Aufgabestellung

Die Hauptziele der Projektarbeit sind

- **sich einen Überblick über die Merkmale (unabhängige Variable) und Ausgabewert (abhängige Variable) mittels Jupyter Notebook zu verschaffen,**
- **den Datensatz je nach Bedarf zu manipulieren,**
- **Prognosemodelle zu erstellen und**
- **die Prognosegüte der Modelle rechnerisch und visuell zu beurteilen.**

Nehmen Sie dazu die Beispiele aus der Vorlesung sowie ‚weiteren Vorlagen' und Beispiel-Codes ***genau*** zur Kenntnis. Wenden Sie diese an und

- importieren Sie die notwendigen Funktionsbibliotheken
- importieren Sie den Datensatz und passen Sie diesen ggf. an
- verschaffen Sie sich einen Überblick über den Datensatz und geben Sie statistische Kennwerte des Datensatzes aus.
- visualisieren Sie Ausgabewert(e) (Abhängige Variable) sinnvoll (gegenüber der Merkmale=Features=unabhängigen Variable).
- modifizieren Sie die Merkmale bei Bedarf.
- Sortieren Sie der/die Ausgabewert(e) der Größe nach (absteigend) und visualisieren Sie die Ergebnisse in Form eines ‚Boxplots'.
- Beobachten Sie, diskutieren und Legen Sie eine geeignetes Perzentil fest, unter und oberhalb dessen die Werte entfernt werden sollen.
- Entfernen Sie anschließend diese Werte aus dem Datensatz.
- Visualisieren Sie im Anschluss den bereinigten Datensatz und vergleichen Sie das Ergebnis mit der ersten Darstellung.
- Erstellen Sie Streudiagramme für die Merkmale des Datensatzes und nutzen Sie dafür die Funktionalität "pairplot" aus der Funktionsbibliothek "seaborn". Betrachten Sie dabei die Streudiagramme des Ausgabewerts und den jeweiligen Merkmalen und nehmen sie eine visuelle Einschätzung hinsichtlich linearer Abhängigkeiten vor. Beschreiben Sie die Ergebnisse.
- Wandeln Sie die Zahlenformate der Merkmale (inkl. Ausgabewert) in das Format "float" um.
- Legen Sie verschiedene ML-Modelle (mind. Lineare Regression und RandomForest Regression) an.
- Teilen Sie für die jeweiligen Modelle den Datensatz in Test- und Trainingsdaten auf.
- Normalisieren Sie die Merkmale (exkl. Ausgabewert) mithilfe der StandardScaler-Funktion aus der Funktionsbibliothek "sklearn.preprocessing".
- Trainieren Sie die Modelle und führen Sie Prognosen für den Testsatz durch.
- Verwenden Sie Evaluationsmetriken, um die Prognosegüte der einzelnen Modelle miteinander zu vergleichen: MAE, MAPE, R^2
  Zur Auswahl eines geeigneten Prognosemodells werden verschiedene Metriken, wie mittlerer absoluter Fehler (MAE), mittlerer absoluter prozentualer Fehler (MAPE) oder Bestimmtheitsmaß R^2 verwendet. Die Genauigkeit der Modelle ergibt sich aus 100% - MAPE.
- Bestimmen Sie anschließend das Prognosemodell mit der höchsten mittleren Prognosegüte bezüglich des R^2-Wertes.
- Erstellen Sie ein Streudiagramm für jedes Modell, in welchem Sie prognostizierte Ausgabewerte und tatsächliche Ausgabewerte gegenüberstellen. Welche Erkenntnisse lassen sich aus den Visualisierungen ableiten?
- Stellen Sie prognostizierte und tatsächliche Ausgabewerte visuell gegenüber.

Beschriften Sie jeden Schritt mit einer Überschrift und nutzen Sie die Kommentarfunktion sowie Textbausteine im Code, um die Code-Zeilen ausführlich zu beschreiben bzw. zu kommentieren.

Achten Sie bei Darstellungen und Diagrammen auf geeignete Achsenbezeichnungen und beschreiben Sie Ihre Erkenntnisse!

(Bei Datensetzen mit Zeitstempel (time series) - Extrahieren Sie die Merkmale Monat, Woche, Wochentag und Tagesstunde aus dem Zeitstempel und hängen Sie diese als zusätzliche Spalten an den Datensatz an, Berechnen Sie die durchschnittlichen Werte des Ausgabewerts je Wochentag und je Tagesstunde und Wählen Sie eine geeignete Darstellung zur Visualisierung der Werte.)

