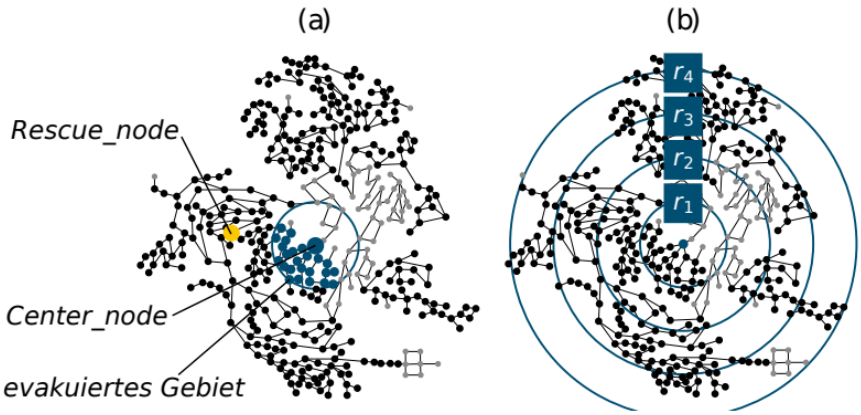
Städtische Wasserversorgungsnetze sind für eine bestimmte räumliche Verteilung von Bedarfen ausgelegt. Wenn es zu einem kritischen Ereignis kommt, infolge dessen gesamte Gebiete einer Stadt evakuiert werden müssen, verändert sich die räumliche Verteilung von Bedarfen stark und es kann passieren, dass manche Teile der Stadt nicht mehr versorgt werden können.
Sie möchten dieses Phänomen erforschen. Während Ihrer Literaturrecherche finden Sie eine wissenschaftliche Studie[1] zur urbanen Wasserversorgung. Sie entscheiden sich, den in dieser Studie verwendeten Datensatz zu eigenen Forschungszwecken wiederzuverwenden. Der Datensatz ist eine HDF5-Datei, in der *pandas DataFrames* mit Metadaten ausgestattet gespeichert sind. Den Datensatz können Sie sich unter folgendem Link herunterladen:



Die HDF5-Datei beinhaltet Simulationsdaten eines städtischen Wasserversorgungssystems. Das Wasserversorgungssystem ist als mathematischer Graph mit Knoten und Kanten modelliert, Abb. 1. Die Knoten sind Quellen (*reservoirs*), Speicher (*tanks*) oder Verbraucher (*junctions*). Die Kanten stellen Rohrleitungen dar. In der Studie wird das Systemverhalten in einem Katastrophenfall untersucht. Hierbei wird ein Stadtgebiet mit dem Radius r um den Ursprungsknoten (*Center_node*) evakuiert und die ansässige Bevölkerung in Notunterkünften (*Rescue_node*, gelber Punkt, Abb. 1 (a)) weiterhin mit Wasser versorgt. Dabei wird der Radius des evakuierten Gebietes variiert mit $r \in \{500\ \text{m}, 1000\ \text{m}, 1500\ \text{m}, 2000\ \text{m}\}$, Abb. 1 (b). Für jeden Radius werden jeweils unterschiedliche Verbraucherknoten als *Center_node* des evakuierten Gebietes gewählt. Für jeden *Center_node* werden anschließend unterschiedliche Verbraucherknoten außerhalb des evakuierten Gebietes gewählt, welche die evakuierte Bevölkerung aufnehmen (*Rescue_node*, Abb.1 (a)).
Für jede Kombination aus *Radius*, *Center_node* und *Rescue_node* wird das Verhalten des Systems über einen Zeitraum von $\Delta t = 48$ h simuliert. Der Zeitschritt der Simulation beträgt 1 h $= 3600$ s. Die Evakuierung findet nach dem Zeitpunkt $t = 3$ h $= 10800$ s statt. Danach beträgt der Bedarf des *Center_node* sowie aller weiterer *junctions* im evakuierten Gebiet 0 m^3/s. Für die Größen Druck (*pressure*), Verbrauch (*demand*) und Bedarf an den einzelnen Verbraucherknoten (*req_demand*) wird bei jedem Zeitschritt ein Wert erfasst. Die Werte für jeden Zeitschritt und jeden Knoten werden in ein *pandas DataFrame* gespeichert. Die *pandas DataFrames* sind in einer Gruppe mit einem eindeutigen Pfad aus der Kombination von *Radius*, *Center_node* und *Rescue_node* in der HDF5-Datei gespeichert.

