecast - dat[(t0 + 1):n, ]$value))
auto_mae = accuracy(dat_fc, dat) |>
  select(.model, MAE) |>
  pull(MAE)
```

```
## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 1 observation is missing at 2023-10-11
```

```
auto_mae == manual_mae
```

```
## [1] TRUE
```

14. (6 pts) Consider the `leisure` data set from the HA book, which the code excerpt below (taken from the ARIMA lecture) prepares for us. Use time series CV, implemented using `stretch_tsibble()`, `model()`, and `forecast()`, as described above, to evaluate the MAE of the following four models:

- $\text{ARIMA}(2, 1, 0)$
- $\text{ARIMA}(0, 1, 2)$
- $\text{ARIMA}(2, 1, 0)(1, 1, 0)_{12}$
- $\text{ARIMA}(0, 1, 2)(0, 1, 1)_{12}$

The last models are motivated by the exploratory analysis done in lecture. The first two remove the seasonal component, which should not be very good (since there is clear seasonality in the data). For each model you should use a burn-in period of length 50 (i.e., set `.init = 50` in the call to `stretch_tsibble()`). A key difference in how you implement time series CV to the above examples: you should consider 1-step, 2-step, all the way through 12-step ahead forecasts. But do not worry! This can be handled with an appropriate call to `forecast()`. For each model, calculate the MAE by averaging over all forecast horizons (1 through 12). Report the results and rank the models by their MAE.

```
leisure = us_employment |>
  filter(Title == "Leisure and Hospitality", year(Month) > 2000) |>
  mutate(Employed = Employed/1000) |>
  select(Month, Employed)

# CODE GOES HERE
n = nrow(leisure)
t0 = 50

leisure_stretched = leisure |> stretch_tsibble(.init = 50)

models = leisure_stretched |>
  model(
    ARIMA210 = ARIMA(Employed ~ 0 + pdq(2, 1, 0) + PDQ(0, 0, 0)),
    ARIMA012 = ARIMA(Employed ~ 0 + pdq(0, 1, 2) + PDQ(0, 0, 0)),
    ARIMA210110 = ARIMA(Employed ~ 0 + pdq(2, 1, 0) + PDQ(1, 1, 0, 12)),
    ARIMA012011 = ARIMA(Employed ~ 0 + pdq(0, 1, 2) + PDQ(0, 1, 1, 12))
  )
leisure_fc = models |> forecast(h = 12)

mae = accuracy(leisure_fc, leisure) |> select(.model, MAE) |> arrange(MAE)
```

```
## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 12 observations are missing between 2019 Oct and 2020 Sep
```

```
print(mae)
```

```
## # A tibble: 4 x 2
```

```
##   .model        MAE
##   <chr>       <dbl>
## 1 ARIMA210110 0.102
## 2 ARIMA012011 0.112
## 3 ARIMA012    0.593
## 4 ARIMA210    0.665
```

15. (Bonus) Break down the MAE for the forecasts made in Q14 by forecast horizon. That is, for each $h = 1, \dots, 12$, calculate the MAE of the h-step ahead forecasts made by each model. Make a plot with the horizon h on the x-axis and MAE on the y-axis, and compare in particular the models ARIMA$(2,1,0)(1,1,0)_{12}$ and ARIMA$(0,1,2)(0,1,1)_{12}$. Do you see anything interesting happening here in the comparison between their MAE as we vary the horizon h?

```
# CODE GOES HERE

mae_mat = matrix(0, 12, 4)
model_names = c("ARIMA210", "ARIMA012", "ARIMA210110", "ARIMA012011")
colnames(mae_mat) = model_names

max_id = max(leisure_fc[".id"])
for (name in model_names) {
  leisure_fc_model = leisure_fc |> filter(.model == name)
  for (h in 1:12) {
    forecasts = leisure_fc[1, ]
    for (id in 1:max_id) {
      id_forecasts = leisure_fc_model |>
        filter(.id == id)
      forecasts[id, ] = id_forecasts[h, ]
    }
    mae_mat[h, name] = accuracy(forecasts, leisure) |> pull(MAE)
  }
}
```

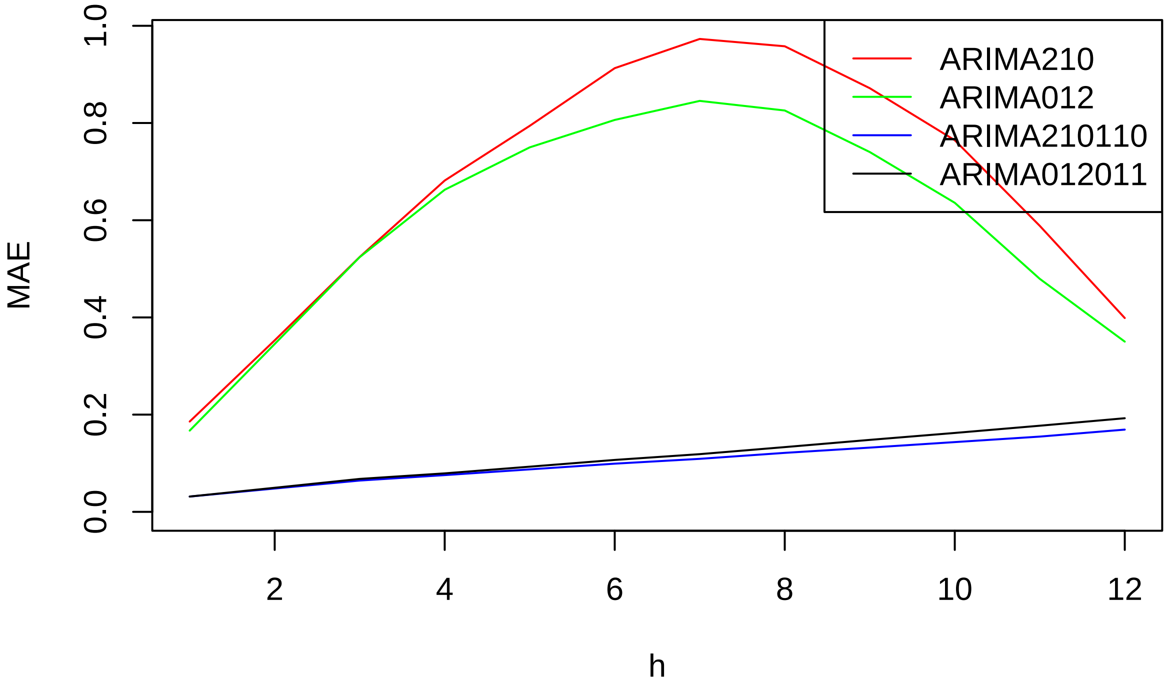
```
## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 1 observation is missing at 2019 Oct

