Wissenschaftliche Tools in der Hochschullehre mit bwJupyter

Jasmin Hörter, Paul Hoger,
Stephanie Hofmann

Karlsruher Institut für Technologie

Universität Stuttgart

Baden-Württemberg
Ministerium für Wissenschaft,
Forschung und Kunst

bwJupyter - der JupyterHub für die Hochschullehre in BW

- Landesdienst bwJupyter als webbasierte interaktive Entwicklungsumgebung seit März 2025 für alle Hochschulen in Baden-Württemberg verfügbar
- MWK-BW Projekt im Rahmen von “Hochschulen in der digitalen Welt”
- Kooperation von KIT und Uni Stuttgart
- aktuell ca. 2500 Nutzenden von 33 Hochschulen

https://hub.bwjupyter.de

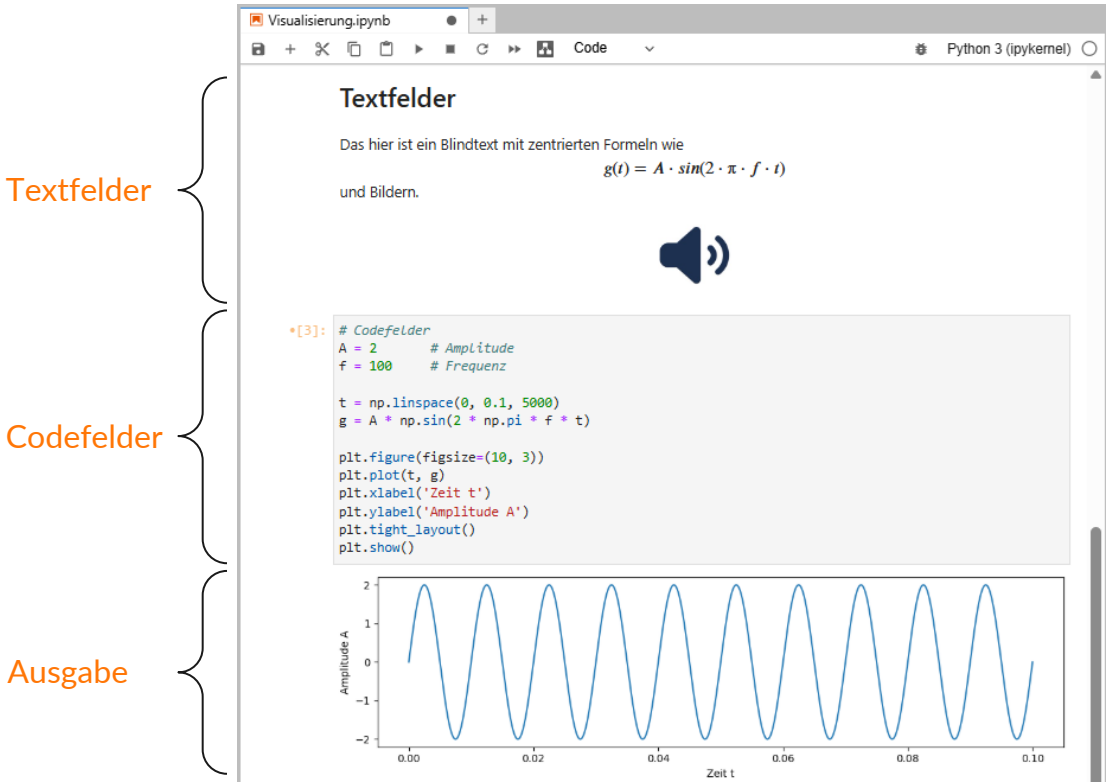
Karlsruher Institut für Technologie

Universität Stuttgart

Baden-Württemberg
Ministerium für Wissenschaft,
Forschung und Kunst

Jupyter Notebooks (JNB)

- interaktive Dokumente mit
  - Code
  - Text (Markdown & HTML)
  - Formeln (LaTeX)
  - Visualisierungen
- verschiedene Programmiersprachen unterstützt:
  - Julia
  - Python
  - R

