

# Vorbereitung für das Seminar „Maschinelles Lernen: Potentiale und Risiken“

Marvin Kastner <marvin.kastner@extern.nordakademie.de>

1. Januar 2025

In dieser Anleitung werden die Schritte aufgezählt, die von den Seminarteilnehmenden selbständig *vor* Beginn des Seminars auf dem eigenen Endgerät (Laptop, MacBook, ...) durchgeführt werden sollen. Bitte planen Sie hierfür zirka eine Stunde ein, bei langsamen Internet oder weniger leistungsfähigen Endgeräten entsprechend mehr. Falls Sie während des Seminars kein eigenes leistungsfähiges Endgerät (z. B. nur einen alten Laptop oder ein Tablet) zur Verfügung haben sollten, sprechen Sie bitte frühzeitig die Seminarleitung an.

Übrigens – falls Sie Anaconda bereits installiert haben sollten, stellen Sie bitte sicher, dass Ihre Version aktuell ist. Nachdem Sie conda ggf. geupdatet haben, können Sie direkt zum Abschnitt *Bezug der Seminar-Materialien* springen und die conda-Umgebung mit den benötigten Bibliotheken installieren.

## Installation von JupyterLab

Gehen Sie auf <https://www.anaconda.com/download> und laden Sie die Anaconda-Version für Ihr Betriebssystem und Prozessorarchitektur herunter. Die auf der Webseite vorgeschlagene Registrierung können Sie ohne Konsequenzen überspringen. Anaconda ist eine Python-Distribution, die vorkompilierte Bibliotheken ausliefert und damit den Installationsaufwand minimiert. Der Download-Bereich der Webseite sollte wie in Abbildung 1 aussehen.

Falls Sie auf Ihrem Endgerät mithilfe von zwei Benutzeraccounts streng zwischen administrativen Tätigkeiten auf der einen Seite und dem alltäglichen Arbeiten mit stark eingeschränkten Rechten auf der anderen Seite unterscheiden, werden Sie vermutlich versuchen, Anaconda mit dem Administrator- bzw. root-Account zu installieren und danach mit dem anderen Account Anaconda zu nutzen. Falls das auf Sie zutrifft, lesen Sie bitte die Notiz in der rechten Spalte. Wenn Sie sich, wie die meisten, nur mit einem Benutzer-Account arbeiten, sind Sie hiervon nicht betroffen.

Rufen Sie nach dem Download den Installer aus dem Download-Ordner auf und folgen Sie den Installationsschritten. Konsultieren Sie im Fehlerfall offizielle Quellen des Anbieters (wie z. B. <https://docs.anaconda.com/anaconda/install/>) oder Foren (wie z. B. <https://stackoverflow.com>).

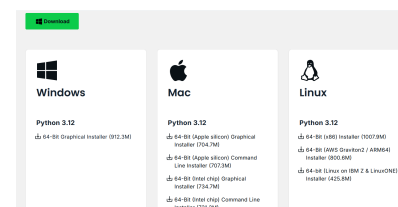


Abbildung 1: Der Download-Bereich von Anaconda (Ausschnitt).

Die Installation von Anaconda erfordert keine erhöhten Rechte. Eine Installation mit einem Nutzer mit erhöhten Rechten kann u. U. dazu führen, dass Anaconda nur für diesen einen Nutzer mit erhöhten Rechten installiert ist bzw. dass es zu Problemen mit der Rechteverwaltung kommt. In der Vergangenheit hat ein erneutes Installieren von Anaconda dieses Problem dann *nicht* beheben können. Lösungen für ein Setup mit mehreren Benutzern lassen sich unter <https://docs.anaconda.com/anaconda/install/multi-user/> finden.

WARUM WERDEN JUPYTER NOTEBOOKS EINGESETZT? Im Bereich Maschinelles Lernen und Data Science spielen Jupyter Notebooks eine große Rolle. Das Medienformat ist auf JSON-Basis und kann neben dem ausführbaren Code u. a. auch Text (d. h. Plaintext, Markdown, LaTeX und HTML), Bilder, Visualisierungen von Daten, Videos und Animationen enthalten. So können die Arbeitsschritte für Dritte nachvollziehbar in einem Dokument zentral dokumentiert und die Ergebnisse visualisiert werden.