Sie möchten die Ergebnisdaten Ihres persönlichen Schlüssels auslesen, analysieren und visualisieren, um sie für Ihre eigene Studie vorzubereiten. Erstellen Sie Ihr Hauptskript *main.py* und importieren Sie alle benötigten Pakete. Achten Sie auf eine übersichtliche Struktur des Skripts. Kommentieren Sie den Code und verwenden Sie *docstrings*, wo dies angemessen ist. Für Kommentare sowie Ausgaben im Programm können Sie als Sprache sowohl Englisch als auch Deutsch wählen. Achten Sie jedoch auf Konsistenz und Klarheit. Versionieren Sie Ihren Code.

Aufgabe 1: Daten prüfen

Für die Wiederverwendung von Daten ist es zuerst wichtig zu prüfen, ob diese verfügbar sind, wie sie entstanden sind und ob sie plausibel sind.

a) Beginnen Sie Ihr Programm mit dem Deklarieren der Variablen für die Kombination Ihres persönlichen Schlüssels. Erweitern Sie das Programm, um die Datensätze Ihrer Gruppe als *pandas DataFrames* auszulesen und in Variablen zu speichern.

b) Definieren Sie in Ihrem Programm eine Funktion, welche die Metadaten (Attribute) eines Objekts in der HDF5-Datei ausliest. Übergeben Sie der Funktion den Dateinamen, den Pfad des Objektes (Datensatz oder Gruppe) und den Namen des Attributes. Die Funktion soll den Wert des Attributes zurückgeben. Fangen Sie den Fehler ab, falls der übergebene Name des Attributes nicht vorhanden ist. Rufen Sie die Funktion in Ihrem Programmablauf auf und lesen Sie die Werte der Attribute *timestamp* und *simulator* aus. Lesen Sie zusätzlich die Software-Versionen *py_version*, *wntr_version* und *simulator_version* aus. Schließlich sollte noch die Größe *quantity* und deren Einheit *units* ausgelesen werden.

c) Definieren Sie in Ihrem Programm eine Funktion, welche die Plausibilität der Daten prüft. Sie wissen, dass ab dem Zeitschritt $t = 4$ h $= 14400$ s der Wert *req_demand* des *Center_node* in jedem weiteren Zeitschritt 0 m^3/s betragen muss. Übergeben Sie der Funktion den *pandas DataFrame req_demand*. Lesen Sie diese Größe für Ihren *Center_node* aus und summieren Sie die Einträge ab dem fünften Zeitschritt bis zum Ende der Simulation. Abhängig davon, ob das Ergebnis mit dem erwarteten Wert 0 m^3/s übereinstimmt, sollte die Funktion eine Aussage darüber ausgeben, ob die Daten plausibel sind oder nicht.

Aufgabe 2: Daten analysieren

Um aus Daten Erkenntnisse zu ziehen, müssen diese verarbeitet und aggregiert werden. Aggregierte Daten vereinfachen die Analyse und erlauben konkrete Aussagen. Verarbeiten Sie Ihre Daten, um die mittlere Bedarfserfüllung im Netzwerk und die Standardabweichung herauszufinden. Führen Sie außerdem eine Verarbeitung durch, um den zeitlichen Mittelwert des Drucks bei verschiedenen Knoten aus Ihren Daten zu berechnen.

