# Recitation 2

Data Analytics (18-899)

Friday 11 February, 2022

# Note on recitation slides

- Recitation slides are intended to be a guide on how to approach the assignment and not a prescription of exactly what to do
- There could be many approaches to any problem
- Seek to understand the problem and solve it instead of just trying to reproduce the steps listed in the slides
- To avoid overdependence on the slides, start your assignment early!
- If you have to choose between a “creative” approach, and following assignment instructions in the PDF, choose to PDF instructions **always**.

# Assignment Objectives

- Explore and visualize time series data
- Investigate variability in time series data over different timescales
- Perform statistical measures: mean reversion, stationarity
- Evaluate model performance based on benchmarking
- Fit ARIMA model

# Question 1

- Download the ***WindGeneration.csv*** file
- Generate dates/timestamps
- Plot wind generation against the timestamps
- Is there any evidence of intra-annual seasonality?
  - ➔ Daily
  - ➔ Weekly
  - ➔ Monthly
  - ➔ Quarterly

# Question 2

- Calculate the change in wind generation over time as a percentage of  the maximum generation.
- Plot it against the timestamps
- Is there any evidence of seasonality?

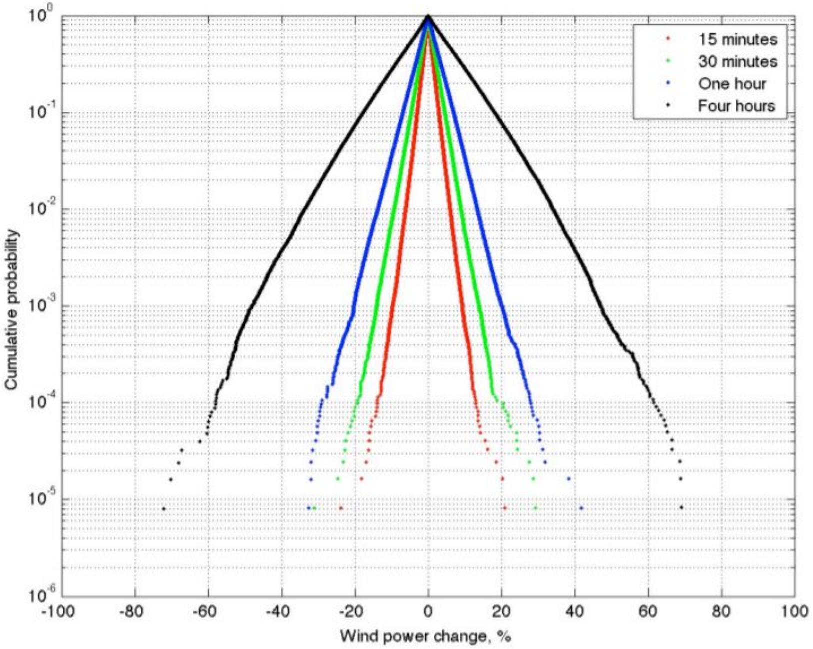
# Question 3

- Ramp function defined as

$$r(t,d) = 100 * [x\,(t + d) - x(t)\,] / \max(x)$$

- Calculate the ramps where d=1 for an hourly sampling period
- Separate them into positive and negative ramps
- Use their absolute values and sort the resulting dataframe
- Plot the ramps using a semilogy plot (vertical logarithmic axis)
- Plot a normal distribution CDF on the same graph

# Wind power fluctuations



# Question 4

- Investigate the variability in wind generation over different timescales

- Timescales: 1h, 2h, 3h, …., 24h

- Use the percentile analysis on the ramps

  **r(t,d) =100 * [x (t + d) - x(t) ] / max(x)**

  Iterate over the ***d*** param using d=1...24

  Compute the percentiles (1%, 5%, 95% and 99%) for each d
- Plot the results
- What did you learn?

*Python: numpy.percentile()     MATLAB: prctile()*

# Question 5

- Calculate the autocorrelation of the wind generation (actual) for 10 days lags

  *10 days lags = 1hr * 24hr * 10 = 240*

- Comment on the autocorrelation plot

# Question 6

- Calculate the autocorrelation of **change in wind generation** for lags over 10 days
  - Calculate the change in wind generation
  - Calculate the autocorrelation with lags of 10 days (240 hours)
- Plot the resulting data

# Question 6 cont'd

- Plot horizontal lines to detect statistically significance values (p<0.05)
    - Corresponding value can be calculated from the normal distribution
    - Plot it for every value
- Is there any evidence of diurnal seasonality?
- Might it be more appropriate to model the change in wind generation than the actual wind generation?

# Question 7

- The variance ratio test will be used to investigate the structure of the  wind generation time series.

  *MATLAB: vratiotest  Python: Arch library, Adfuller from statsmodels*

- Using the result from the functions above, can the null hypothesis of a random walk be rejected?
- Test mean reversion
- Is there evidence of either mean-reversion of mean aversion?

# Question 8

- Test window size using n = [1: 24]
- For each n, calculate the simple moving average (SMA)

  *MATLAB: tsmovavg or equivalent  Python: df.rolling()*

  Calculate the mean absolute error (MAE) between the SMA and the actual wind power.
- For which n, do you obtain the minimum MAE?
- Is there a simple benchmark that improves on the persistence  benchmark?

# Question 9

- For each forecast horizon n = [1 : 24]
- Calculate the persistence of ***n*** X_predicted(t) = X(t-n)
- Handle missing or NaN values
- Calculate the mean absolute error (MAE) between the predicted wind power and the actual wind power
- Plot MAE as a percentage of the maximum generation for the persistence benchmark. *Hint (X = timescale, Y = max MAE percentage wind generation)*

# Question 10

- Understand an AutoRegressive Integrated Moving Average (ARIMA) model
- Find parameters it takes (p, d, q)
- Loop through a range of parameters (p and q = [1:4]) to find the optimal parameters
    - Pass the parameters to the arima model
    - Fit and return the model estimates
    - calculate the AIC and BIC from the estimation

# Question 10 cont'd

- Determine if there is some improvement in the model's performance (small AIC and BIC are better)
- What are the optimal ARIMA model parameters?

# Submission Instructions

Submission Instructions

- **Single** Python/MATLAB code file(.ipynb or .m) **[Do not Submit checkpoints for .ipynb]**
- Assignment report(.pdf) - remember to name the file as instructed
  - Indicate the libraries you have used in your code at the beginning of the report (After the title page)
- Data files (as given)

**Submission process:**

1. Put code **file and data files** in a single folder
2. Name of the folder should be the same as your andrew ID
3. **Zip this folder and attach the zipped file on assignment submission page (CANVAS)**
4. After attaching zipped file, click on "Add Another File" from assignment submission page and **attach your report**
5. Submit your assignment

# Q&A