Review – Hydro-informatics

1. Info

1.1. Information Technology & Aquatic Environment

Environmental concern
-Permanent Effects: Blocking, Water Exchange of the Baltic Sea, Ecology of the Baltic, sensitive to salinity.

-Temporary Effects: Dredging Spill, Shading of Algae, Impact on Mussels, Fish Migration,Birds.

1.2. Kent Ridge Experimental Catchment

Infiltration rates in cities as influenced by greenery
Variation of soil type and land use;
Using tension disk infiltrometers;
Distrubed soil sampling for water content.

1.3. Data Handling and Analysis in Hydroinformatics– Attributes

Attribute Type	Description	Operations
Nominal	The values of a nominal attribute are just different names, i.e., nominal attributes provide only enough information to distinguish one object from another. (=, ≠)	mode, entropy, contingency correlation, $\chi 2$ test
Ordinal	The values of an ordinal attribute provide enough information to order objects. (<, >)	median, percentiles, rank correlation
Interval	For interval attributes, the differences between values are meaningful, i.e., a unit of measurement exists. (+, -)	mean, standard deviation, Pearson's correlation, t or F tests
Ratio	For ratio variables, both differences and ratios are meaningful. (*, /)	geometric mean, harmonic mean, percent variation

Basic classification of data:

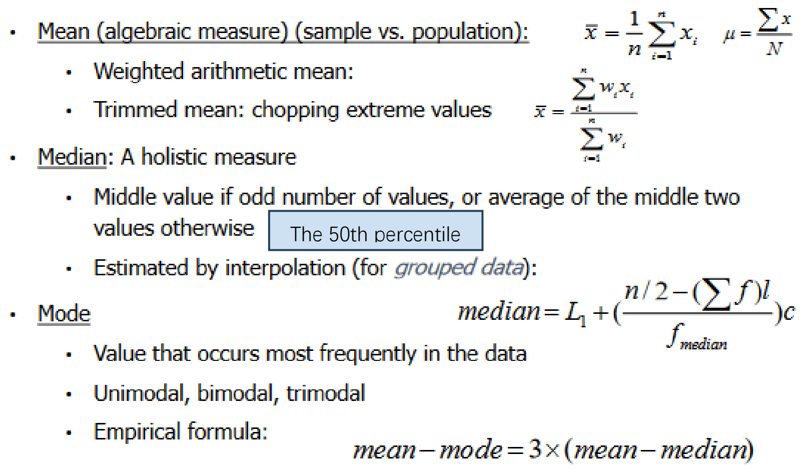
- Processes: Hydrological, Meteorological, Oceanographic, Environmental, Hydrodynamic, Biological, Geographical.
- Nature of data: Time series, Spatial series, Continuous, Discrete, Parameters, Constants.
- Data types: Numeric, Text, Date/Time, Image, Audio/video.
- Source: In-situ measurement, Altimetry, Standards, Simulation results, Residuals, Processed data.
- Data structure: Relational, Hierarchical, Unstructured/Manual.

More issues: Data acquisition (collection)…

2. Data Visualisation - Descriptive Statistics

2.1. Central Tendency – Mean, Median and Mode

Central Tendency (or Groups' "Middle Values") • Mean • Median • Mode



- Mean (algebraic measure) (sample vs. population): $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad \mu = \frac{\sum x}{N}$
  - Weighted arithmetic mean: $\bar{x} = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$
  - Trimmed mean: chopping extreme values
- Median: A holistic measure
  - Middle value if odd number of values, or average of the middle two values otherwise

    The 50th percentile
  - Estimated by interpolation (for *grouped data*):

$$median = L_1 + \left(\frac{n/2 - (\sum f)l}{f_{median}}\right)c$$

- Mode
  - Value that occurs most frequently in the data
  - Unimodal, bimodal, trimodal
  - Empirical formula:

$$mean - mode = 3 \times (mean - median)$$

2.2. Used to Report on Populations and Samples

Organize Data

• Tables: Frequency Distributions, Relative Frequency Distributions;

• Graphs: Bar Chart or Histogram, Stem and Leaf Plot, Frequency Polygon.

2.3. Summarizing Data - Measures of the Dispersion of Data

Variation (or Summary of Differences Within Groups)

i. Range

The spread, or the distance, between the lowest and highest values of a variable.

range for a variable = highest value - lowest value

ii. Interquartile Range

The interquartile range is the distance or range between the 25th percentile and the 75th percentile.

iii. Variance

A measure of the spread of the recorded values on a variable.

A measure of dispersion.

- The larger the variance, the further the individual cases are from the mean.
- The smaller the variance, the closer the individual scores are to the mean.

$$\sum_{i=1}^{n} \frac{(x_i - \mu)^2}{n} = \sigma^2,$$

$$\sqrt{\sum_{i=1}^{n} \frac{(x_i - \mu)^2}{n}} = \sigma$$

iv. Standard Deviation

Variance is the square of Standard Deviation, to eliminate negative signs:

Var = StDev^2

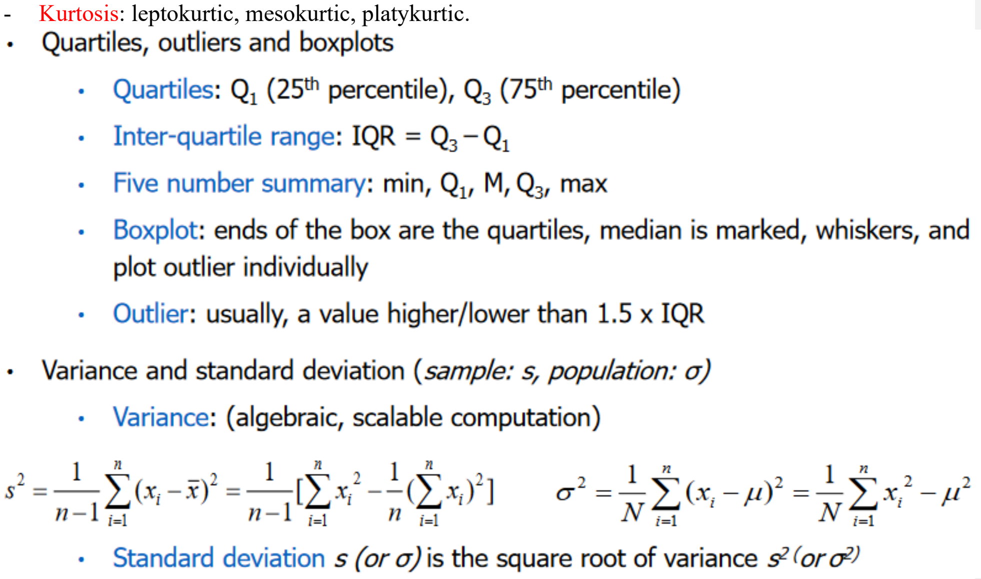
v. Statistical Distribution - Symmetric vs. Skewed Data

- Symmetric: Normal distribution (typically)

- Skewness - Asymmetrical distribution

Positive skew -. Right skew
Negative skew -. Left skew

- Kurtosis: leptokurtic, mesokurtic, platykurtic.



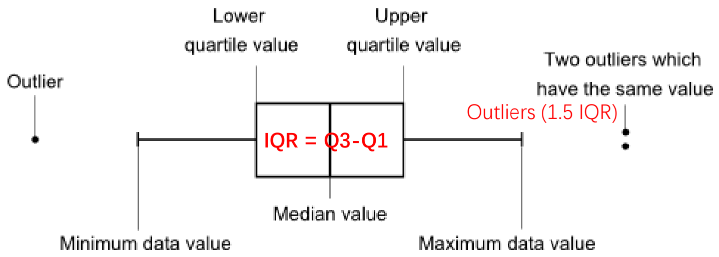
- Quartiles, outliers and boxplots
  - Quartiles: Q_1 (25th percentile), Q_3 (75th percentile)
  - Inter-quartile range: IQR = $Q_3 - Q_1$
  - Five number summary: min, Q_1, M, Q_3, max
  - Boxplot: ends of the box are the quartiles, median is marked, whiskers, and plot outlier individually
  - Outlier: usually, a value higher/lower than 1.5 x IQR
- Variance and standard deviation (*sample: s, population: σ*)
  - Variance: (algebraic, scalable computation)

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2 = \frac{1}{n-1}[\sum_{i=1}^{n}x_i^2 - \frac{1}{n}(\sum_{i=1}^{n}x_i)^2] \qquad \sigma^2 = \frac{1}{N}\sum_{i=1}^{n}(x_i-\mu)^2 = \frac{1}{N}\sum_{i=1}^{n}x_i^2 - \mu^2$$

  - Standard deviation s (*or* σ) is the square root of variance s^2 (*or* σ^2)

> **Boxplot**

> A way to graphically portray almost all the descriptive statistics at once is the boxplot.



Interpreting a boxplot can be done once you understand what the different lines mean on a **box and whisker diagram**.

The line splitting the box in two represents the **median value**. This shows that 50% of the data lies on the left hand side of the median value and 50% lies on the right hand side.

The left edge of the box represents the lower quartile; it shows the value at which the first 25% of the data falls up to. The right edge of the box shows the upper quartile; it shows that 25% of the data lies to the right of the upper quartile value.

The values at which the horizontal lines stop at are the values of the upper and lower values of the data. The single points on the diagram show the **outliers**.

- Index of Qualitative Variation

3. Data Pre-processing

3.1. Data Dirty – split out

1) **incomplete:** lacking attribute values, lacking certain attributes of interest, or containing only aggregate data;
2) **noisy**: containing errors or outliers;
3) **inconsistent:** containing discrepancies in codes or names.

Multi-Dimensional Measure of Data Quality • A well-accepted multidimensional view: • Accuracy • Completeness • Consistency • Timeliness • Believability • Value added • Interpretability • Accessibility • Broad categories: • intrinsic, contextual, representational, and accessibility.

