

Hinweise:

- **Abgabe bis 27. Juni 2021, 23:59 Uhr per E-Mail an jens.flemming@fh-zwickau.de.**
 - *Alle Dateien sind in einer ZIP-Datei als E-Mail-Anhang oder Cloud-Link abzugeben. Die ZIP-Datei muss den Name `vorname_nachname.zip` haben.*
 - *Der Quellcode muss in der `ds_lecture`-Umgebung lauffähig sein (vgl. Lehrveranstaltung).*
 - *Lesen Sie das Aufgabenblatt vollständig bevor Sie mit der Umsetzung beginnen!*
-

Zusammenfassung: Nutzen Sie faltende neuronale Netze zur Wettervorhersage. Das erstellte Modell soll für einen gewählten Tag wesentliche Wetterparameter (Temperatur, Niederschlag usw.) für die Stadt Nürnberg vorhersagen. Die Vorhersage soll ausschließlich auf von der Wetterstation Nürnberg aufgezeichneten Wetterdaten basieren.

Daten: Zum Trainieren und Evaluieren des Modells stehen stündliche Messwerte verschiedener Wetterparameter der Wetterstation Nürnberg von September 1995 bis Dezember 2018 zur Verfügung. Alle Daten sind in der Datei `station3668.csv` enthalten. Verwenden Sie ausschließlich die mit der Aufgabenstellung bereitgestellte CSV-Datei! Folgende Werte stehen zur Verfügung:

- Spalten `ta`, `mo`, `jahr`, `HH`: Zeitpunkt der Messung (Tag, Monat, Jahr, Stunde); Stunde 14 bedeutet zum Beispiel „Messung zwischen 14:00 Uhr und 15:00 Uhr“
- Spalte `TM`: Temperatur (in °C)
- Spalte `RR`: Niederschlagsmenge (in mm)
- Spalte `RF`: relative Luftfeuchtigkeit (in %)
- Spalte `PP`: Luftdruck (in hPa)
- Spalte `SD`: Sonnenscheindauer (in Minuten)
- Spalte `NN`: Bedeckungsgrad (in Achtel der Himmelsfläche)
- Spalte `FF`: Windgeschwindigkeit (in m/s)
- Spalte `WR`: Windrichtung (in °, Ost = 0°, Nord = 90° usw.)

Vorhersage: Die Wettervorhersage für einen Tag soll folgende Werte umfassen:

- niedrigste Temperatur (in °C),
- höchste Temperatur (in °C),
- Niederschlagsmenge (in mm),
- mittlere Windrichtung (in °, Ost = 0°, Nord = 90° usw.),
- mittlere Windgeschwindigkeit (in m/s).

Alle vor 00:00 Uhr aufgezeichneten Messwerte können für die Vorhersage verwendet werden. Beispiel: Zur Vorhersage des Wetters am 5. Oktober dürfen alle bis zum 4. Oktober, 23:59 Uhr aufgezeichneten Messwerte verwendet werden (einschließlich der Werte zur Stunde 23).

Vorgehen: Halten Sie sich beim Lösen der Aufgabe an die folgenden Arbeitsschritte. Die Reihenfolge der Bearbeitung kann variiert werden. Der Quellcode kann auf mehrere Dateien verteilt werden. Bearbeiten Sie bitte jede Teilaufgabe und verzichten Sie auf zusätzliche Funktionalität, die nicht gefordert wurde. Die in den Arbeitsschritten enthaltenen Fragen müssen **nicht schriftlich** beantwortet werden, sondern sollen zur Vorbereitung auf die Projektverteidigung dienen. Wählen Sie Jupyter-Notebooks und/oder Python-Skripte für die Umsetzung.

