UNIVERSITY OF BRITISH COLUMBIA
Department of Statistics
Stat 443: Time Series and Forecasting
Assignment 1: Exploratory Data Analysis

The assignment is due on **Thursday, January 30** at **9:00pm**.

- Submit your assignment online on `canvas.ubc.ca` in the **pdf format** under module "Assignments".
- This assignment should be completed in **RStudio** and written up using **R Markdown**. Display all the R code used to perform your data analyses.
- Please make sure your submission is clear and neat. It is the student's responsibility that the submitted file is in good order (i.e., not corrupted).
- Remember to properly label all your plots and have them clearly displayed.
- **Late submission penalty**: 1% of the assignment score per hour or fraction of an hour. (In the event of technical issues with submission, you can email your assignment to the instructor to get a time stamp but submit on canvas as soon as possible to make it available for grading.)

1. The file `employee_wages_total_industry.csv` contains average hourly wage rates in Canada from January 1997 to November 2024, computed over all industry classes. The wage rate includes both full-time and part-time employees of both sexes, and only those aged 15 or older are considered.

   Data source: Statistics Canada, Table 14-10-0063-01. Employee wages by industry, monthly, unadjusted for seasonality. https://www150.statcan.gc.ca/t1/tbl1/en/tv.action?pid=1410006301).

   (a) Read in the data and create a time-series object. Plot the series and comment on any features of the data that you observe. In particular, address the following points:
   - Does the series have a trend?
   - Is there seasonal variation, and if so would an additive or multiplicative model be suitable? Explain your reasoning.
   - Is the series stationary? Justify referring to the *definition* of a weakly stationary stochastic process.

   (b) Create training and test datasets. The training dataset should include all observations up to and including December 2018; this dataset will be used to fit ("train") the model. The test dataset should include all observations from January 2019 to December 2020; this dataset will be used to assess forecast accuracy. You can use the command `window()` on a `ts` object to split the data.

   Using a suitable decomposition model and the loess method (`R` function `stl()`), decompose the training series into trend, seasonal, and error components. Plot the resulting decomposition.

   (c) Fit a linear model to the trend component (you can use `R` function `lm()`).

- Write down the fitted model for the trend component.
- Does the linear model provide evidence of a trend at the 95% confidence level? Without doing any further analysis, would you use this trend component to make predictions? Justify your answer using the linear model results and the trend component plot.

(d) Examine the sample autocorrelation function of the error term from the seasonal decomposition model. Comment on whether it is consistent with a white noise process.

(e) Predict the monthly average values of the hourly wage rates for the period from January 2019 to December 2020 using your seasonal decomposition model.

- Plot your predictions along with the actual observed values (on the same plot). Make sure to include a legend for your plot.
- Comment on the performance of your prediction method, explain why or why not the method worked well for this data, and summarize what you learned from it.
- As a statistician, what other information would you like to add to your forecasts in addition to the point forecasts you produced above?

2. The file `MSFT_closing_price.csv` contains the daily closing prices (in USD) of the Microsoft Corp. from 2010 to the end of 2019.

In this question, we introduce the `zoo` package which is useful when working with time series of irregular frequencies or aggregating high frequency data into a lower frequency (e.g., aggregating daily data into monthly means or maxima).

Instructions for working with `zoo` objects are given below:

- Load the `zoo` library using the command `library('zoo')`. If you do not have this package installed, type `install.packages('zoo')`;
- Use the command `zoo(x, as.Date(dat$Date))` to create `zoo` object `x`;
- Create monthly mean from the daily data by using the command `aggregate(x, as.yearmon, FUN=mean)`.

(a) Read the data into `R`, create a `zoo` object for daily closing prices, plot the time series and comment on its features.

(b) In financial risk management, it is typical to model daily log-returns defined as $X_t = \log(S_t/S_{t-1})$, where S_t denotes the closing price at time t (the series of daily log-returns can be viewed as an approximation to daily *relative* returns $(S_t - S_{t-1})/S_{t-1}$). Plot the series of daily log-returns and its sample autocorrelation function and comment on the stochastic behaviour of the daily log-returns series. (You can apply command `coredata` to your `zoo` object in order to use the `acf` function.)

(c) Now consider the absolute values of the daily log-returns, $\{|X_t|\}$. What can you say about serial dependence of this series? How can you reconcile your observations here with those in part (b)?

(d) Create monthly mean time series by aggregating values of series $\{X_t\}$. Plot the monthly returns series and compare it to daily series. Do you expect the monthly return series to exhibit weaker or stronger serial dependence compared to the daily return series? Explain your reasoning.

(e) Commands `qqnorm()` and `qqline()` can be used to make normal quantile-quantile (Q-Q) plots, which compare theoretical normal quantiles against empirical quantiles based on the data. Use these commands to assess whether the distribution of either of the daily or monthly index log-returns can be well approximated by a normal distribution. Can you provide any insights on your findings?

3. In this question you will explore the sampling distribution of the sample autocorrelation coefficient for a white noise process through a simulation study. Recall that, for a time series of length n, from a white noise process, the sample autocorrelation coefficient at lag h approximately follows a normal distribution with mean $-1/n$ and variance $1/n$:

$$r_h \sim \mathcal{N}(-1/n, 1/n)$$

for large values of n.

To confirm this theoretical fact, conduct the following simulation study for lags $h = 1$ and $h = 2$:

(i) Simulate a time series of length $n = 1000$ from a white noise process $\{Z_t\}_{t\in\mathbb{Z}}$ with $Z_t \sim \mathcal{N}(0, 1)$ (function `rnorm()`).

(ii) Evaluate r_h, the sample autocorrelation coefficient at lag h, for $h = 1$ and $h = 2$. Store these values.

(iii) Repeat steps (i) and (ii) $m = 10,000$ times; i.e., generate 10,000 time series of length n and for each of them compute r_1 and r_2 (you can use `for` loop). You should now have two vectors of length m with estimates r_1 and r_2.

To summarize results of the simulation study, present the following information:

- Compute the mean and variance of r_1 and r_2 values from your simulation study.
- In two separate figures, plot the two histograms for the sample of r_1 and r_2 values from the simulation study (function `hist()`), add the smoothed version of the histogram (function `density()`) and the theoretical asymptotic normal density (function `dnorm()`). Make sure your plots are well-presented, including a suitable title, axes labels, curves of different type or colour, and a legend.
- Comment whether there is an agreement between the empirical estimates of the bias, variance and sampling density of the estimator of the autocorrelation at lag h and their theoretical approximation.