ST5209/X Assignment 2

Due 16 Feb, 11.59pm

Set up

1. Make sure you have the following installed on your system: LaTeX, R4.2.2+, RStudio 2023.12+, and Quarto 1.3.450+.
2. Clone the course repo.
3. Create a separate folder in the root directory of the repo, label it with your name, e.g. `yanshuo-assignments`
4. Copy the assignment1.qmd file over to this directory.
5. Modify the duplicated document with your solutions, writing all R code as code chunks.
6. When running code, make sure your working directory is set to be the folder with your assignment .qmd file, e.g. `yanshuo-assignments`. This is to ensure that all file paths are valid.[1]

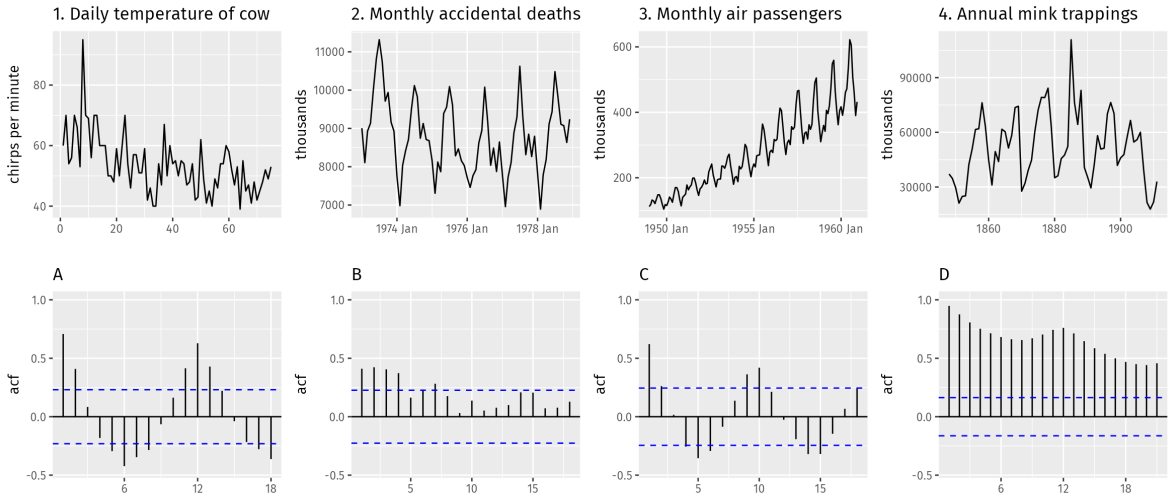
Submission

1. Render the document to get a .pdf printout.
2. Submit both the .qmd and .pdf files to Canvas.

Question 1 (ACF plots, Q2.9 in FPP)

The following time plots and ACF plots correspond to four different time series. Match each time plot in the first row with one of the ACF plots in the second row.

[1] You may view and set the working directory using `getwd()` and `setwd()`.



Question 2 (Box-Cox, Q3.3 in FPP)

Why is the Box-Cox transform unhelpful for the `canadian_gas` data?

Question 3 (Decomposition with outliers, Q3.7 in FPP)

Consider the last five years of the Gas data from `aus_production` .

```
gas <- tail(aus_production, 5*4) |> select(Gas)
```

a. Plot the time series. Can you identify seasonal fluctuations and/or a trend-cycle?

b. Use `classical_decomposition` with `type=multiplicative` to calculate the trend-cycle and seasonal indices.

c. Do the results support the graphical interpretation from part a?

d. Compute and plot the seasonally adjusted data.

e. Change one observation to be an outlier by running the following snippet:

```
# Change to eval: TRUE in order to run
gas |>
  mutate(Gas = if_else(Quarter == yearquarter("2007Q4"), Gas + 300, Gas))
```

Recompute the decomposition. What is the effect of the outlier on the seasonally adjusted data?

f. Does it make any difference if the outlier is near the end rather than in the middle of the time series?

Question 4 (STL decomposition, Q3.10 in FPP)

Consider the `canadian_gas` dataset.

a. Do an STL decomposition of the data.

b. How does the seasonal shape change over time? [Hint: Try plotting the seasonal component using `gg_season()`.]

c. Apply a calendar adjustment and compute the STL decomposition again. What is the effect on the seasonal shape?

Question 5 (Time series classification)

We continue out investigations of this dataset, which contains 600 synthetic control charts for an industrial process. There are 6 types of control charts, and our goal is to build a model to classify them correctly.

We have already processed the raw data into a convenient, labeled form. Run the following code snippet to load it and create train and test sets.

```
ccharts <- read_rds("../_data/CLEANED/ccharts.rds")
ccharts_train <- ccharts[["train"]]
ccharts_test <- ccharts[["test"]]
```

a. Make a time plot of one time series from each category in the training set. Note that the category is recorded under the `Type` column.

b. Compute all time series features for both the training and test set using the snippet. What is the difference between `acf1` and `stl_e_acf1` for Increasing, Decreasing, Upward, and Downward types? Why is there a difference?

```
# Change to eval: TRUE in order to run
train_feats <- ccharts_train |> features(value, feature_set(pkgs = "feasts"))
test_feats <- ccharts_test |> features(value, feature_set(pkgs = "feasts"))
```

c. Investigate the relationship between the following features and `Type`. Pick two features whose scatter plot gives a good separation between all 6 chart types.

   i. `linearity`

ii. `trend_strength`

iii. `acf1`

iv. `stl_e_acf1`

v. `shift_level_max`

vi. `shift_var_max`

vii. `n_crossing_points`

Make the scatter plot and explain why these features are able to separate the different chart types.

d. Install the `caret` package and use the following snippet to fit a k-nearest neighbors model on the two features you have selected (substitute `X` and `Y` for the two features you selected in b), and then predict on the test set. What percentage of the test examples are correctly classified? Write code to compute this value. (You should get $> 90\%$ accuracy)

```
# Change to eval: TRUE in order to run
library(caret)
knn_fit <- train(Type ~ X + Y, data = train_feats, method = "knn")
predict(knn_fit, newdata = test_feats)
```