STAT 443: Time Series and Forecasting

Lab 1
Introducing Time Series in R.
Sample Autocorrelation Function.

Objectives of the lab:

- Learn to read time series data into `R`.
- Create a time series object using package `ts`.
- Perform basic manipulations on time series.
- Plot a time series and identify its main features.
- Plot a sample autocorrelation function and interpret its behaviour.
- Creating a reproducible report using `R Markdown`.

Getting started

- Open *RStudio*
- Create a new `R` script: File → New File → R Script
- Set working directory: Session → Set Working Directory → ...
- Install (via Tools → Install Packages) and load necessary package(s)

  ```
  library(tseries)
  ```

- To use the "Help" facility in R, enter `help(function)` or just `?function`, which will produce the help page for the R command/object `function`.

1. In this question, you will learn how to read in data into `R`, define a time series object and plot time series.

   (a) (**Reading in data**) Datasets can be read into `R` by one of several methods, the most common being `read.table()` and `read.csv()`

   - The dataset "LakeLevels.csv" contains the daily depths (in meters) of a lake from 2007 to 2011 inclusive. Read the data into `R` using `read.csv()`. Use the command `<-` to assign your dataset to a named object, `dat` say.

     To look at the data, you can use commands `head()` and `tail()`.
     Try `head(dat,10)` to view the first 10 rows of the data.
   - Objects in `R` are each of a prescribed "class", this determining characteristics of the object and how functions will act on the object. Your object is a so–called "dataframe", which is similar to a matrix except it allows for different columns to be of different class and uses a different vocabulary for its manipulation.
     The command `names()` can be used on a data frame to obtain a list of the vectors contained in the data frame. The call `object$name` extracts the vector `name` from the data frame `object`.
   - Create a plot of your dataframe `dat`. How does this plot differ from one you would like for these data?

   (b) **(Creating a time series object)** Time series objects in `R` are of either `ts` or `zoo` class. We will initially learn how to work with the `ts` object.

   - To determine whether the dataframe you have created is of class `ts`, the command `is.ts()` can be used.
   - Objects can be coerced into being time series objects using `ts` command. Read the help page on the command `ts`.
   - Create a time series object `x` containing the lake level data, correctly specifying arguments `start` and either `end` or `frequency`.

   (c) **(Plotting time series)** Using command `plot()`, make a plot of your time series object, giving your plot a suitable title and labels for the axes using the `main`, `xlab,` and `ylab` arguments. How does your new plot differ from your first plot?

   Comment on the features of this time series.

2. In this question, you will explore the sample autocorrelation function (acf) for a white noise process.

   (a) Use the `rnorm()` function to create 200 independent observations from the standard normal distribution. Create an object for your data $\{x_t\}_{t=1,2,\ldots,200}$ and coerce it into class `ts`.

   (b) Plot your simulated time series $\{x_t\}$. How many of your observations are outside the range ± 2? How many would you expect to be outside ± 2?

   (c) Use the function `acf()` to create the sample autocorrelation function for your time series. Comment on the features of the sample acf here.

You can now prepare a lab report using `R Markdown` by either creating a new R Markdown file (File $\rightarrow$ New File $\rightarrow$ R Markdown) or using the provided R Markdown template. Knit your file to either PDF or HTML and upload as your lab submission.

Useful R commands for time series analysis

- `read.csv()`
- `ts()`
- `window()`
- `start()`
- `end()`
- `time()`
- `diff()`
- `lag()`
- `lag.plot()`
- `head()`
- `tail()`
- `class()`
- `str()`