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 2 observations are missing between 2019 Oct and 2019 Nov

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 3 observations are missing between 2019 Oct and 2019 Dec

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 4 observations are missing between 2019 Oct and 2020 Jan

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 5 observations are missing between 2019 Oct and 2020 Feb

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 6 observations are missing between 2019 Oct and 2020 Mar

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 7 observations are missing between 2019 Oct and 2020 Apr

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 8 observations are missing between 2019 Oct and 2020 May

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 9 observations are missing between 2019 Oct and 2020 Jun
```

```
## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 10 observations are missing between 2019 Oct and 2020 Jul

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 11 observations are missing between 2019 Oct and 2020 Aug

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 12 observations are missing between 2019 Oct and 2020 Sep

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 1 observation is missing at 2019 Oct

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 2 observations are missing between 2019 Oct and 2019 Nov

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 3 observations are missing between 2019 Oct and 2019 Dec

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 4 observations are missing between 2019 Oct and 2020 Jan

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 5 observations are missing between 2019 Oct and 2020 Feb

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 6 observations are missing between 2019 Oct and 2020 Mar

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 7 observations are missing between 2019 Oct and 2020 Apr

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 8 observations are missing between 2019 Oct and 2020 May

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 9 observations are missing between 2019 Oct and 2020 Jun

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 10 observations are missing between 2019 Oct and 2020 Jul

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 11 observations are missing between 2019 Oct and 2020 Aug

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 12 observations are missing between 2019 Oct and 2020 Sep

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 1 observation is missing at 2019 Oct

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 2 observations are missing between 2019 Oct and 2019 Nov

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 3 observations are missing between 2019 Oct and 2019 Dec

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 4 observations are missing between 2019 Oct and 2020 Jan

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 5 observations are missing between 2019 Oct and 2020 Feb

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 6 observations are missing between 2019 Oct and 2020 Mar
```

```
## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 7 observations are missing between 2019 Oct and 2020 Apr

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 8 observations are missing between 2019 Oct and 2020 May

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 9 observations are missing between 2019 Oct and 2020 Jun

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 10 observations are missing between 2019 Oct and 2020 Jul

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 11 observations are missing between 2019 Oct and 2020 Aug

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 12 observations are missing between 2019 Oct and 2020 Sep

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 1 observation is missing at 2019 Oct

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 2 observations are missing between 2019 Oct and 2019 Nov

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 3 observations are missing between 2019 Oct and 2019 Dec

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 4 observations are missing between 2019 Oct and 2020 Jan

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 5 observations are missing between 2019 Oct and 2020 Feb

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 6 observations are missing between 2019 Oct and 2020 Mar

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 7 observations are missing between 2019 Oct and 2020 Apr

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 8 observations are missing between 2019 Oct and 2020 May

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 9 observations are missing between 2019 Oct and 2020 Jun

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 10 observations are missing between 2019 Oct and 2020 Jul

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 11 observations are missing between 2019 Oct and 2020 Aug

## Warning: The future dataset is incomplete, incomplete out-of-sample data will be treated as missing.
## 12 observations are missing between 2019 Oct and 2020 Sep
```

```r
plot(1:12, mae_mat[, model_names[1]],
  type = "l", col = "red",
  ylim = c(0, max(mae_mat)),
  xlab = "h", ylab = "MAE"
)
lines(1:12, mae_mat[, model_names[2]], col = "green")
lines(1:12, mae_mat[, model_names[3]], col = "blue")
lines(1:12, mae_mat[, model_names[4]], col = "black")
legend("topright",
```

```
  legend = model_names,
  col = c("red", "green", "blue", "black"),
  lty = c(1, 1, 1, 1)
)
```



ARIMA$(2,1,0)(1,1,0)_{12}$ and ARIMA$(0,1,2)(0,1,1)_{12}$ have similar MAE over the horizon h. Also, they follow a linear trend. On the other hand, ARIMA$(2,1,0)$ and ARIMA$(0,1,2)$ have similar MAE over the horizon h, but they follow a quadratic trend.

16. (Bonus^2) Evaluate the forecasts made by auto-ARIMA in this time series CV pipeline. Remember, this means that auto-ARIMA will be rerun (yikes!) at each iteration in time series CV. This may take a very long time to run (which is why this is a Bonus^2). If it finishes for you, how does its MAE compare?

```
# CODE GOES HERE
```