> Data extraction, cleaning, and transformation aim to provide a quality data.

Broad categories: • intrinsic, contextual, representational, and accessibility.

Data Pre-processing	Major Tasks	quality data
Data cleaning	Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies	A well-accepted multidimensional view: • Accuracy • Completeness • Consistency • Timeliness • Believability • Value added • Interpretability • Accessibility
Data integration	Integration of multiple databases, data cubes, or files	
Data transformation	Normalization and aggregation	
Data reduction	Obtains reduced representation in volume but produces the same or similar analytical results	
Data discretization	Part of data reduction with particular importance, especially for numerical data	

3.2. Data Cleaning

- **Data cleaning tasks**

1) Fill in missing values;
2) Identify outliers and smooth out noisy data;
3) Correct inconsistent data;
4) Resolve redundancy caused by data integration.

- **Handle Missing Data**

1) Ignore the observation: usually done when class label is missing (assuming the tasks in classification - not effective when the percentage of missing values per attribute varies considerably. – NA

2) Fill in the missing value manually or automatically: Point Estimation

- **Duplicate Data**

1) Identify and Remove;
2) Drop Duplicates: Use functions or methods provided by your programming language or data manipulation libraries to drop duplicate rows based on certain criteria.

3.3. Data Integration

Combines data from multiple sources into a coherent store.
Detecting and resolving data value conflicts.

- **Handling Redundancy in Data Integration**

1) Redundant data occur often when integration of multiple databases;
2) The same attribute may have different names in different databases;
3) One attribute may be a “derived” attribute in another table;
4) Redundant data may be able to be detected by correlational analysis;
5) Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality. 仔细整合多个来源的数据可能有助于减少/避免冗余和不一致，并提高挖掘速度和质量

3.4. Data Transformation and Discretization

3.4.1. Normalization

Data are scaled to fall within a small, specified range [0,1].
They will show better properties that will help the predictive power of the model.

Commented [XP1]: Enhancing Model Interpretability: Normalizing data makes it easier to interpret the coefficients or weights learned by the model. When features are on the same scale, the magnitude of the coefficients directly reflects their importance or contribution to the output, making the model more interpretable and actionable.

- min-max Normalisation

$$v' = \frac{v - min_A}{max_A - min_A}(new - max_A - new - min_A) + new - min_A$$

- z-score Normalisation

$$v' = \frac{v - \overline{A}}{\sigma_A}.$$

- Normalisation by decimal scaling

$$v' = \frac{v}{10^j},$$

Where j is the smallest integer such that
Max(| v' |)<1

3.4.2. Smoothing and Outliers

> **Handle Noisy Data; Remove noise from data.**

It can help in making the underlying structure more apparent, especially in datasets where there is a lot of variability or random fluctuations.

1) Simple Discretization Methods: Binning

Commented [XP2]: Note: smoothing

Binning is one approach to data smoothing, where **data points are grouped into bins** or intervals and replaced by a summary statistic (such as the mean, median, or mode) of the data points within each bin. Binning can help reduce the impact of outliers and noisy data points, making the overall trend more apparent.

- Equal-width (distance) partitioning: W= (B–A)/N.
- Equal-depth (frequency) partitioning: Divides the range into N intervals, each containing approximately same number of samples.

Though **scatterplot** to visualize and verify smoothing res.

2) Running line (**local regression**)
3) **Polynomial**: Linear and cubic parametric least squares fits.

4) Non-parametric Smoothing: the Loess Method

- LOWESS= LOESS is an Acronym for **LO**cally re**WE**ighted **S**catterPlot **S**moothing.
- In the LOESS (LOWESS) method, weighted least squares is used to fit linear or quadratic functions of the **predictors** at the centers of neighborhoods.
- The smooth at the target point is the fit of a **locally weighted linear fit** (tricube weight).
- Pseudo code:

Yi=g(xi) + ei

where g is the **regression function** and ei is a **random error**.

3.4.3. Detrending

Commented [XP3]: note

Necessity

1) Detrending data involves removing the trend component from a dataset. This is done to better isolate and analyze other patterns or variations in the data that may not be related to the overall trend.
2) Detrending is often necessary because many datasets exhibit a trend over time due to various factors such as growth, seasonality, or other underlying processes. By detrending the data, you can focus on the fluctuations around this trend, which may contain valuable information for analysis or modeling.

- There are several reasons why detrending data may be beneficial

1) **Improved Analysis**: Removing the trend allows for a clearer analysis of other patterns or fluctuations in the data, which may be obscured by the trend.
2) **Stationarity**: Detrending is often a preprocessing step in time series analysis to ensure that the data is stationary. Stationarity is a key assumption in many time series models, and detrending helps to achieve this by stabilizing the mean and variance of the data over time.
3) **Noise Reduction**: Trends in data can sometimes be the result of noise or random fluctuations. Detrending can help to separate these random fluctuations from the underlying structure of the data.
4) **Forecasting**: Detrending can be useful for forecasting future values by isolating the component of the data that is not related to the overall trend. This can lead to more accurate forecasts,

especially when the trend component is expected to continue into the future.

For an additive series: Data value = **Trend** + **Seasonal** + **Irregular**; Otherwise, respectively.

- First thing: look at the series you are analysing before you start.

Draw a graph.
Look for the different components.
Think about what might be the best way of analysing it.
Log Transformation: Makes the distribution less skewed

- **For stochastic trend**

1) **Differencing:** differencing is generally good for removing trend from time series data.
2) For **linear** trend, new data is $\boldsymbol{Zt = Yt - Yt\text{-}1}$
3) To remove **quadratic** trend, do it again: $\boldsymbol{W_t = Z_t - Z_{t-1} = Y_t - 2Y_{t-1} + Y_{t-2}}$
4) Like taking derivatives

- **For deterministic trend**

Commented [XP4]: Lagging or leading time

1) Fit a plain linear regression, then subtract it out:
2) Fit Yt = m*t + b,
3) New data is Zt = Yt – m*t – b
4) or use quadratic fit, exponential fit, etc.

3.5. Data Reduction - Dimensionality Reduction PCA

Commented [XP5]: note

Dimensionality Reduction - **Principal Component Analysis (PCA)**

Principal Component Analysis (PCA) is a commonly used technique for dimensionality reduction and feature extraction. Its principle involves transforming high-dimensional data into a lower-dimensional space through a linear transformation, aiming to retain as much of the original data's variance as possible while minimizing information loss.

The principle of PCA can be understood as identifying the primary features or principal components within the data and then projecting the data onto these components. Principal components are the directions in the data that capture the highest variance, representing the most significant patterns of variation. By retaining the principal components with the highest variance, PCA effectively reduces the dimensionality of the data.

Here are the main steps in applying PCA:

1) **Data Standardization**: Standardize the data to ensure that each feature has a mean of 0 and a variance of 1, removing any differences in scale among features.
2) **Compute the Covariance Matrix:** Calculate the covariance matrix of the standardized data, representing the relationships between features.
3) **Compute Eigenvalues and Eigenvectors:** Perform an eigen decomposition on the covariance matrix to obtain the eigenvalues and corresponding eigenvectors.
4) **Select Principal Components:** Based on the eigenvalues, select the top k eigenvectors as the principal components, where k is the desired dimensionality of the reduced data.
5) **Data Projection:** Project the original data onto the selected principal components to obtain the lower-dimensional representation of the data.

4. Linear Regression

4.1. Covariance and Correlation – Model diagnostics

4.1.1. Covariance

Covariance is a statistical measure of the **joint variability** of two random variables.
Covariance refers to a systematic relationship between two random variables in which a change in the other reflects a change in one variable.

$$Cov_{XY} = \frac{\Sigma(X-\overline{X})(Y-\overline{Y})}{N-1}$$

When X increases and Y increases: cov(x,y)= positive;
When X increases and Y decreases: cov(x,y)= negative.
A higher number denotes higher dependency.

4.1.2. Correlation

- A strong association means that knowing one variable helps to predict the other variable to a large extend.
- The correlation coefficient is a numerical value expressing the **strength of the association**.

When the covariance is normalized, one obtains the Pearson correlation coefficient, which gives the goodness of the fit for the best possible linear function describing the relation between the variables.

$$\rho_{xy} = Correlation\ (x, y) = \frac{cov(x, y)}{\sqrt{var(x)}\sqrt{var(y)}.}$$

- Correlation ranges from -1 to +1.
- If the correlation value is 0, it suggests that there is no linear link between the variables, but another functional relationship may exist.
- 1 indicates a perfect positive linear relationship between two variables. When one variable increases, the other variable also increases proportionally, and vice versa.
- -1 indicates a perfect negative linear relationship, and suggests a strong negative association between X and Y.

4.1.3. R2 statistic – Good-fit test

The R2 statistic has become the almost universally standard measure for model fit in linear models. R-squared (R^2) represents the proportion of the variance in the dependent variable that is explained by the independent variable(s) in a regression model. It is a measure of **how well the independent variable(s) predict the variation in the dependent variable.**

$$R^2 = 1 - \frac{\sum(y_i - f_i)^2}{\sum(y_i - \bar{y})^2}$$

← Model error

← Variance in the dependent variable

- R-squared values range from 0 to 1, where:

- 0 indicates that the independent variable(s) **do not explain** any of the variance in the dependent variable.
- 1 indicates that the independent variable(s) **perfectly explain all** the variance in the dependent variable.

- However, sometimes, a higher R2 means **overfitting**.

Overfitted models will have high R2 values, but will perform poorly in predicting out-of-sample cases.
Too much complexity can diminish model’s accuracy on future data -> Bias Variance Tradeoff.

4.1.4. Residuals - ε

Residual analysis is a statistical technique used to assess the adequacy of a model's fit to the data by examining the residuals, which are the differences between observed values and the values predicted by the model.
Residuals are **calculated as the differences** between the observed values (actual data points) and the values predicted by the model.

