

Problem Set 3

Econ 6376 — Time Series Econometrics

Parameters

- Feel free to work in groups, however your code and write ups must be your own.
- Submissions should be made through Blackboard and include two files:
 - Your write up as a single PDF, **no more than 10 pages**, ending with the **AI Use Disclosure** paragraph.
 - Your **organized, commented** R script that reproduces every result in your write up, top to bottom, from `ps3_series.csv`.
- Your submission is due by 11:59pm on the due date listed in Blackboard.
- Late submissions **WILL NOT** be accepted unless previously arranged.
- Make sure your answer is sufficient for a reader that is not familiar with the problem and/or time series to understand your solution.
- Treat your write up like a professional report. If you wouldn't turn it in to your boss, don't turn it in to me.

Use of AI

AI assistance is permitted on this problem set, as it is on every graded deliverable in this course. What gets graded is understanding you can defend.

End your write up with a short, free-form paragraph headed **AI Use Disclosure** describing how you used it — which tools, for what, and where you checked its work. It does not count toward your 10 pages and it is not graded on its content. A submission without it is incomplete, in the same way a submission without your R script is incomplete.

Grading

Component	Points
Question 1 — Modeling	40
Question 2 — Forecasting	30
Question 3 — Evaluation and combinations	30
Total	100

Part	Question 1	Question 2	Question 3
(a)	10	4	7
(b)	6	10	7
(c)	14	4	10
(d)	10	12	6

The Data and the Design

The series

Download `ps3_series.csv` from the Data and Software section of the course site (https://grycrnwl.github.io/gwu6376/data/ps3_series.csv). It holds a single **custom series**, `y`, observed monthly from January 1990 through December 2019 ($N_{\text{total}} = 360$), with two columns: `date` and `y`. Treat it as you would any series you were handed: you know nothing about how it was produced beyond what the data tell you.

Read it in as a monthly `ts` object, and keep it one:

```
dat <- read.csv("ps3_series.csv")
y <- ts(dat$y, start = c(1990, 1), frequency = 12)
```

The split

This problem set uses the same three-block split as the midterm, and the same names:

```
|<----- train ----->|<-- weights -->|<-- holdout -->|
1990-01                2017-12  2018-01  2018-12  2019-01  2019-12
  estimate every model here      fit comb. weights    score Q3
                        ^
                    forecast origin T = 336 (December 2017)
```

- **train** — January 1990 to December 2017, $T = 336$ observations. Every model in this problem set is estimated here, **once**, and never re-estimated.
- **weights** — January to December 2018, $T_w = 12$ observations. Question 3 fits combination weights on this window.
- **holdout** — January to December 2019, $N = 12$ observations. Question 3 scores everything on this window.

The design is the midterm's, applied here to a common custom series. Use the 2018 and 2019 data only in the places the questions use them.

Helpers

Download `helpers.zip` from the course site. For this problem set, three files are canonical:

- `track_utils.R` — `upto(x, n)` (the first n observations of a `ts`, **keeping** its monthly frequency; use it instead of `x[1:n]`) and `kavg(x, k, dates)` (with `k = 1`, the `locf` track). Its `tracks_on()` builds the midterm's six tracks from exactly three ARIMA fits; it does not handle the regression model in Question 2, so build the four tracks this problem set needs directly, as shown in Question 2(d).
- `combination_utils.R` — `comb_simple()`, `comb_inv_mse()`, `comb_gr()`, and `comb_cls()`, each taking the weights-window actuals `y` and a $T_w \times M$ forecast matrix `F` **with column names**, and returning `list(intercept, weights)`.
- `forecast_utils.R` — `rmse()` and `mae()`.

You may write any of these yourself instead; if you do, say so.

Question 1 — Modeling (*Modules 2–5*)

Use the **training block only** (1990-01 to 2017-12) throughout this question.

- (a) What is the order of integration of y ? Provide all the evidence that supports your claim: plots, appropriately specified ADF and KPSS tests (state and justify the deterministic terms and lag choices), and the same evidence for the series after each difference you take. Discuss any evidence that conflicts with your conclusion.
- (b) Is the series seasonal? Explain, using the evidence you consider relevant. If it is seasonal, say what kind of seasonality your evidence points to and how you intend to model it.
- (c) Examine the ACF and PACF of the stationary series. Determine the model class (AR, MA, or ARMA, including any seasonal terms) and a specific starting model, and explain which features of the correlograms lead you there. Choose an information criterion **before fitting alternative models**, write down its formula, define each of its terms (the maximized log-likelihood, the parameter count k — count every estimated coefficient plus the innovation variance — and the sample size), and justify the choice. Then use the Box–Jenkins method to find your “best” candidate model: move from your starting model one justified revision at a time, and report a compact table of every model you fit with its criterion value. Identify your **best** and **second-best** models by your chosen criterion; you will forecast with both in Question 2. If candidates are within 1 unit of each other on your criterion, you may rank them in either order; say which order you chose and why.
- (d) Define serial correlation. Give two ways to examine a fitted model’s residuals for its presence (for example, the residual ACF and the Ljung–Box test); for any test, state the null hypothesis, the statistic, and its reference distribution, including the degrees-of-freedom adjustment for a fitted model. Determine whether your best candidate model exhibits serial correlation, and say what you would do if it did. Also plot the residual histogram or QQ plot and comment on the tails.