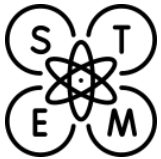


JNB in der Lehre - verschiedene Einsatzszenarien

JNB in **Dozierendenhand**

JNB in **Studierendenhand**

verschiedene **Fachrichtungen**



JNB in der Lehre - Vielseitigkeit der JNB

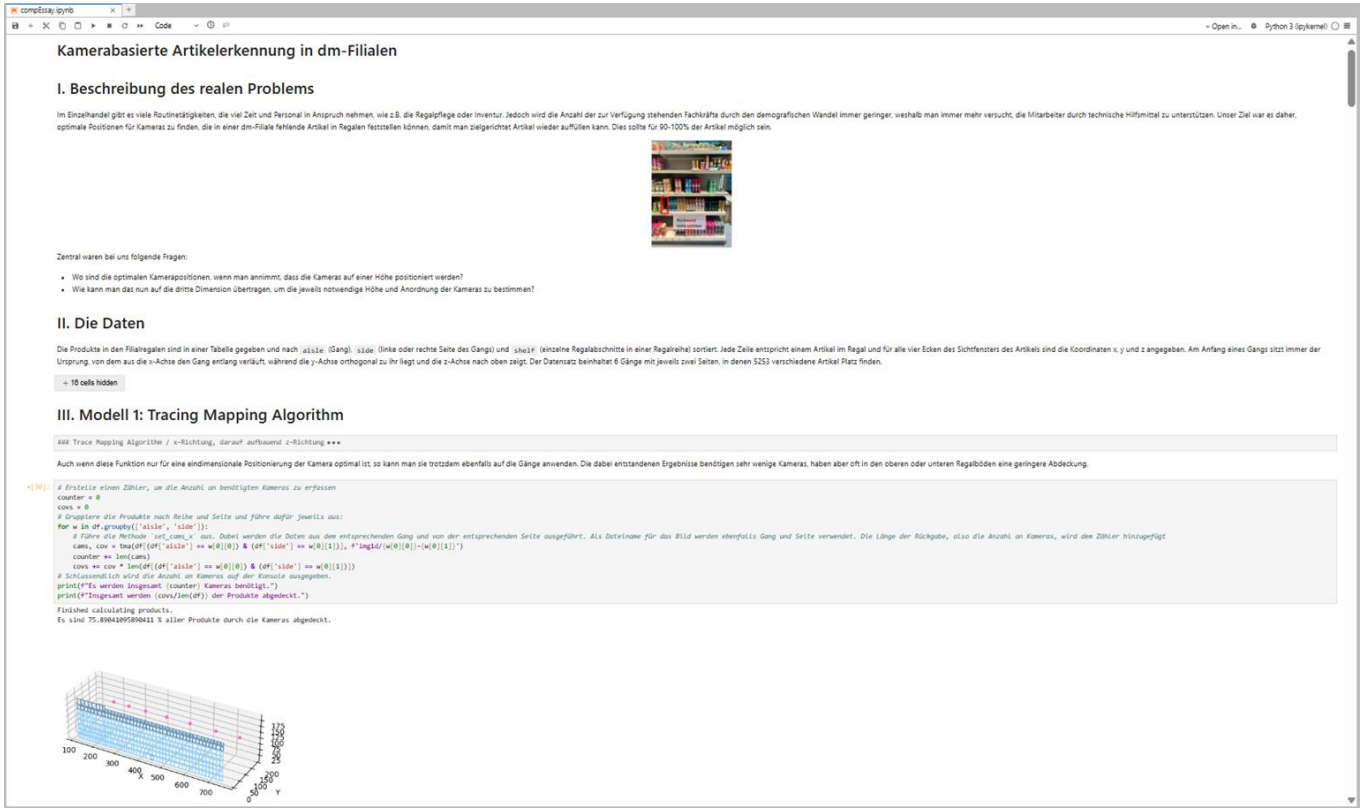
- offene Aufgabenformate
- exploratives, authentisches wissenschaftliches Arbeiten
- Programmierkenntnisse notwendig

- geschlossene Aufgabenformate
- angeleitetes Arbeiten
- für Lerngruppen mit unterschiedlichen Programmierkenntnissen einsetzbar
- Unterstützung des Lernprozesses auf verschiedenen Ebenen möglich

JNB als Werkzeug

JNB als Lernumgebung

Bsp. 1: JNB als Computational Essay



Kamerabasierte Artikelerkennung in dm-Filialen

I. Beschreibung des realen Problems

Im Einzelhandel gibt es viele Routinetätigkeiten, die viel Zeit und Personal in Anspruch nehmen, wie z.B. die Regalpflege oder Inventur. Jedoch wird die Anzahl der zur Verfügung stehenden Fachkräfte durch den demografischen Wandel immer geringer, weshalb man immer mehr versucht, die Mitarbeiter durch technische Hilfsmittel zu unterstützen. Unser Ziel war es daher, optimale Positionen für Kameras zu finden, die in einer dm-Filiale fehlende Artikel in Regalen feststellen können, damit man zielgerichtet Artikel wieder auffüllen kann. Dies sollte für 90-100% der Artikel möglich sein.