- **residual analysis**

A "residuals versus fits plot" is the most frequently created plot. It is a scatter plot of residuals on the y-axis and fitted values (estimated responses) on the x-axis.

Here are the characteristics of a well-behaved residual vs. fits plot and what they suggest about the **appropriateness** of the simple linear regression model:

1) The residuals **"bounce randomly" around the residual = 0 line**. This suggests that the assumption that the relationship is linear is reasonable.
2) The residuals roughly form a **"horizontal band"** around the residual = 0 line. This suggests that the variances of the error terms are equal.
3) No one residual "stands out" from the basic random pattern of residuals. This suggests that there are **no outliers**.

4.1.5. Heteroscedasticity

When the requirement of a constant variance is violated, we have heteroscedasticity.
It means that the spread of the residuals (or errors) is not constant across different values of the predictor variables.
Heteroscedasticity violates one of the assumptions of ordinary least squares (OLS) regression, namely, the assumption of constant variance of residuals. This can lead to inefficient and biased estimates of the regression coefficients, as well as incorrect inferences about the statistical significance of the predictors.

1) **Biased Estimates:** Heteroscedasticity can lead to biased estimates of the regression coefficients. This means that the coefficients obtained from the model may not accurately reflect the true relationships between the independent and dependent variables.
2) **Inefficient Estimates**: This can reduce the precision of the estimates and weaken the statistical

power of the model.

4.1.6. p-value

- Definition: The p-value represents the probability of observing the data or more extreme results if the null hypothesis is true. It measures the level of statistical significance of the observed results.
- Interpretation: **A small p-value (typically less than a predetermined significance level, often denoted as α, such as 0.05) suggests strong evidence against the null hypothesis.** It indicates that the observed data are unlikely to have occurred by random chance alone, leading to the rejection of the null hypothesis in favor of the alternative hypothesis. Conversely, a large p-value indicates weak evidence against the null hypothesis. It suggests that the observed data are reasonably consistent with the null hypothesis, and there is insufficient evidence to reject it.

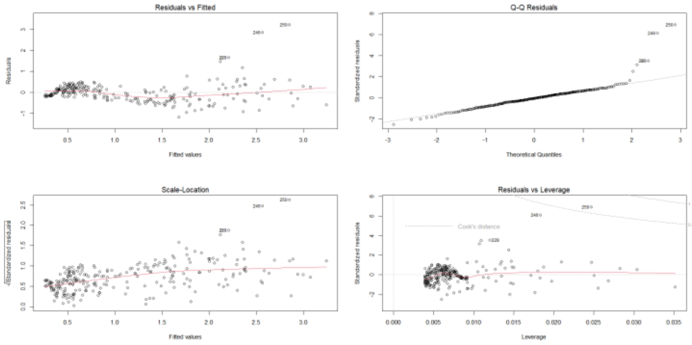
4.2. Measures of Accuracy – Model diagnostics

Model diagnostic procedures involve both graphical methods and formal statistical tests. These procedures allow us to explore whether the assumptions of the regression model are valid and decide whether we can trust subsequent inference results.

The constructed model is valid:

1) **p-value** = 2.144e-11<<0.05, suggests strong evidence against the null hypothesis, and model results are highly statistically significant.
2) **R-square** = 0.7156 ~ 1, indicating that most data can be fitted by the model.
3) **Residuals vs. fitted:** This plot shows the line *y* approximates to 0 and slightly fluctuates without the obvious trend, meaning the assumption of constant variance of residuals (homoscedasticity) and the assumption of linearity is reasonable.
4) **Normal Q-Q plot:** Check the normality residuals and the validity of model. The residuals are normal as this graph is close to a straight line, that means the model is good.
5) **Scale-Location**: Check Identically Distributed & Assess the homogeneity of variances. It shows the fitted line is close to be horizontal, but the points do not equally (randomly) spread.
6) Cook's distance: It is seen that all cases are well inside of the Cook’s distance lines. Typically, points with Ci greater than 1 are classified as being influential.

Commented [XP6]: Cook's distance is a measure used to assess the influence of individual data points (observations) on the regression coefficients and overall model fit in a regression analysis.



4.3. Simple Linear Regression

1) Describing Relationship between Two (and more) Quantities
2) The goal of **linear regression** is to **fit a straight line**, ŷ = ax + b, to data that gives best prediction of y for any value of x.
3) This will be the line that **minimises distance between data and fitted line**, i.e. the **residuals**.

➔ BEST FIT LINE

4.4. Multiple Linear Regression

The different x variables are combined in a linear way and each has its own regression coefficient.

5. Time series - TSA

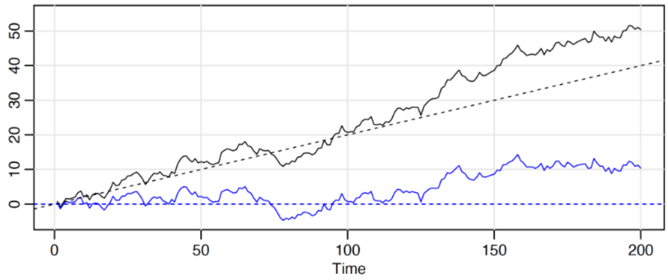
5.1. Intro - TSA

Time series is a realization or sample function from a certain stochastic process.
A time series is a set of observations generated sequentially in time. Therefore, they are dependent to each other. This means that we do NOT have a random sample.

- We assume that observations are equally spaced in time.
- We also assume that closer observations might have stronger dependency.

Cases: The Tren d Component, The Cycle Component, The Seasonal Component (Seasonality).

Detrending: Random Walk without and with Drift



5.2. Autocorrelation, Cross-Correlation and Stationarity

5.2.1. Stationary

1) **A strictly stationary process** is one where the distribution of its values remains the same as time proceeds, implying that the probability lies in a particular interval is the same now as at any

point in the past or the future.

2) However, we tend to use the criteria relating to a '**weakly stationary process**' to determine if a series is stationary or not.

- constant mean
- constant variance
- constant autocovariance structure: refers to the covariance between y(t-1) and y(t-2) being the same as y(t-k) and y(t-k-1).

The autocorrelation analysis helps in **detecting hidden patterns and seasonality and in checking for randomness**.

5.2.2. ACF

1) The ACF measures the correlation between **a time series and its lagged values**. In simpler terms, it shows how each observation in a time series is related to its past observations at different lags.
2) It is defined as the correlation between the time series at time *t* and the time series at time (*t-k*), where k is the lag.
3) ACF is often used to identify the presence of seasonality or trend in the data. If ACF shows significant correlations at specific lags, it suggests that those lags are important for forecasting.

$$\rho(s,t) = \frac{\gamma(s,t)}{\sqrt{\gamma(s,s)\gamma(t,t)}} \qquad \rho(h) = \frac{\gamma(t+h,t)}{\sqrt{\gamma(t+h,t+h)\gamma(t,t)}} = \frac{\gamma(h)}{\gamma(0)}.$$

- ACF > 0: A positive autocorrelation indicates that there is a tendency for the observations to be similar to previous observations.
- Magnitude of Correlation [-1,1]: The magnitude of the ACF value indicates the strength of the correlation. Values closer to 1 (either positive or negative) indicate a stronger correlation, while values closer to 0 indicate a weaker correlation.
- **Decay in Correlation**: In many time series, **the ACF values tend to decrease as the lag increases**. This is often referred to as "decay" in the autocorrelation. The rate of decay and the pattern of ACF values can provide insights into the underlying structure of the time series, such as the presence of seasonality or trends.

➔ **Define autoregressive pattern or leading/lagging average component.**

ACF plot is also useful for identifying non-stationary time series. For a stationary time series, the ACF will drop to zero relatively quickly, while the ACF of non-stationary data decreases slowly. Also, for non-stationary data, the value of r1 is often large and positive.

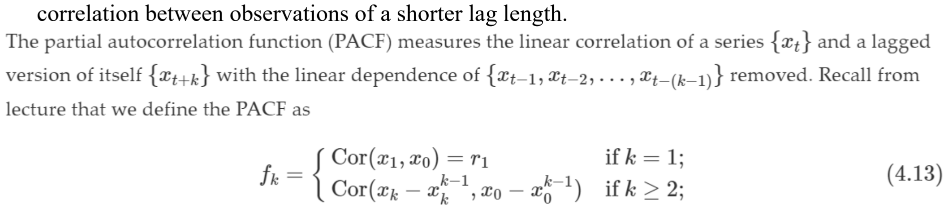
● **Partial Autocorrelation Function - PACF**

➔ # ACF -considers all components in finding correlations of present with lags

➔ # PACF - finds correlations of **residuals** (that remain after removing the effects which are already explained by earlier lags).

> Both the ACF and PACF start with a **lag of 0**, which is the correlation of the time series with itself and therefore results in a **correlation of 1**.

> The difference between ACF and PACF is the inclusion or exclusion of indirect correlations in the calculation. A PACF is similar to an ACF except that each partial correlation controls for any correlation between observations of a shorter lag length.