1. Daten beschaffen und vorverarbeiten:

- (a) Laden Sie die Wetterdaten aus `station3668.csv` in einen Pandas-DataFrame. (1 Punkt)
- (b) Berechnen Sie für jeden Zeitpunkt einen „Windvektor“. Die Richtung des Vektors entspricht der Windrichtung; die Länge entspricht der Windgeschwindigkeit. Fügen Sie zwei Spalten zum Speichern dieser Vektoren zum DataFrame hinzu (Komponente in West-Ost-Richtung und Komponente in Süd-Nord-Richtung). Arbeiten Sie im Folgenden stets mit dem „Windvektor“ statt mit Richtung und Geschwindigkeit. (1 Punkt)
- (c) Entfernen Sie Spalten, die Sie nicht benötigen. Warum verzichten Sie gegebenenfalls auf einige Spalten? (1 Punkt)
- (d) Füllen Sie kleine Lücken in den Daten mit `pd.DataFrame.interpolate`. Welche Lückengröße kann noch sinnvoll durch Interpolation gefüllt werden? (1 Punkt)
- (e) Standardisieren oder Normalisieren Sie alle Messwerte. Überlegen Sie für welche Messgrößen Standardisieren sinnvoller sein kann und für welche Messgrößen Normalisieren sinnvoller sein könnte. Warum sollten die beiden Komponenten des „Windvektors“ nicht unabhängig voneinander skaliert werden? (1 Punkt)

Bemerkung zum Verständnis: Sinn des „Windvektors“ ist eine einfachere Handhabung der Winddaten. Wird direkt mit der Windrichtung gearbeitet, so ist die mittlere Windrichtung nicht immer aussagekräftig (Beispiel: das Mittel aus 1° und 359° (beide Westwind) wäre rechnerisch 180° , also Ostwind). Auch die mittlere Windgeschwindigkeit ist nicht immer aussagekräftig. Weht zum Beispiel zunächst Ostwind mit 10 m/s und anschließend Westwind mit 10 m/s , so ist die mittlere Windgeschwindigkeit 10 m/s , obwohl sich die Luftmasse im Mittel gar nicht bewegt hat. Wird der Wind als Vektor betrachtet, so können beide Komponenten separat gemittelt werden. Ein durch Mittelwertbildung entstandener „Windvektor“ beschreibt dann die mittlere Bewegung der Luftmasse. Aus der Vektorschreibweise

können jederzeit wieder Windrichtung (in °) und Windgeschwindigkeit berechnet werden.

2. Daten auswählen: Nutzen Sie für die Vorhersage des Wetters für einen Tag die stündlichen Wetterdaten der vergangenen 28 Tage.

- (a) Erstellen Sie einen DataFrame, der die vorherzusagenden Wetterdaten für jeden Tag von Oktober 1995 bis Dezember 2018 enthält (Modell-Outputs). Nehmen Sie nur Tage mit vollständigen Messwerten für die relevanten Messgrößen in den DataFrame auf. (1 Punkt)
- (b) Erstellen Sie für jeden Tag von Oktober 1995 bis Dezember 2018 einen DataFrame, der die stündlichen Wetterdaten der vergangenen 28 Tage enthält (Modell-Inputs), sofern die Messwerte für die 28 Tage vollständig vorliegen. (1 Punkt)
- (c) Erstellen Sie ein NumPy-Array mit 3 Dimensionen (Tag, Messzeitpunkt, Messgröße), welches für jeden Tag mit vollständigen Daten (28-Tage-Zeitraum und Vorhersage) die Messwerte der vergangenen 28 Tage enthält. (1 Punkt)
- (d) Erstellen Sie ein NumPy-Array mit 2 Dimensionen (Tag, Vorhersagegröße), welches für jeden Tag mit vollständigen Daten (28-Tage-Zeitraum und Vorhersage) die Vorhersagen enthält. (1 Punkt)

Achten Sie beim Erstellen der beiden NumPy-Arrays darauf, dass bei Bedarf eine Zuordnung der Zeilen zu Datumsangaben möglich ist. Legen Sie dazu beispielsweise eine Liste mit den zugehörigen Datumsangaben an.