Zentral waren bei uns folgende Fragen:

- Wo sind die optimalen Kamerapositionen, wenn man annimmt, dass die Kameras auf einer Höhe positioniert werden?
- Wie kann man das nun auf die dritte Dimension übertragen, um die jeweils notwendige Höhe und Anordnung der Kameras zu bestimmen?

II. Die Daten

Die Produkte in den Filialregalen sind in einer Tabelle gegeben und nach `aisle` (Gang), `side` (linke oder rechte Seite des Gangs) und `shelf` (einzelne Regalabschnitte in einer Regalreihe) sortiert. Jede Zeile entspricht einem Artikel im Regal und für alle vier Ecken des Sichtfensters des Artikels sind die Koordinaten x, y und z angegeben. Am Anfang eines Gangs sitzt immer der Ursprung, von dem aus die x-Achse den Gang entlang verläuft, während die y-Achse orthogonal zu ihr liegt und die z-Achse nach oben zeigt. Der Datensatz beinhaltet 6 Gänge mit jeweils zwei Seiten, in denen S253 verschiedene Artikel Platz finden.

+ 16 cells hidden

III. Modell 1: Tracing Mapping Algorithm

```
### Trace Mapping Algorithm / x-Richtung, darauf aufbauend z-Richtung •••
```

Auch wenn diese Funktion nur für eine eindimensionale Positionierung der Kamera optimal ist, so kann man sie trotzdem ebenfalls auf die Gänge anwenden. Die dabei entstandenen Ergebnisse benötigen sehr wenige Kameras, haben aber oft in den oberen oder unteren Regalböden eine geringere Abdeckung.

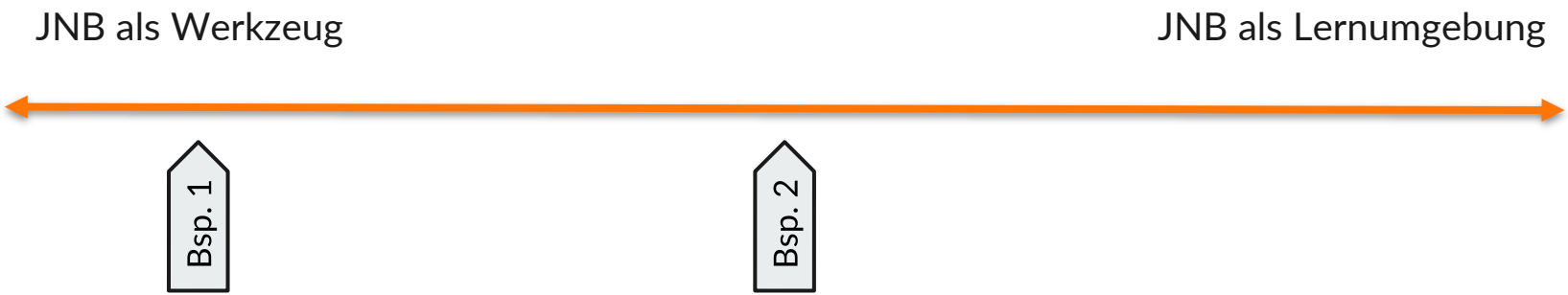
```
# Erstelle einen Zähler, um die Anzahl an benötigten Kameras zu erfassen
counter = 0
covs = 0
# Gruppiere die Produkte nach Reihe und Seite und führe dafür jeweils aus:
for w in df.groupby(['aisle', 'side']):
    # Führe die Methode `set_cams_x` aus. Dabei werden die Daten aus dem entsprechenden Gang und von der entsprechenden Seite ausgeführt. Als Dateiname für das Bild werden ebenfalls Gang und Seite verwendet. Die Länge der Rückgabe, also die Anzahl an Kameras, wird dem Zähler hinzugefügt
    cams, cov = tma(df[(df['aisle'] == w[0][0]) & (df['side'] == w[0][1])], f'img1d/{w[0][0]}-{w[0][1]}')
    counter += len(cams)
    covs += cov * len(df[(df['aisle'] == w[0][0]) & (df['side'] == w[0][1])])
# Schlussendlich wird die Anzahl an Kameras auf der Konsole ausgegeben.
print(f"Es werden insgesamt {counter} Kameras benötigt.")
print(f"Insgesamt werden {covs/len(df)} der Produkte abgedeckt.")
```

```
Finished calculating products.
Es sind 75.89041095890411 % aller Produkte durch die Kameras abgedeckt.
```