The partial autocorrelation function (PACF) measures the linear correlation of a series $\{x_t\}$ and a lagged version of itself $\{x_{t+k}\}$ with the linear dependence of $\{x_{t-1}, x_{t-2}, \ldots, x_{t-(k-1)}\}$ removed. Recall from lecture that we define the PACF as

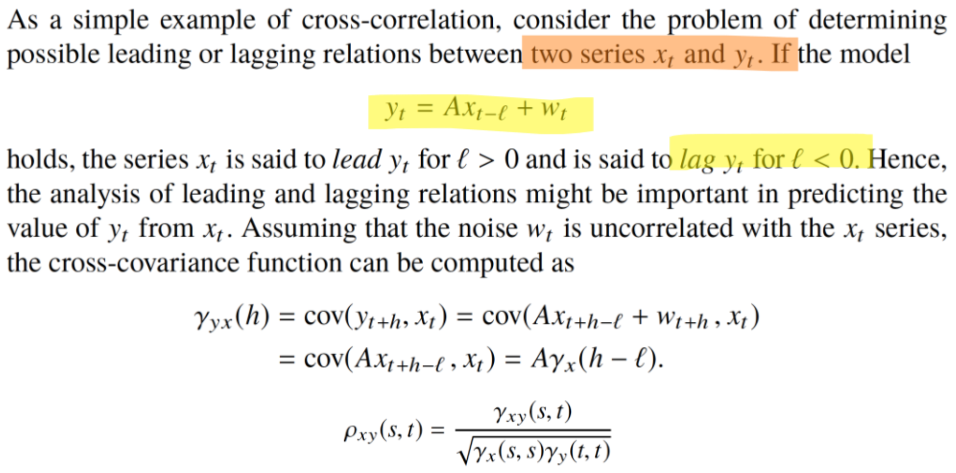
$$f_k = \begin{cases} \mathrm{Cor}(x_1, x_0) = r_1 & \text{if } k = 1; \\ \mathrm{Cor}(x_k - x_k^{k-1}, x_0 - x_0^{k-1}) & \text{if } k \geq 2; \end{cases} \tag{4.13}$$

- **Correlogram**

The sample correlogram is the plot of the ACF against k.

5.2.3. CCF

In Jointly Stationary Time Series



As a simple example of cross-correlation, consider the problem of determining possible leading or lagging relations between two series x_t and y_t. If the model

$$y_t = Ax_{t-\ell} + w_t$$

holds, the series x_t is said to *lead* y_t for $\ell > 0$ and is said to *lag* y_t for $\ell < 0$. Hence, the analysis of leading and lagging relations might be important in predicting the value of y_t from x_t. Assuming that the noise w_t is uncorrelated with the x_t series, the cross-covariance function can be computed as

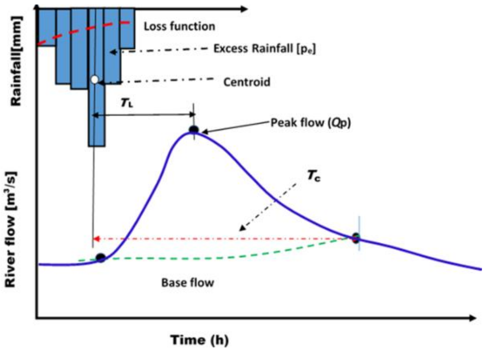
$$\begin{aligned}\gamma_{yx}(h) &= \mathrm{cov}(y_{t+h}, x_t) = \mathrm{cov}(Ax_{t+h-\ell} + w_{t+h}, x_t) \\ &= \mathrm{cov}(Ax_{t+h-\ell}, x_t) = A\gamma_x(h - \ell).\end{aligned}$$

$$\rho_{xy}(s,t) = \frac{\gamma_{xy}(s,t)}{\sqrt{\gamma_x(s,s)\gamma_y(t,t)}}$$

5.2.4. In rainfall-runoff model

- **in direct runoff hydrographs**
- Lag time (TL) and time of concentration (TC) are two measures of **how quickly a stream or flow in a rainwater pipe network responds to runoff-producing rainfall.** These parameters are the main inputs used to estimate peak flow under flood conditions in ungauged watersheds.
- Quantify flash flood response time -> chemical tracers in reality.
- The commonly accepted definitions of TL and TC are derived from direct runoff hydrographs, where TL is the time from the centroid of rainfall excess to peak flow of a direct runoff

hydrograph.



- **in rainwater pipe network**
- The lag time in the rainwater drainage network refers to the temporal delay between the onset of rainfall and the occurrence of a similar trend or pattern in discharge observed at a specific monitoring location.
- It represents the time taken for rainwater to travel through the network, reach the monitoring point, and manifest as an increase in discharge.
- The lag time is influenced by various factors including the hydraulic characteristics of the drainage system, the topography of the area, and the flow velocity within the pipes.

5.3. Steps in TSA - Lagged Regressions

1) **Time Series plot of the series**:
   - Begin by visualizing the time series data to understand its overall pattern and behavior. This initial exploration helps in identifying any obvious trends, seasonality, or irregularities in the data.
2) **Check for the existence of a trend or seasonality**:
   - After plotting the time series, assess whether there is a systematic trend (long-term movement) and/or seasonality (repeating patterns) present in the data. This step helps in understanding the underlying structure of the series.
   - Utilize techniques like decomposition (such as Seasonal Decomposition of Time Series or STL decomposition) to separate the trend and seasonal components from the raw data.
3) **Check for possible outliers**:
   - Identify and examine any data points that deviate significantly from the overall pattern of the time series. Outliers can affect the analysis and modeling process, so it's important to detect and understand their potential impact.
   - Use methods like Boxplot, Z-score, or statistical tests to detect outliers and assess their influence on the analysis.
4) **Check for the sharp changes in behavior**:
   - Look for abrupt changes or shifts in the behavior of the time series. These sudden changes could indicate structural breaks or shifts in underlying dynamics, which may

need to be addressed in the analysis.

5) **Model Identification**:
   - Use statistical methods such as Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots to identify potential models that best capture the autocorrelation structure of the data.
   - Consider different time series models such as Autoregressive Integrated Moving Average (ARIMA), Seasonal ARIMA (SARIMA), or other models based on the characteristics observed in the ACF and PACF plots.
6) **Remove the trend and the seasonal component to get stationary residuals**:
   - If the time series exhibits trend and/or seasonality, it's often necessary to remove these components to achieve stationarity. This involves techniques such as differencing or decomposition to isolate the stationary residuals, which are essential for modeling.
   - Use differencing to remove trend and seasonal components, and ensure that the resulting series is stationary by conducting statistical tests like the Augmented Dickey-Fuller (ADF) test.

Case:

1) The largest correlation versus Q_MD04 occurs in Q_MD02 as 0.979, showing these two data are **very highly correlated.** That means when it rains in this catchment, the discharge in these two monitors shows a closely linear relationship.
2) The plots show that with the lag increases, the correlation decreases, indicating **a weakening dependence over time**. As the lagging time increases, the correlation between the current discharge data and observations at earlier time points decreases, that means the time series becomes less dependent on its own past values.
3) As the lagging time increases, the correlation ccf between Q_MD04 and rainfall decreases, but it is still obviously higher than their autocorrelation acf, showing that hat changes in rainfall have a **delayed impact** on Q_MD04.
4) The higher cross-correlation compared to autocorrelation suggests that the relationship between the two variables **is not instantaneous but exhibits a time lag**, that means there is often a time delay between precipitation events and their impact on rainfall-runoff discharge. The delayed cross-correlation may reflect the time it takes for rainfall to be translated into changes in runoff.
5) Plots show the similar decreasing trend as increasing kernel smoothing bandwidths. And the obviouse smoothing results occur in rainfall as the long-term precipitation and small rainfall. Smoothing data **decreases sensitivity to short-term variations**, thus in the longer-term pattern, there is a strong dependency between Rainfall and Q_MD04 showing the similar changing trends.
6) The resulting plot shows the higher cross-correlation occurs in lagging time as 5mins(0.91) and 6mins(0.91), considering the corresponding autocorrelation, the lag h=5mins is more suitable, as the highest autocorrelation(0.91) and cross-correlation(0.91). Within lag h=5mins, it indicates a strong correlation between the current discharge Q_MD04 and the rainfall 5 minutes ago, that means the **rainfall-runoff decay time is about 5mins** and the system between rainfall and Q_MD04 needs 5 mins to adapt.

5.4. AIC Corrected & Bayesian InformaEon Criterion (BIC) – Evaluate ARMA

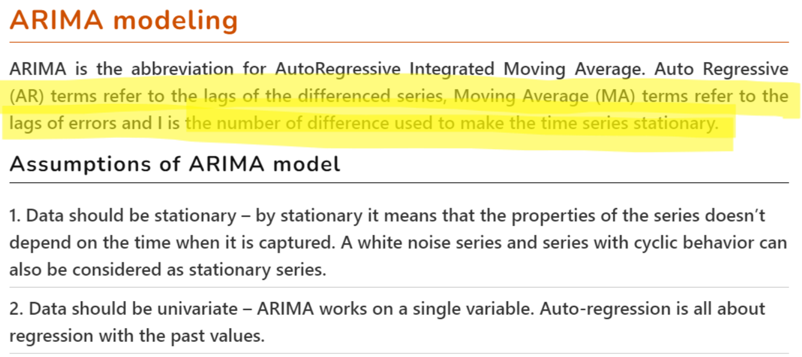
They provide a trade-off between the goodness of fit of the model and its complexity, helping to balance model accuracy with parsimony.

AIC	- A measure of the relative quality of a statistical model for a given set of data. - It quantifies the balance between the goodness of fit of the model and its complexity, penalizing models that are overly complex.	**Lower** AIC or BIC values indicate better-fitting models, with lower complexity. the model with the lowest AIC value is generally preferred.
BIC	- A similar measure to AIC but places a higher penalty on model complexity. - Like AIC, BIC balances model fit and complexity, but it tends to favor simpler models more strongly than AIC.	

# check the value of AIC (BIC) to define the fitting of model
# AIC is smaller, it will be better and more fitting

5.5. Forecasting using ARMA

Commented [XP7]: note



ARIMA modeling

ARIMA is the abbreviation for AutoRegressive Integrated Moving Average. Auto Regressive (AR) terms refer to the lags of the differenced series, Moving Average (MA) terms refer to the lags of errors and I is the number of difference used to make the time series stationary.

Assumptions of ARIMA model

1. Data should be stationary – by stationary it means that the properties of the series doesn't depend on the time when it is captured. A white noise series and series with cyclic behavior can also be considered as stationary series.

2. Data should be univariate – ARIMA works on a single variable. Auto-regression is all about regression with the past values.