Projektgruppen

- Gruppe 1: Hr. Sabbagh, Hr. Schwab
- Gruppe 2: Hr. Dolaplis, Hr. Kendzia
- Gruppe 3: Fr. Ziebarth, Hr. Toth
- Gruppe 4: Fr. Iskandar, Fr. Gokmauli
- Gruppe 5: Hr. Kalina, Hr. Kruschinski
- Gruppe 6: Hr. Arslan, Hr. Tiedemann
- Gruppe 7: Fr. Ben Ismail, Fr. Taktak

Datensätze und Zuordnung

A: Gruppe 1 und Gruppe 2

https://www.kaggle.com/datasets/claytonmiller/cubems-smart-building-energy-and-iaq-data?select=2019Floor3.csv

Energieverbrauch als Funktion der Tageszeit; Vorhersage der Beleuchtungsverbrauch von der Lichtintensität,…

B: Gruppe 3 und Gruppe 4

Titel: “Energy Consumption depending on: ‘green’ certification, building type and building size”

Suchen Sie den Datensatz “Voluntary Disclosed Building Energy Performance Data“ unter https://data.gov.sg) – Achten Sie darauf, dass nicht alle Merkmale (z.B. Building Adresse und Building name) bei der Prognose wichtig sind.

C: Gruppe 5 und Gruppe 6

Titel: “Electricity consumption prediction in a house”

http://archive.ics.uci.edu/ml/datasets/Appliances+energy+prediction

Bitte Weiteren Weblink unter ,**Data Set Information**, beachten!

D: Gruppe 7

Titel: "Building heating and cooling Load prediction"

http://archive.ics.uci.edu/ml/datasets/Energy+efficiency

Projektrahmenbedingungen

- Zur Unterstützung werden generische .ipynb-Vorlagen hochgeladen
- Abgabefrist **17.07.2022**
- Präsentationen **07.07.2022 – 15-17:30 Uhr (10 Minuten pro Gruppe)**
- **Projektoutput**/Abgaben sind: **I)** Die csv-Datei des Ursprungs- ***und*** ‚manipulierten' Datensatzes, funktionierende ipynb-Datei mit Outputs und ausführlichen Kommentaren, **II)** Bericht über die Bearbeitung, Annahmen, Erkenntnisse und Lessons-Learned (ca. 10 Seiten, exkl. Titelblatt und Verzeichnisse), **III)** Präsentation
- Gruppen dürfen sich gegenseitig mit ‚Tipps' unterstützen aber die Lösung bzw. Bearbeitung soll eigenständig erfolgen (Plagiat vermeiden!).

Ergänzende Hinweise

Datenimport:

- Ggf. Trennzeichen der CSV Datei angeben (oftmals ';')
- Angabe des Komma-Zeichens [decimal=','], falls Werte in CSV-Datei Kommata enthalten (z.B. 10,8 und nicht 10.8)

Grundsätzlich:

- Pandas und Numpy -Funktionsbibliotheken stellen die Infrastruktur zum effizienten Rechnen mit Zahlen und Matrizen dar. Die grundlegenden Befehle und Ausdrücke sollten verstanden werden. https://pandas.pydata.org/pandas-docs/stable/user_guide/10min.html
- keine Sonderzeichen od. Ä,Ö,Ü usw. in den Datei- oder Variablennamen verwenden!
- Alle Funktionalitäten und Befehle können unter https://scikit-learn.org nachgelesen werden, es werden viele vereinfachte Beispiele gezeigt die das Verständnis stärken. *Beschäftigen Sie sich aktiv mit diesen Informationsquellen!*
- Bei Problemen und Schwierigkeiten immer den betreffenden Teil des Codes googlen und bspw. auf https://stackoverflow.com/ in den jeweiligen Foren recherchieren, somit kann nahezu jedes Problem gelöst werden (meistens müssen die gefundene Lösungen jedoch noch auf das eigene Problem angepasst werden, was wiederum eine gute Übung ist, um die Funktionalität zu verstehen)
- Falls dennoch Unklarheiten auftreten hilft es oft ein überschaubaren „Mini"- Datensatz zu erstellen (bspw. Mit Zufallszahlen oder selbst festgelegten Werten) um dann Schritt für Schritt aufzuschlüsseln und nachzuvollziehen was die jeweiligen Code-Zeile bewirken
- Viele Befehle bewirken Berechnungen und Aktion im „Hintergrund", die Ergebnisse sind nicht immer transparent insbesondere wenn es zu Verkettungen von Funktionen und Befehlen kommt. Oftmals hilft es Zwischenzeilen einzufügen (z.B. ‚print(df)'), um so nachzuvollziehen, welche weiteren Merkmale an den Datensatz angehangen worden sind. Oder beispielsweise sich Variable wie ‚test_index' ausgeben zu lassen um den Wert zu überprüfen.
- Wenn Datenbeschreibungen (Descriptions, Metdata, Readme, …) in Links zu Datensätzen vorhanden sind, sollen diese unbedingt zur Kenntnis genommen werden!