a) Definieren Sie eine Funktion, die Ihnen ein *pandas DataFrame* mit der prozentualen Bedarfserfüllung für jeden Verbraucherknoten zu jedem Zeitschritt zurückgibt. Übergeben Sie der Funktion die *pandas DataFrames* für die Größen *req_demand* und *demand*. Beachten Sie, dass im *pandas DataFrame demand* abgesehen von Einträgen für *junctions* auch Einträge für *tanks* und *reservoirs* vorhanden sind. Reduzieren Sie daher den *pandas DataFrame demand* auf die Einträge für *junctions*, so dass er die gleiche Form wie der *pandas DataFrame req_demand* aufweist. Teilen Sie den *pandas DataFrame demand* durch den *pandas DataFrame req_demand* und entfernen Sie alle Spalten, bei denen die Division durch 0 den Eintrag *NaN* ausgelöst hat. Die Funktion sollte den resultierenden *pandas DataFrame* zurückgeben. Rufen Sie die Funktion in Ihrem Programmablauf auf und speichern Sie den *pandas DataFrame* in einer Variable.

b) Fügen Sie dem *pandas DataFrame* der prozentualen Bedarfserfüllung eine neue Spalte hinzu, die den Mittelwert der Zeileneinträge enthält. Fügen Sie eine weitere Spalte hinzu, welche die Standardabweichung der Zeileneinträge beinhaltet. Denken Sie daran, die Spalte des Mittelwerts dabei auszulassen. Nutzen Sie die dafür in *pandas* implementierten Funktionen.

c) Definieren Sie eine Funktion, die Ihnen das zeitliche Mittel des Drucks sowie die Standardabweichung bei fünf Verbraucherknoten aus Ihren Daten als *pandas DataFrame* zurückgibt. Einer der Verbraucherknoten muss der *Rescue_node* sein. Übergeben Sie Ihrer Funktion den *pandas DataFrame pressure* sowie ein weiteres Objekt, das Ihnen erlaubt in der Funktion vier Verbraucherknoten und Ihren *Rescue_node* auszuwählen. Erstellen Sie in der Funktion einen neuen *pandas DataFrame*, dessen Spalten die fünf gewählten Verbraucherknoten sind. Als Zeileneinträge sollte er den gemittelten Wert des Drucks und dessen Standardabweichung der jeweiligen Knoten beinhalten. Die Funktion sollte den *pandas DataFrame* zurückgeben. Rufen Sie die Funktion in Ihrem Programmablauf auf.

Aufgabe 3: Daten visualisieren

Visualisierung ist notwendig, um Zusammenhänge, die sich aus analysierten Daten erkennen lassen, schnell erfassbar zu machen. Plotten Sie Ihre analysierten Daten in einer Abbildung mit zwei Plots.

a) Erstellen Sie einen Plot der mittleren Bedarfserfüllung im Netzwerk über der Zeit. Fügen Sie den Einträgen Fehlerbalken hinzu. Die Fehlerbalken sollten den Wert der Standardabweichung darstellen. Übersteigt der obere Fehlerbalken 100 %, wählen Sie stattdessen die Differenz zwischen maximalem Wert und Mittelwert als Fehlerbalken. Unterschreitet der untere Fehlerbalken 0 %, wählen Sie stattdessen die Differenz zwischen minimalem Wert und Mittelwert als Fehlerbalken.

b) Erstellen Sie einen weiteren Plot, in dem Sie den zeitlichen Mittelwert des Drucks der fünf von Ihnen gewählten Verbraucherknoten darstellen. Fügen Sie den Einträgen Fehlerbalken hinzu, welche die Standardabweichung darstellen. Passen Sie die Fehlerbalken wie in Aufgabenteil a) an, falls der untere Fehlerbalken einen Wert von 0 m unterschreitet.

c) Speichern Sie beide Plots als Subplot einer Abbildung in einer skalierbaren Vektorgraphik (*plot.svg*). Erstellen Sie Abbildungen hoher formaler und ästhetischer Qualität, die selbsterklärend sind.