Steps:

5.5.1. EDA Provides the p,d,q estimate for ARIMA models.

#1. Exploratory analysis:
#a. Autocorrelation
#-Used to estimate which value in the past has a correlation with the current value
#-provides p,d,q estimate for ARIMA models
#-Eg: ARIMA(1, 0, 12) - describing some response variable by combining a 1st order Auto-Regressive model and a 12th order Moving Average model
#-(Auto-Regressive Parameters) + (Integrative Part/Differencing) + (Moving Average Parameters)
#-look at an autocorrelation graph of the data (will help if Moving Average (MA) model is appropriate) - q
#-look at a partial autocorrelation graph of the data (will help if AutoRegressive (AR) model is appropriate) -

p
#-look at extended autocorrelation chart of the data (will help if a combination of AR and MA are needed)

#b.try Akaike's Information Criterion (AIC) on a set of models and investigate the models with the lowest AIC values
#c.try the Schwartz Bayesian Information Criterion (BIC) and investigate the models with the lowest BIC values

#d.Spectral analysis to examine cyclic behavior
#-describe how variation in a time series may be accounted for by cyclic components
#-Using this, periodic components in a noisy environment can be separated out

#e.Trend estimation and decomposition
#-It seeks to construct, from an observed time series, a number of component series, where each has a certain characteristics

# 2. Fit the model
# 3. Diagnostic measures

5.5.2. achieve data stationarity

remove non-stationary part for ARIMA and achieve data stationarity, using difference: diff().
To remove seasonality from the data, we subtract the seasonal component from the original series and then difference it to make it stationary.
To achieve stationarity:

1) #a. difference the data,
2) #b. log or square root the series data to stabilize non-constant variance,
3) #c. if the data contains a trend, fit some type of curve to the data and then model the residuals from that fit,
4) #d. Unit root test – This test is used to find out that first difference or regression which should be used on the trending data to make it stationary.
5) #e. In Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test, small p-values suggest differencing is required.

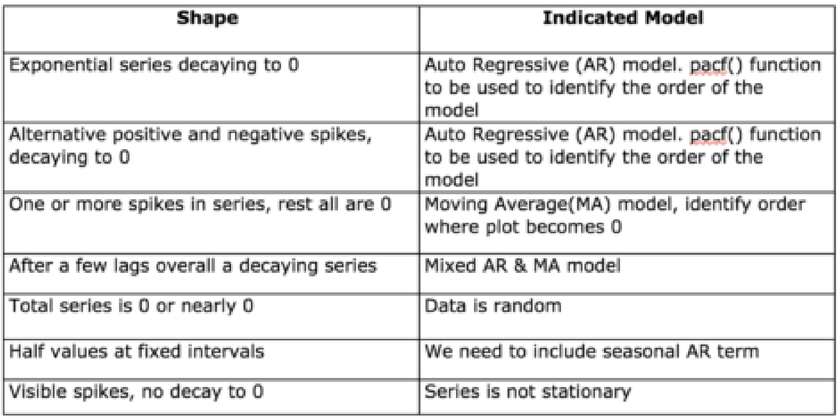
5.5.3. Smoothing

Smoothing is usually done to help us better see patterns, trends in time series. Generally it smooths out the irregular roughness to see a clearer signal.

5.5.4. Diagnostic measures

Examine which p and q values will be appropriate we need to run acf() and pacf() function.
#ACF at lag 1 of 24 is -ve and slightly smaller than -0.4. **If ACF falls below -0.5, then series is over differenced**.
# Positive spikes becoming negative is a sign of possible over differencing and the over differencing

is to be compensated by a MA term.



Shape	Indicated Model
Exponential series decaying to 0	Auto Regressive (AR) model. pacf() function to be used to identify the order of the model
Alternative positive and negative spikes, decaying to 0	Auto Regressive (AR) model. pacf() function to be used to identify the order of the model
One or more spikes in series, rest all are 0	Moving Average(MA) model, identify order where plot becomes 0
After a few lags overall a decaying series	Mixed AR & MA model
Total series is 0 or nearly 0	Data is random
Half values at fixed intervals	We need to include seasonal AR term
Visible spikes, no decay to 0	Series is not stationary

the middle terms I as "diff()", mark the variables difference order, instead of the order or setting to deal with the variables in this process

5.5.5. limitations in long-tern projection

#Long-term forecasts using ARIMA/SARIMA should be avoided.
#For long-term prediction, it would be good to also draw on other sources of information during the model selection process
#and/or use an auxiliary model and take into account the factors such as economic conditions, average income etc. in this case.
#We could try fitting time series models that allow for inclusion of other predictors
#using methods such ARMAX or dynamic regression.

6. Computer Deterministic Models

6.1. Intro Process - Modelling

Application of ICT (Information and Communications Technology) to address the problems.

1) Data: Databases, GIS, Data analysis methods;
2) Models: Deterministic and Data-driven
3) **Predominant issues**:

- Physical-Scale Model: Model scale; Time; Cost (slow and expensive); Large temporal/areal studies.
- Numerical Model: Scale Issues; Breadth of application.

● **Need for computer model**:

- Schemes (models): Typically, differential equations of conservation
- Algorithms
- Boundary and initial conditions

- Data
- Calibration and Verification

- **Hydrological process**
- Water **balance mass** components:

$$P = ET_a + Q + \Delta S + \Delta G$$

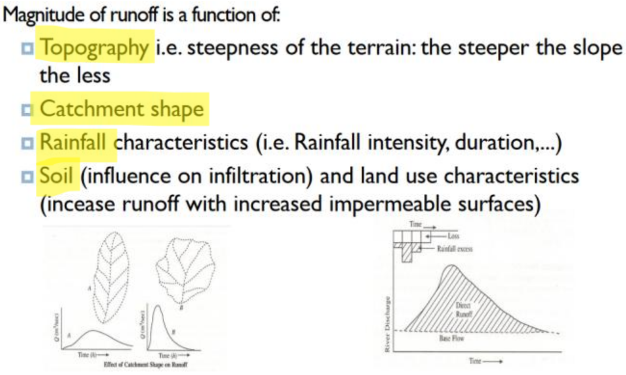
with Q= Runoff + Interflow + Baseflow

Evaporation + Runoff + interflow + baseflow

- Translate the hydrological cycle into **discrete** processes:

Precipitation = Infiltration + Evaporation/Transpiration + Rainfall - Runoff + Stream/Channel Flow

- Prediction of peak runoff rates



Magnitude of runoff is a function of:

- □ Topography i.e. steepness of the terrain: the steeper the slope the less
- □ Catchment shape
- □ Rainfall characteristics (i.e. Rainfall intensity, duration,...)
- □ Soil (influence on infiltration) and land use characteristics (incease runoff with increased impermeable surfaces)

Rational method: Q (m3/s) = Ci (mm/hr) * A (m2)

6.2. Numerical Solution

Components	linearized equations; equation solvers
Practical issues	Efficiency, accuracy, stability, robustness, conservation
Mathematical issues	Consistency: Truncation error~0 Stability: Scheme is considered stable when errors from solution process are not magnified in the solution process. Convergence:
General Issues - contradiction	Conservation; Boundedness -BCs; Realizability

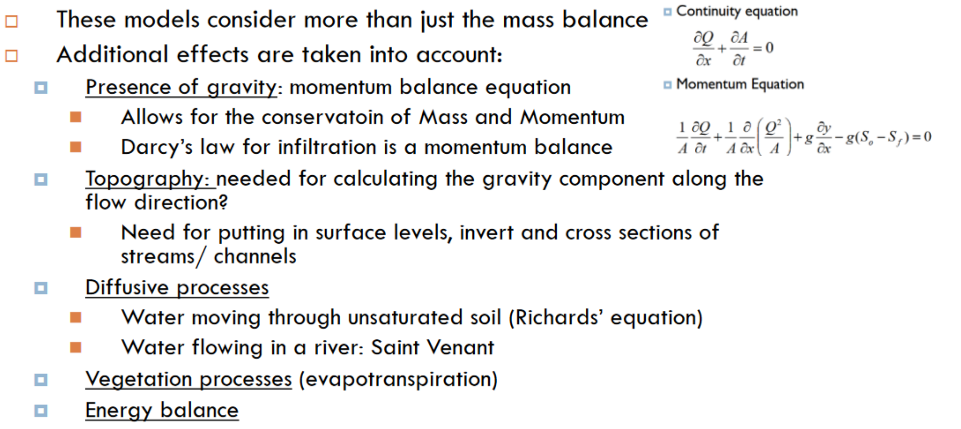
Methods	
Finite Difference Method	
Finite Volume Method	
Finite Element Method	

- **Hydrological modelling**

Physical Law: mass balance:

dV/dt = Qin – Qout (as black box)

Fully Distributed model:



- These models consider more than just the mass balance
- Additional effects are taken into account:
  - Presence of gravity: momentum balance equation
    - Allows for the conservatoin of Mass and Momentum
    - Darcy's law for infiltration is a momentum balance
  - Topography: needed for calculating the gravity component along the flow direction?
    - Need for putting in surface levels, invert and cross sections of streams/ channels
  - Diffusive processes
    - Water moving through unsaturated soil (Richards' equation)
    - Water flowing in a river: Saint Venant
  - Vegetation processes (evapotranspiration)
  - Energy balance

- Continuity equation

$$\frac{\partial Q}{\partial x}+\frac{\partial A}{\partial t}=0$$

- Momentum Equation

$$\frac{1}{A}\frac{\partial Q}{\partial t}+\frac{1}{A}\frac{\partial}{\partial x}\left(\frac{Q^2}{A}\right)+g\frac{\partial y}{\partial x}-g(S_o-S_f)=0$$

Advantages	physical processes are captured in the best possible manner
	Future scenarios can be modeled (e.g. climate change) because the model is not calibrated statically
Disadvantages	Highly detailed information needed
	Many parameters (soil, vegetation, channel flow etc.)
	Parameter uncertainty issues (see e.g. GLUE procedure)

- **1D modelling**
- The 1D model must be orientated along the river axis.
- Processes that take place in the two remaining space-dimensions (vertical and transversal) are either parameterized or ignored.