Question 2 — Forecasting (*Module 6*)

You will forecast with three models, all estimated **once** on the training block:

1. your **best** model from Question 1;
2. your **second-best** model from Question 1, by the same criterion;
3. a regression on monthly seasonal dummies with ARIMA(1,1,0) errors (an ARIMAX-style model), specified by me:

```
library(forecast)
train  <- upto(y, 336)      # same as window(y, end = c(2017, 12))
fit_dum <- Arima(train, order = c(1, 1, 0), xreg = seasonaldummy(train))
```

`seasonaldummy()` reads the calendar from the `ts` object, which is why `train` must keep its monthly frequency. To forecast this model, supply the future dummies: `forecast(fit_dum, h = 24, xreg = seasonaldummy(train, h = 24))`.

- (a) **Loss function.** State your loss function $g(\cdot)$ at the start of Question 2, before part (b) and before you look at any forecast errors, and justify it: say which errors matter most for this series given what Question 1 showed about its shocks (e.g., the tails of the residuals), and why $g(\cdot)$ reflects that.
- (b) **Fixed origin.** Standing at December 2017, produce 24-step-ahead forecasts $\tilde{y}_{T+h|T}$, $h = 1, \dots, 24$, from each of the three models, with 80% and 95% prediction intervals. Present them as fan charts (one panel per model is fine), with the actual 2018–2019 values overlaid. Why do the intervals widen with the horizon? Answer in terms of the model, not the picture.
- (c) **Score the fixed-origin paths.** Score the three 24-step paths on 2018–2019 under the loss you chose in part (a). One loss function, applied identically to all three.
- (d) **One-step-ahead tracks.** Build one-step-ahead forecast tracks $\tilde{y}_{t|t-1}^{(m)}$ for every month of 2018–2019 (24 dates) for **four** forecasters: your best model, your second-best model, the dummy model, and locf (last observation carried forward, $\tilde{y}_{t|t-1} = y_{t-1}$). Parameters stay at their training values; only the information set rolls. As in the Module 6 and 7 notes, refilter each estimated model on the longer series without re-estimating it:

```
T_origin <- 336
dates    <- T_origin + 1:24                # 2018-01 .. 2019-12
ys       <- upto(y, T_origin + 24)
os_best  <- as.numeric(fitted(Arima(ys, model = fit_best)))[dates]
os_dum   <- as.numeric(fitted(Arima(ys, model = fit_dum,
                                   xreg = seasonaldummy(ys))))[dates]
os_locf  <- kavg(y, 1, dates)
```

Check the alignment before you use the tracks: `os_locf` on date t must equal y_{t-1} exactly. Plot the four tracks against the actuals. Why do these one-step errors differ from the errors of the 24-step paths in part (b)?

Question 3 — Evaluation and combinations (*Module 7*)

Use the one-step tracks from Question 2(d). Write $e_{m,t} = y_t - \tilde{y}_{t|t-1}^{(m)}$ for forecaster m 's one-step forecast error.

- (a) Define RMSE, and give the intuition for how it behaves relative to MAE: which errors does each weight most heavily, and what does that imply for this series, given what Question 1 showed about its shocks (e.g., the tails of the residuals)? Then score all four one-step tracks — best, second-best, dummy model, and `locf` — on the **2019 holdout** by RMSE. Did any model beat `locf`? RMSE is the common yardstick for this question whatever loss you chose in Question 2(a), so that answers remain comparable across the class.
- (b) On the **2018 weights window**, fit combination weights for the pool `{locf, second-best, dummy model}` using the four Module 7 methods: simple average, inverse-MSE, Granger–Ramanathan (unconstrained least squares with an intercept), and constrained least squares (no intercept, weights non-negative and summing to one). Report the intercept and the weights for each method in one table. Your best model is deliberately left out of the pool: it is the benchmark, and the question is whether combining weaker forecasts can match or beat it.
- (c) Apply each set of weights, unchanged, to the 2019 forecasts of the same three tracks, $\tilde{y}_{t|t-1}^{(c)} = \hat{w}_0 + \sum_m \hat{w}_m \tilde{y}_{t|t-1}^{(m)}$, and report the 2019 holdout RMSE of each combination next to your best model's. Then run a Diebold–Mariano test of **each combination against your best model**: squared-error loss, `dm.test(e_comb, e_best, h = 1, power = 2)`. These four comparisons are declared here, in advance; report all four, whatever they show. For each, state the sign of the statistic, who had the smaller average loss, and whether the test rejects at 5%. With `h = 1`, the test assumes the loss differential $d_t = g(e_{1,t}) - g(e_{2,t})$, here with $g(e) = e^2$, is serially uncorrelated; say whether that assumption is plausible here.
- (d) In one paragraph: did the estimated weights pay for themselves (i) against the simple average and (ii) against your best model? Weigh the gain that estimated weights could attain against the cost of estimating them from twelve observations, using your weights from (b) and your results from (c). Given this evidence, would you combine or pick a single model? Defend your answer.