3. Modell erstellen: Nutzen Sie Keras und den Keras-Tuner zum Trainieren eines faltenden neuronalen Netzes (CNN).

- (a) Nutzen Sie in Ihrem CNN die Keras-Layer `Conv1D`, `MaxPooling1D` und `Dense`. (1 Punkt)
- (b) Ermitteln Sie eine geeignete Layer-Anzahl und die Anzahl der Neuronen pro Layer durch automatische Hyperparameteroptimierung (Keras-Tuner). (3 Punkt)
- (c) Verwenden Sie für Training und Hyperparameteroptimierung ausschließlich Wetterdaten von September 1995 bis Dezember 2014. Warum ist Datenaugmentierung hier nicht sinnvoll? (1 Punkt)
- (d) Achten Sie darauf, dass die Vorhersagen Ihres Modells im richtigen Wertebereich liegen (die Niederschlagsmenge sollte zum Beispiel nicht negativ sein). Wählen Sie dazu geeignete Aktivierungsfunktionen für den Output-Layer oder korrigieren Sie die Vorhersagen nachträglich. (1 Punkt)

Hinweis: Keras unterstützt bei `Sequential`-Modellen nur eine Aktivierungsfunktion für alle Output-Neuronen. Falls Sie verschiedene Aktivierungsfunktionen nutzen möchten, müssen Sie die sogenannte Functional-API von Keras nutzen (siehe

Keras-Dokumentation). Dies ist für das Projekt optional. Alternativ können Sie bei Bedarf die Vorhersagen des Modells nachträglich anpassen.

4. Modell evaluieren:

- (a) Ermitteln Sie die Wurzel des mittleren quadratischen Fehlers (RMSE) der Vorhersagen Ihres Modells für die Tage von Januar 2015 bis Dezember 2018. (1 Punkt)
- (b) Machen Sie die Standardisierung/Normalisierung der Vorhersagen rückgängig, sodass Sie im Folgenden mit den üblichen Einheiten ($^{\circ}\text{C}$ usw.) arbeiten können. (1 Punkt)
- (c) Ermitteln Sie RMSE und maximalen Fehler im Evaluierungszeitraum für jede der fünf Vorhersagegrößen (min./max. Temperatur, Niederschlag, Komponenten des „Windvektors“). (1 Punkt)
- (d) Stellen Sie separat für jedes Jahr des Evaluierungszeitraums vorhergesagte und tatsächliche Werte für die fünf Messgrößen grafisch so dar, dass eine einfache Beurteilung der Vorhersagefehler möglich ist. Denken Sie an Übersichtlichkeit, Diagrammbeschriftungen, Achsenbeschriftungen, Legenden. (3 Punkte)
- (e) Speichern Sie die Darstellungen für die 4 Jahre in einer PDF-Datei im A4-Format. (1 Punkt)

5. Modell anwenden:

- (a) Speichern Sie Ihr trainiertes Modell. (1 Punkt)
- (b) Sagen Sie mit Ihrem Modell das Wetter für den 1. Januar 2019 vorher. Sollten trotz Interpolation in 1(d) noch Lücken in den dazu benötigten Daten sein, füllen Sie die Lücken mit `pd.DataFrame.interpolate`. Geben Sie die Vorhersage in den üblichen Einheiten ($^{\circ}\text{C}$ usw.) aus. Geben Sie Windrichtung und -geschwindigkeit an, nicht den „Windvektor“ (2 Punkt)

Abgabe: Geben Sie die folgenden Dateien ab:

- kompletter Quellcode,
- PDF-Datei aus 4(e),
- gespeichertes CNN-Modell aus 5(a),
- falls Sie keine Jupyter-Notebooks nutzen: pro Python-Skript eine Textdatei mit allen Ausgaben des Skripts.

Packen Sie alle Dateien in eine (!) ZIP-Datei. Ist die Dateigröße unter 5 MB, schicken Sie die Datei bitte per E-Mail an den Vorlesenden. Ist die Dateigröße über 5 MB, laden Sie die Datei bitte in eine Cloud (z.B. TUCcloud) und schicken dem Vorlesenden den Link.