- Most lD hydraulic model computational schemes are based on the implicit time-marching schemes rather than the explicit schemes. -> **iterations**

Commented [XP8]: Note. Generally applied into iteration methods

1) **Stability:** Implicit schemes tend to be more numerically stable than explicit schemes, especially for stiff systems of equations that arise in hydraulic modeling. Implicit schemes allow for larger time steps and better stability properties, reducing the risk of numerical instability and oscillations in the solution.
2) **Convergence:** Implicit schemes often converge more quickly and reliably than explicit schemes, particularly for highly nonlinear systems or when modeling complex hydraulic phenomena. Implicit schemes facilitate the solution of large systems of equations by iteratively solving linearized equations at each time step, leading to faster convergence and robust numerical solutions.
3) **Flexibility:** Implicit schemes provide greater flexibility in choosing time step sizes and integration methods, allowing for adaptive time stepping and efficient solution strategies. This flexibility is particularly valuable when modeling hydraulic systems with variable flow

conditions or complex geometries, where adaptive time stepping can improve computational efficiency and solution accuracy.

4) **Handling Nonlinearities:** Implicit schemes are well-suited for handling nonlinearities inherent in hydraulic modeling, such as frictional losses, flow resistance, and nonlinear boundary conditions. Implicit schemes can effectively handle these nonlinearities by iteratively updating the solution at each time step, ensuring accurate representation of complex hydraulic processes.

6.3. Deterministic model

6.3.1. Definition

Deterministic models assume that the process described by the model is free from random variation. While natural variation exists in reality, deterministic models provide estimates by describing system behavior based on well-defined mathematical equations and physical principles.

Inverse: Stochastic model.

A deterministic model is a mathematical model in which the output is determined only by the specified values of the input data and the initial conditions. This means that a given set of input data will always generate the same output.
e.g. MIKE, SWAH, HEC-HMS…
Deterministic models essentially ignore variation in input by assuming fixed input. The input may be changed for different scenarios or historical periods, but the input still takes on a single value. Such an assumption may seem too significant for the resulting model to produce meaningful results. However, **deterministic models nevertheless are valuable tools because of the difficulty of characterizing watersheds and the hydrologic environment in the first place.**

6.3.2. Necessity

- Compared with the obvious scale issues in physical or numerical model, and the expensive cost, deterministic model can provide the following conveniences:

1) **Understanding System Dynamics**: By incorporating known physical laws and relationships, deterministic models help in understanding the dynamics of hydrological systems.
2) **Validation and Calibration**: Deterministic models can be validated and calibrated using historical data, which enhances their accuracy and reliability. This validation process ensures that the model reflects the actual system behavior.
3) **Long-Term Planning/Projection**: Deterministic models are valuable for long-term planning as they can simulate various scenarios and assess the potential outcomes under different conditions. This capability is essential for sustainable water resource management and adaptation to changing environmental conditions.
4) Deterministic models are useful in predicting outcomes in situations where the parameters are known with certainty.

The scaling problem is considered to be fundamental as this is the main source of uncertainties introduced by modelling. In general, it could be summarized in the following main statements:

- Parameters of macro scale models are generalized parameters at the micro scale.
- Relationships and equations are different for different scales. ž Equation parameters are different

at different scales.

- A universal scaling methodology, allowing transition from one set of scale parameters to any other, is highly desired and still undeveloped.

➔ the scale problem, which is related mainly to methods of mathematical description of water movement from the places of runoff formation to the basin outlet, has been minimized.

6.3.3. Limitation

Deterministic models assume known average rates with no random deviations.
Refer to modelling by MIKE.
lack of systematic data -> higher Spatial and Temporal Resolution required; topographic data.
Quality of measurement data
Uncertainty – uncertainties in input data, model parameters, and boundary conditions can propagate through the model simulations, affecting the reliability of the results.

6.3.4. Important considerations as modelling

1. **Hydrological Processes**: Capture the key hydrological processes relevant to tropical urban areas, including rainfall interception, infiltration, surface runoff, and groundwater recharge. Ensure the model accounts for the complex interactions between these processes, considering factors such as soil type, land cover, and vegetation characteristics.
2. **Urban Infrastructure**: Incorporate detailed representations of urban drainage infrastructure, such as stormwater drainage networks, sewers, culverts, and retention/detention ponds. Include information on pipe sizes, flow capacities, storage volumes, and operational rules to accurately simulate the behavior of the drainage system during rainfall events.
3. **Land Use and Land Cover**: Use high-resolution land use and land cover data to characterize the urban environment and its impact on hydrological processes. Differentiate between impervious and pervious surfaces, considering variations in runoff coefficients, infiltration rates, and surface roughness parameters across land use categories.
4. **Rainfall Data**: Utilize high-quality rainfall data with fine temporal and spatial resolution to represent the intense and variable rainfall patterns characteristic of tropical regions. Incorporate radar data, rain gauge observations, or satellite-derived precipitation estimates to capture spatial variability and storm dynamics accurately.
5. **Model Calibration and Validation**: Calibrate the model using observed hydrological data, such as streamflow measurements, rainfall records, and water level observations, to ensure that simulated outputs match observed behavior. Validate the model against independent datasets and historical events to assess its performance under different conditions and across various spatial and temporal scales.
6. **Climate Change Considerations**: **Tropical regions are vulnerable** to the impacts of climate change, including changes in rainfall patterns, increased frequency of extreme weather events, and rising sea levels. Incorporate projections of future climate change scenarios to assess the potential impacts on rainfall-runoff processes in the catchment. Use climate models or downscaled climate projections to simulate future precipitation patterns, temperature changes, and extreme weather events, and evaluate their implications for flood risk and water resources management.

7. Data Assimilation & Data driven

7.1. Intro

Data Assimilation combines observations into a dynamical model, using the model's equations to provide time continuity and coupling between the estimated fields.

1) **Improving Model Accuracy:** Hydrologic data assimilation aims to utilize both our hydrologic process knowledge, as embodied in a hydrologic model, and information that can be gained from observations. Both model predictions and observations are imperfect and we wish to use both synergistically to obtain a more accurate result. Moreover, both contain different kinds of information, that when used together, provide an accuracy level that cannot be obtained individually.
2) **Enhancing Model Predictability:** Incorporating observational data into numerical models increases their skill in predicting future states of the system. Data assimilation techniques help extract useful information from observations and assimilate it into the model, thereby improving the model's ability to capture the underlying dynamics and variability of the system.

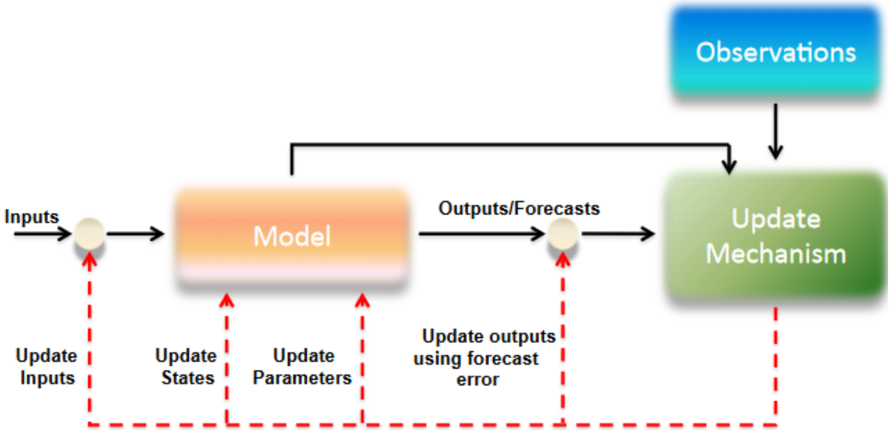
Optimal blending within a lower cost.

The insertion of the reliable data into the dynamical model to improve the quality and accuracy of the estimate.

One of the main purposes of concurrent use of monitoring and modeling systems is to combine data and results of numerical models in data assimilation sense of the word.

7.2. Approaches

1) Update model inputs (address input uncertainties);
2) Update model state variables (Eg. Kalman filtering);
3) Update model parameters;
4) Update output variables (Error forecasting).



7.3. 3 methods / types

- Variational methods
  - Statistical minimisation of cost function that measure differences between simulations and observations conditioned on model dynamics
    - ➢ Optimisation algorithms
    - ➢ Adjoint method (strong/weak constraint)
- Sequential methods
  - Updating of system state during a forward model integration according to error correlation structure (on-line assimilation)
    - ➢ Fixed error correlation structure: e.g. optimal interpolation, nudging
    - ➢ Dynamical evolution of error correlation structure: Kalman filter
- Error correction
  - Correction of model simulation with a forecast of the model error in measurement locations
    - ➢ Time series analysis methods (e.g. ARMAX, local linear models, ANN, GP)

Statistical methods are efficient for simple problems, while variational methods are more suitable for complex systems but require more computational resources and prior information.

Method	Description	Advantages	Disadvantages
Statistical /Sequential Methods (e.g., Kalman Filter)	Update the model state based on the previous estimate and observations. Observations are used as soon as they are available to correct the present state of the model. In contrast to variational methods, sequential methods lead to discontinuities in the time series of the corrected state.	- This approach is more suitable in situations where the system is driven by boundary conditions. - Effective for handling linear and Gaussian problems.	- - Limited applicability to nonlinear and non-Gaussian problems.
Variational Methods (e.g., 3D-Var)	Seek to estimate the model state by minimizing a cost function that incorporates both model predictions and observational constraints. The past observations, from the start of the modelling until the present time, are used simultaneously to correct the initial conditions of the model and obtain the best overall fit of the state to the observations.	- Can handle long time horizons and high-dimensional systems with complex dynamics effectively. - Provides accurate results. - Suitable for the behavior of the system is driven by the accuracy of the initial conditions.	- Requires prior information about the system and its error covariances, which may not always be available. - High computational cost.
Error Correction (e.g., Artificial Neural Networks)	They adjust model outputs based on the discrepancies between model simulations and observations, effectively correcting model biases and errors.	- Can capture complex relationships between model states and observations. - Adaptive and self-learning, capable of improving performance over time.	- Requires large training data to train effectively. - May suffer from overfitting if the training dataset is not representative of the underlying system dynamics.