### *Bezug der Seminar-Materialien*

Klonen Sie das git-Repository <https://github.com/1kastner/ml-potentials-and-risks>, damit Sie die Seminar-Materialien lokal haben – dies umfasst nur den Programmierteil. Am einfachsten ist es, wenn Sie die Dateien unterhalb des benutzereigenen Ordners Dokumente ablegen (unter Windows meist C:\Users\BENUTZERNAME\Documents). Denn dieser Ordner ist, wenn JupyterLab aus Anaconda heraus gestartet wird, direkt verfügbar. Falls Sie noch nie mit git gearbeitet haben, lesen Sie bitte die nächsten Absätze.

WARUM SOLLTE ICH GIT LERNEN? Für die Versionsverwaltung ist git quasi der Standard und wird immer häufiger auch außerhalb der Software-Entwicklung eingesetzt. Deswegen lohnt es sich für (fast) alle, sich Fähigkeiten mit diesem Tool anzueignen. Es ist als ein Kommandozeilentool entwickelt worden, welches über <https://git-scm.com/> heruntergeladen werden kann. Wer lieber grafische Oberflächen mag, kann sich einen der vielen GUI-Clients<sup>1</sup> aussuchen. Hier sollte neben dem Betriebssystem auch die ggf. kostenpflichtige Lizenz beachtet werden. Einige Lizenzen unterscheiden z. B. zwischen der privaten Verwendung und der Verwendung im Arbeitskontext. Bis zum Start des Seminars werden u. U. die Materialien noch überarbeitet oder erweitert. Aktualisieren Sie also bitte regelmäßig Ihre vorliegende Version über ein `git pull` bzw. durch das Klicken auf den „Pull“-Button im GUI-Client Ihrer Wahl.

IST ES FÜR DAS SEMINAR ZWINGEND NOTWENDIG, GIT ZU LERNEN? Falls Ihnen git unbekannt ist und Sie keine Zeit dafür haben, sich mit git auseinanderzusetzen, gibt es auch die Möglichkeit, den Inhalt als ZIP-Ordner herunterzuladen. Klicken Sie dafür auf den Button, wie er in Abbildung 2 zu sehen ist. Falls Lernmaterialien später noch angepasst werden, müssen Sie diese dann allerdings erneut herunterladen und in einem neuen Ordner entpacken.

<sup>1</sup> Eine Liste ist auf <https://git-scm.com/downloads/guis> zu finden.

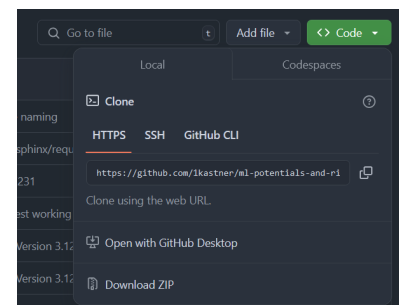


Abbildung 2: Ein GitHub-Repository bietet verschiedene Möglichkeiten zum Bezug der Inhalte an, auch den Download als ZIP-Datei.

## Installation der Bibliotheken

Mit der Installation von Anaconda ist u. a. die Applikation Anaconda Navigator mitinstalliert worden. Mit dem Anaconda Navigator können die im Seminar benötigten Bibliotheken automatisch installiert werden. In der Abbildung 3 sehen Sie unten den Button „Import“ (im Screenshot mit einer 1 markiert). Klicken Sie diesen an. Damit öffnet sich das Fenster „Import environment“. Klicken Sie hier auf den Ordner in der Zeile, die mit „Local drive“ bezeichnet wird (mit einer 2 markiert). Danach öffnet sich ein Fenster zur Datei-Auswahl (mit einer 3 markiert), in dem Sie dann zu den von GitHub bezogenen Dateien navigieren können. Wählen Sie die Datei `environment.yml` aus – sie liegt auf der Wurzelebene des Projektordners – und klicken Sie auf „Öffnen“. Der zu erstellenden Umgebung können Sie nun einen sprechenden Namen geben, z. B. `ml-potentials-and-risks`. Unter diesem Namen werden Sie die Umgebung später in Anaconda wiederfinden. Das Erstellen der Umgebung beinhaltet den Download und die Installation der benötigten Bibliotheken; es nimmt für gewöhnlich mehrere Minuten in Anspruch.

Falls Sie Windows 10/11 verwenden und kein Multi-User-Setup haben, können Sie den folgenden Schritt dadurch abkürzen, dass Sie die Batch-Datei `create-env.bat` auf der Wurzelebene des Projekts durch einen Doppelklick ausführen. Im Falle einer Nachfrage des Betriebssystems bestätigen Sie, dass Sie dem Urheber der Batch-Datei vertrauen.