7.4. Kalman filtering

The Kalman filter is the most well-known data assimilation technique. Its popularity is due to the simplicity of its implementation and its relative robustness to the misspecification of the error sources.

The Kalman filter is a Best Linear Unbiased Estimate (BLUE). It means that it is optimal in situations where the model is linear. It is a minimum variance estimate, i.e. given an observation and a model forecast, it provides the estimate that minimizes the estimation variance. This property does not require any assumption about the distribution of the model error, just that the error is zero-mean uncorrelated in time. The meaning of the variance in case of non-Gaussian distributions, and especially in case of skewed distributions, needs to be specified in each case.

In DA process:

1. **Update Step**: Incorporate observations into the model predictions to update the state estimate using the Kalman Filter equations. This update step involves calculating the **Kalman Gain K**, which determines the weight given to the model prediction and the observations when updating the state estimate.
2. **State Correction**: Adjust the state estimate based on the observations and the Kalman Gain. This correction step improves the accuracy of the state estimate by incorporating information from both the model predictions and the observations.
3. **Covariance Update**: Update the covariance matrix to account for the information gained from the observations. This covariance update reflects the uncertainty in the state estimate and how it changes over time due to the assimilation process.
4. **Repeat Steps**: Iterate the prediction and update steps over the entire assimilation period or until the desired level of accuracy is achieved.

Updated estimate: linear combination

$$x_1^a = x_1^f + K(z_1 - Cx_1^f)$$

Updated variance

$$P_1^a = E\{(x_1 - x_1^a)^2\} = (1 - CK)^2 P_1^f + CK^2$$

8. Machine Learning and Neural Networks

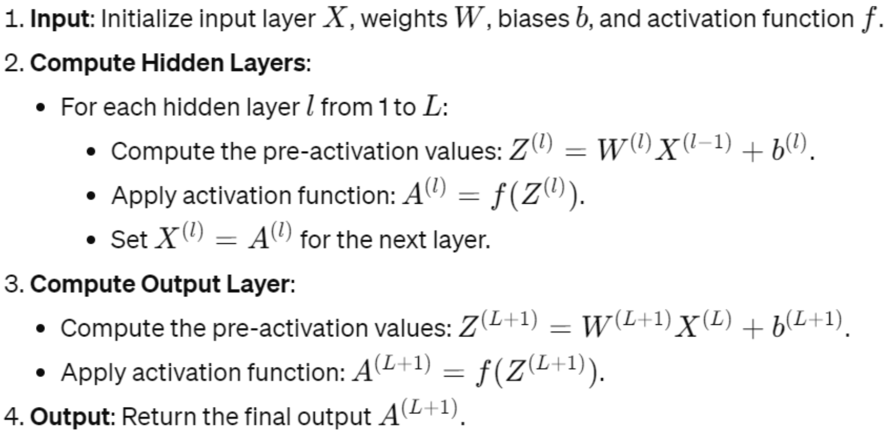
8.1. Data mining

Compared to database (simply analyse), data mining is based on the classification, clustering and association rules.
when something we cannot use the math to express, but can use the statistical listing or similar function expand.grid() to list all probabilities -> **prediction**

Data mining is a process that examines large preexisting databases to generate new information.

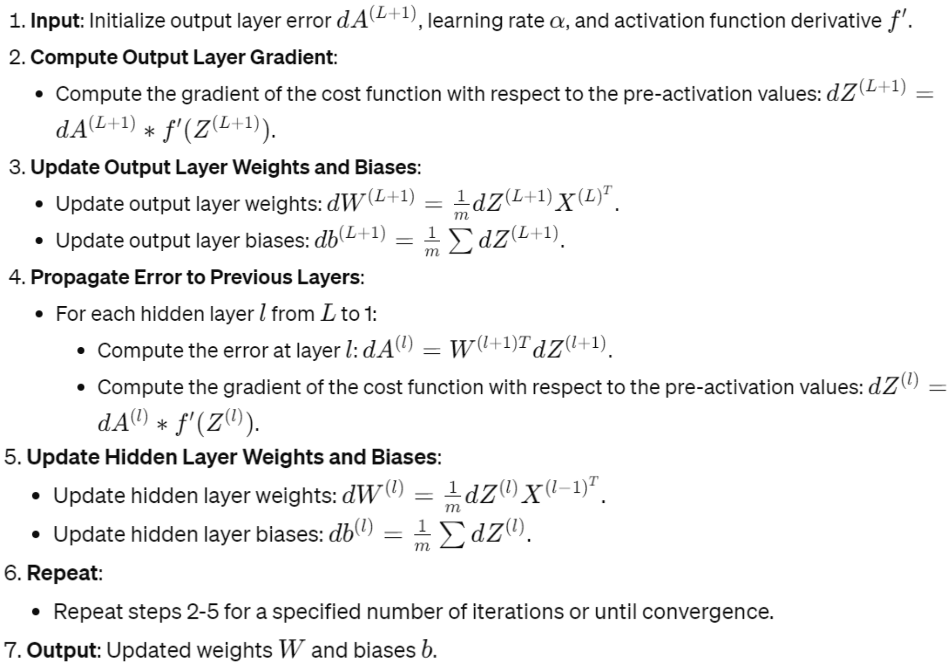
8.2. Pseudo code

- Forward Propagation:



```
1. Input: Initialize input layer $X$, weights $W$, biases $b$, and activation function $f$.
2. Compute Hidden Layers:
    • For each hidden layer $l$ from 1 to $L$:
        • Compute the pre-activation values: $Z^{(l)} = W^{(l)} X^{(l-1)} + b^{(l)}$.
        • Apply activation function: $A^{(l)} = f(Z^{(l)})$.
        • Set $X^{(l)} = A^{(l)}$ for the next layer.
3. Compute Output Layer:
    • Compute the pre-activation values: $Z^{(L+1)} = W^{(L+1)} X^{(L)} + b^{(L+1)}$.
    • Apply activation function: $A^{(L+1)} = f(Z^{(L+1)})$.
4. Output: Return the final output $A^{(L+1)}$.
```

- Back Propagation:



```
1. Input: Initialize output layer error $dA^{(L+1)}$, learning rate $\alpha$, and activation function derivative $f'$.
2. Compute Output Layer Gradient:
    • Compute the gradient of the cost function with respect to the pre-activation values: $dZ^{(L+1)} = dA^{(L+1)} * f'(Z^{(L+1)})$.
3. Update Output Layer Weights and Biases:
    • Update output layer weights: $dW^{(L+1)} = \frac{1}{m} dZ^{(L+1)} X^{(L)^T}$.
    • Update output layer biases: $db^{(L+1)} = \frac{1}{m} \sum dZ^{(L+1)}$.
4. Propagate Error to Previous Layers:
    • For each hidden layer $l$ from $L$ to 1:
        • Compute the error at layer $l$: $dA^{(l)} = W^{(l+1)T} dZ^{(l+1)}$.
        • Compute the gradient of the cost function with respect to the pre-activation values: $dZ^{(l)} = dA^{(l)} * f'(Z^{(l)})$.
5. Update Hidden Layer Weights and Biases:
    • Update hidden layer weights: $dW^{(l)} = \frac{1}{m} dZ^{(l)} X^{(l-1)^T}$.
    • Update hidden layer biases: $db^{(l)} = \frac{1}{m} \sum dZ^{(l)}$.
6. Repeat:
    • Repeat steps 2-5 for a specified number of iterations or until convergence.
7. Output: Updated weights $W$ and biases $b$.
```

1) Optimize and adjust the two parameters, w and b, at the same time: According to the **gradient descent** algorithm, we need to calculate the gradient value of the loss function for w and b, that is, the partial derivative. Both parameters are then updated along the opposite direction of the gradient.
2) better find the partial derivative of L with respect to w and b: Using the chain rule for derivatives, find the partial derivative of L with respect to y.

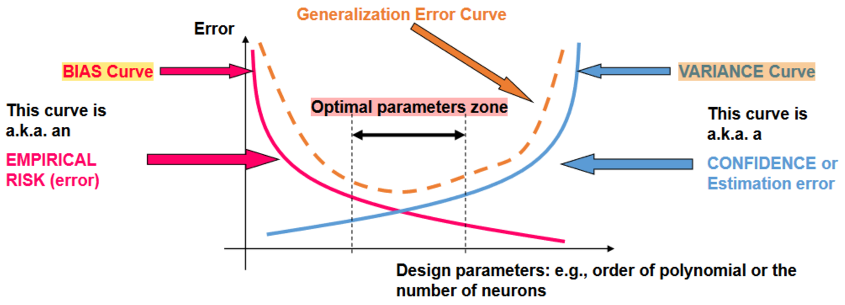
- **Difference**

ANNs are best suited for problems with a fixed-sized input and output, while RNNs are designed to handle sequential data. ANNs use fully connected layers to capture non-linear relationships between input and output variables, while RNNs use feedback loops to capture information from previous time steps.

8.3. Dataset Split Percentage

Subset	Purpose	Key Activities	Outcome
Training Set	Used to train the machine learning model, enabling it to learn patterns in the data.	- Model Training - Parameter Optimization	Model captures underlying patterns, ready for making predictions on unseen data.
Validation Set	Used to provide an unbiased evaluation of a model fit on the training data set while tuning the model's hyperparameters	- Hyperparameter Tuning - Early Stopping	Optimal hyperparameters selected, preventing overfitting, improving generalization.
Test Set	Assesses the final performance (i.e. generalization) of the trained model on unseen data.	- Model Evaluation - Performance Assessment	Provides unbiased estimate of model's ability to generalize, informs decision-making for model deployment.

8.4. Bias-variance and over fitting


Generalization Error Curve
Error
BIAS Curve
VARIANCE Curve
This curve is a.k.a. an
EMPIRICAL RISK (error)
Optimal parameters zone
This curve is a.k.a. a
CONFIDENCE or Estimation error
Design parameters: e.g., order of polynomial or the number of neurons

- **Over fitting:**

the perfect training results, but very bad generalization ones.

When machine learning algorithms are constructed, they leverage a sample dataset to train the model. However, when the model trains for too long on sample data or when the model is too complex, it can start to learn the "noise," or irrelevant information, within the dataset. When the model memorizes the noise and fits too closely to the training set, the model becomes "overfitted," and it is unable to generalize well to new data.

- **Early stopping:** As we mentioned earlier, this method seeks to pause training before the model starts learning the noise within the model. This approach risks halting the training process too soon, leading to the opposite problem of underfitting. Finding the "sweet spot" between underfitting and overfitting is the ultimate goal here.
- **Train with more data:** Expanding the training set to include more data can increase the accuracy of the model by providing more opportunities to parse out the dominant relationship among the input and output variables. That said, this is a more effective method when clean, relevant data is injected into the model. Otherwise, you could just continue to add more complexity to the model, causing it to overfit.
- **Data augmentation:** While it is better to inject clean, relevant data into your training data, sometimes noisy data is added to make a model more stable. However, this method should be done sparingly.
- **Feature selection:** When you build a model, you'll have a number of parameters or features that are used to predict a given outcome, but many times, these features can be redundant to others. Feature selection is the process of identifying the most important ones within the training data and then eliminating the irrelevant or redundant ones. This is commonly mistaken for dimensionality reduction, but it is different. However, both methods help to simplify your model to establish the dominant trend in the data.
- **Regularization:** If overfitting occurs when a model is too complex, it makes sense for us to reduce the number of features. But what if we don't know which inputs to eliminate during the feature selection process? If we don't know which features to remove from our model, regularization methods can be particularly helpful. Regularization applies a "penalty" to the input parameters with the larger coefficients, which subsequently limits the amount of variance in the model. While there are a number of regularization methods, such as lasso regularization, ridge regression and dropout, they all seek to identify and reduce the noise within the data.
- **Ensemble methods:** Ensemble learning methods are made up of a set of classifiers—e.g. decision trees—and their predictions are aggregated to identify the most popular result. The most well-known ensemble methods are bagging and boosting. In bagging, a random sample of data in a training set is selected with replacement—meaning that the individual data points can be chosen more than once. After several data samples are generated, these models are then trained independently, and depending on the type of task—i.e. regression or classification—the average or majority of those predictions yield a more accurate estimate. This is commonly used to reduce variance within a noisy dataset.

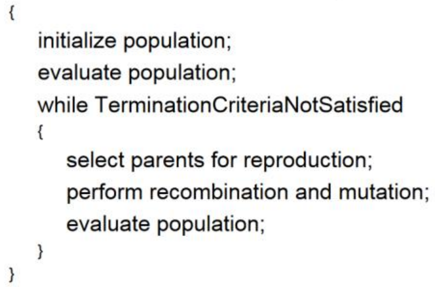
- **Bias**

Bias: A difference between the model output and unknown target function.

model's bias is a measure of how well we can model the underlying unknown function with some function from hypothesis space H.
Good model designer would try to **get medium** both the Bias and the Variance.

9. GA - Genetic Algorithms - Evolutionary Algorithms

Good-fit while tight interaction between domain and method knowledge is secured.



```
{
    initialize population;
    evaluate population;
    while TerminationCriteriaNotSatisfied
    {
        select parents for reproduction;
        perform recombination and mutation;
        evaluate population;
    }
}
```

The genetic algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution. The genetic algorithm repeatedly modifies a population of individual solutions.
At each step, the genetic algorithm selects individuals from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the **population "evolves" toward an optimal solution.**
You can apply the genetic algorithm to solve a variety of optimization problems that are not well suited for standard optimization algorithms, including problems in which the objective function is discontinuous, nondifferentiable, stochastic, or highly nonlinear. The genetic algorithm can address problems of mixed integer programming, where some components are restricted to be integer-valued.

10. Automatic calibration

- Calibration objectives:

1) A good agreement between average simulated and recorded catchment runoff (good water balance)
2) A good overall agreement of the shape of the hydrograph
3) A good agreement of peak flows
4) A good agreement for low flows

Trade-off

11. Assignment

11.1.Data Handling and Analysis, Visualisation and Descriptive Statistics

```
# assign the specific value in rows or columns to a new data frame
my_data_new<-my_data[c(11:50),c(1:9)]

# create a new vector to save the specific sets splitted out
theta<-my_data_new$theta # define the column as a new vector
percentage_2<-length(theta[theta>1.5])/length(theta); percentage_2

# mean values for each column
column_means <-colMeans(my_data, na.rm = TRUE);
# calculate mean for the specific column
mean_theta<-mean(my_data[,1]);

# variance
column_var <- colVars(my_data, na.rm = TRUE);
# or use var() for every column
var(my_data[,1])

# standard deviation
column_sds <- colSds(my_data, na.rm = TRUE);column_sds
sd(my_data[,1])   # for single column

# interquartile range
IQR(my_data[,1])  # for single column

# skewness
column_skew <- colSkewness(my_data, na.rm = TRUE);column_skew
skewness(my_data[,1])   # for single column

# kurtosis
column_kurt <- colKurtosis(my_data, na.rm = TRUE);column_kurt
kurtosis(my_data[,1])   # for single column

# or use lapply() for all columns in df
lapply(my_data,var,na.rm = TRUE)
lapply(my_data,sd,na.rm = TRUE)
lapply(my_data,IQR,na.rm = TRUE)
lapply(my_data,skewness,na.rm = TRUE)
lapply(my_data,kurtosis,na.rm = TRUE)

# ggplot()
# geom_point() + geom_line(); x or y axis can break by specific value
ggplot(df_Q3, aes(x=x, y=y)) + geom_point() + geom_line()+
  scale_x_datetime(breaks=date_breaks("4 months"),labels=date_format("%Y-%m-%d %H:%M:%S"))+
  ggtitle('Time series of Residual') + xlab('Date') + ylab('Residual')

ggplot2.multiplot(f1, f2, f3, cols=1)
```

11.2.Exploratory data analysis (EDA) – Data pre-processing

```
# 1) min-max method
min_max <- function(x){
  y<-(x - min(x, na.rm=TRUE))/(max(x,na.rm=TRUE) - min(x, na.rm=TRUE))
  return (y)
}
min_max1 <- as.data.frame(lapply(sediment, min_max))

#Normalize:
normalize <- function(x) {
    return ((x - min(x)) / (max(x) - min(x)))
}
ddnorm <- as.data.frame(lapply(dd,normalize))

#Denormalize:
minvec <- sapply(dd,min)
maxvec <- sapply(dd,max)
denormalize <- function(x,minval,maxval) {
    x*(maxval-minval) + minval
}
as.data.frame(Map(denormalize,ddnorm,minvec,maxvec))

# 2) Z score method;
Z_score<-function(x){
  y<-(x - mean(x, na.rm=TRUE))/(sd(x,na.rm = TRUE))
  return(y)
}
z_score2<-as.data.frame(lapply(sediment,Z_score))

# 3) decimal normalization method - decimal scaling
# we need to define a function to round up a number to the nearest power of 10
roundUp <- function(x) 10^ceiling(log10(x))
Decimal_norm<-function(x){
  y<-x/roundUp(max(x)) # say, the maximum number of x is 87, we round it up to the nearest power of 10, which is 100.
  return(y)            # then, for decimal normalization, all the values of x should be divided by 100
}
decimal_nor3<-as.data.frame(lapply(sediment,Decimal_norm))
```

# smooth with Loess using **scatter.smooth()**: Plot and add a smooth curve computed by loess to a scatter plot.

```
scatter.smooth(theta1,theta_p1, span = 0.5, degree = 1,main='scatter plot of theta versus theta_p',xlab='theta',ylab='theta_p',
            family = c("symmetric", "gaussian"), evaluation = 50,col=2)
```

11.3.Linear Regression

```
# retrieve the value of correlation
# Calculate the correlation matrix using Pearson’s
cor_matrix <- cor(sediment, method = "pearson")

or
ggpairs(sediment, corMethod = "pearson")
```

11.3.1. SLR

```
# M1: the simple regression model can be built using lm(y~x).
# y is the dependent variable, x is the independent variable or predictor
# theta ~ theta_p
library(UsingR)
model <- lm(theta ~ theta_p, data = sediment)
lm(theta ~ theta_p, data = sediment)
# Summary of the regression model
summary(model)

# M2: using simple.lm(x,y) function
lm.result<-simple.lm(theta_p,theta) # the result of simple.lm(x,y) is of class lm, so the plot and summary commands adapt themselves to that
# if you want to plot confidence interval, set show.ci=TRUE
simple.lm(x=theta_p,y=theta, show.ci=TRUE)
summary(lm.result)
```

11.3.2. MLR

```
# Creating a linear model with three predictors
threePredictorModel <- lm(theta ~ theta_p + uf + uf_p, data=df_training)
```

11.4.EDA - TS

Estimate the regression coefficients using least square method, and list the results of the regression coefficients.

```
lag_time<- -5
Rainfall<-lag(Rainfall,lag_time)
# Construct a linear regression by the least square method
Rain = Rainfall-mean(Rainfall) # center rainfall: subtract the mean to eliminate the overall shift in the data
Rain2 = Rain^2
trend = time(Q_MD04) # time
fit = lm(Q_MD04 ~ trend + Rain + Rain2 , na.action=NULL)
```

11.5.ANNs