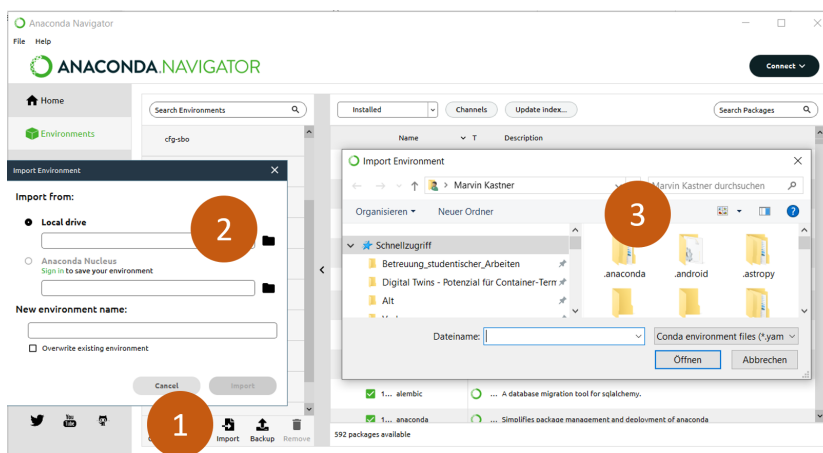


Abbildung 3: Der Anaconda Navigator erlaubt das Importieren von `environment.yml`-Dateien.

**WARUM IST DIESER SCHRITT NOTWENDIG?** Die Jupyter Notebooks, die Sie soeben von GitHub bezogen haben, benötigen eine bestimmte Umgebung, damit sie wie gewünscht geöffnet und ausgeführt werden können. Zu dieser Umgebung gehört z. B. ein Python-Interpreter mit einer bestimmten Python-Version ebenso wie eine Reihe ausgewählter Python-Bibliotheken. Auf der obersten Ebene des git-Repositorys befindet sich die Datei `environment.yml`, in der alle Anforderungen an die Entwicklungsumgebung aufgeführt werden. Die Struktur der Datei `environment.yml` ist von Anaconda vorgege-

ben und erlaubt es, die Abhängigkeiten von Bibliotheken automatisch aufzulösen. Damit man auf einem Endgerät in verschiedenen Projekten unterschiedliche Versionen einer gleichen Bibliothek (und auch JupyterLab und andere Werkzeuge) haben kann, strukturiert Anaconda die zu einem Projekt gehörenden Bibliotheken standardmäßig in Umgebungen (eng. Environments). Für das Seminar erstellen wir die Umgebung `ml-potentials-and-risks` basierend auf der gegebenen `environment.yml`.

GEHT ES DENN NUR ÜBER DIE GUI? Natürlich gibt es auch ein Kommandozeilentool, das mit Anaconda ausgeliefert worden ist. Es heißt `conda`. Unter Windows wird dieses Tool je nach Auswahl während der Installation nicht in die Pfad-Variable aufgenommen. Es steht Ihnen aber auf jeden Fall in der *Anaconda Powershell Prompt* (basierend auf der PowerShell) zur Verfügung. Auf <https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html#creating-an-environment-from-an-environment-yml-file> wird erläutert, wie eine existierende `environment.yml` zur Erstellung einer Anaconda-Umgebung genutzt werden kann.

### *Start von JupyterLab*

JupyterLab kann nun über den Anaconda Navigator gestartet werden. In Abbildung 4 ist dies abgebildet. Zunächst wird links im Menü (im Screenshot mit einer 1 markiert) „Home“ ausgewählt. Im zweiten Schritt muss die Umgebung `ml-potentials-and-risks` ausgewählt werden (mit einer 2 markiert). Danach startet ein Klick auf Launch die Anwendung JupyterLab im Browser (mit einer 3 markiert). Standardmäßig öffnet sich nun ein neuer Tab im Browser und JupyterLab zeigt zunächst den Inhalt vom Benutzer-Ordner an, wie in Abbildung 5 zu sehen. Von hier können Sie in den Unterordner Dokumente navigieren und zu den heruntergeladenen Seminarunterlagen gelangen.

Falls Sie Windows 10/11 verwenden und kein Multi-User-Setup haben, können Sie den folgenden Schritt dadurch abkürzen, dass Sie die Batch-Datei `start-jupyterlab.bat` auf der Wurzelebene des Projekts durch einen Doppelklick ausführen. Dann ist auch die Wurzelebene des Projekts in JupyterLab der Wurzelordner.

### *Erste Schritte mit JupyterLab*

Wenn Sie die Seminar-Materialien wie unter Abschnitt „Bezug der Seminar-Materialien“ angegeben abgelegt haben, können Sie in JupyterLab zum Ordner `00-installationscheck` navigieren und dort das Jupyter Notebook (die Datei mit der Endung `.ipynb`) mit einem Doppelklick öffnen. Überprüfen Sie, ob Sie alle Zellen ausführen können.

Falls es Fehlermeldungen gibt, melden Sie sich bitte umgehend per Mail bei der Seminarleitung.

Bitte schauen Sie ebenfalls im Ordner `01-einfuehrung-in-toolbox`

Wenn Sie nicht `start-jupyterlab.bat` zum Starten von JupyterLab verwendet haben, können Sie den Inhalt aus dem Ordner `.user-settings` nach `%USERPROFILE%\jupyterlab\user-settings` kopieren, um die gleichen Voreinstellungen zu verwenden. Falls Sie Apple oder Linux verwenden, ist dieser Pfad entsprechend `~/jupyterlab/user-settings`.

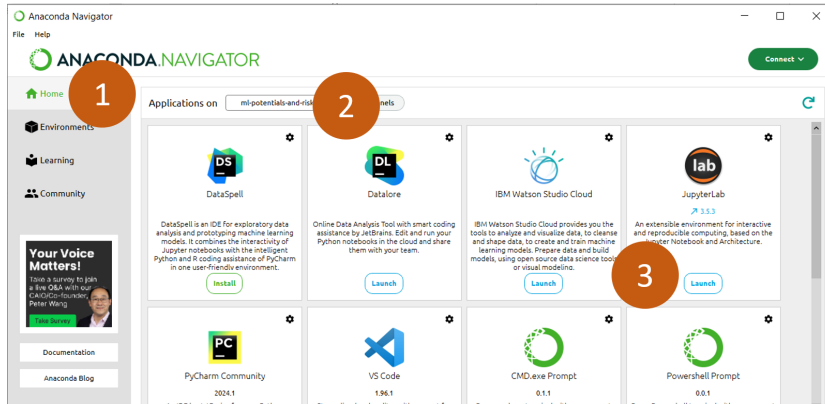


Abbildung 4: Aus dem Anaconda Navigator kann JupyterLab gleich in der richtigen Umgebung gestartet werden. Die Reihenfolge und Sortierung der Logos kann variieren.

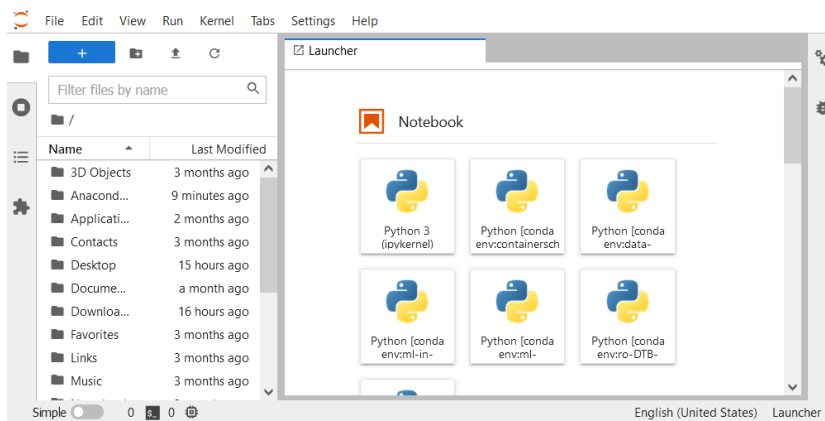


Abbildung 5: In JupyterLab können links alle Dateien und Ordner aus dem Benutzerordner betrachtet werden.

um. Falls Sie noch keine Erfahrungen mit Python haben, arbeiten Sie bitte 01-Einfuehrung-in-Python.ipynb durch. Falls Ihnen das Format des Jupyter Notebooks noch unbekannt sind, schauen Sie sich bitte 02-Was-kann-ein-Jupyter-Notebook.ipynb an. Falls Sie noch nicht mit der Anwendung JupyterLab gearbeitet haben, könnte 03-Tipps-zu-JupyterLab.ipynb für Sie interessant sein.

Dieses Werk ist lizenziert unter einer Creative Commons „Namensnennung 4.0 International“ Lizenz.