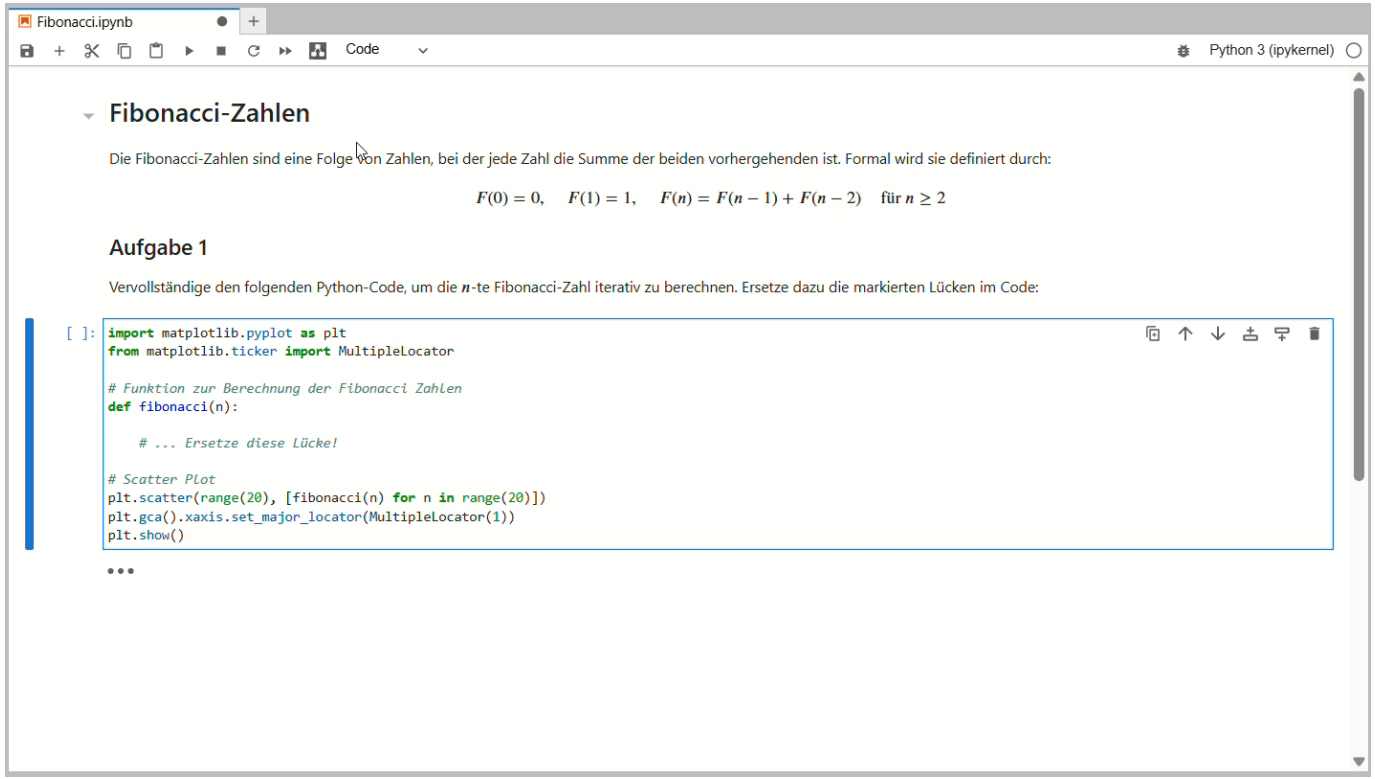
JNB in der Lehre - Vielseitigkeit der JNB

- offene Aufgabenformate
- explorativ, authentisches wissenschaftliches Arbeiten
- Programmierkenntnisse notwendig

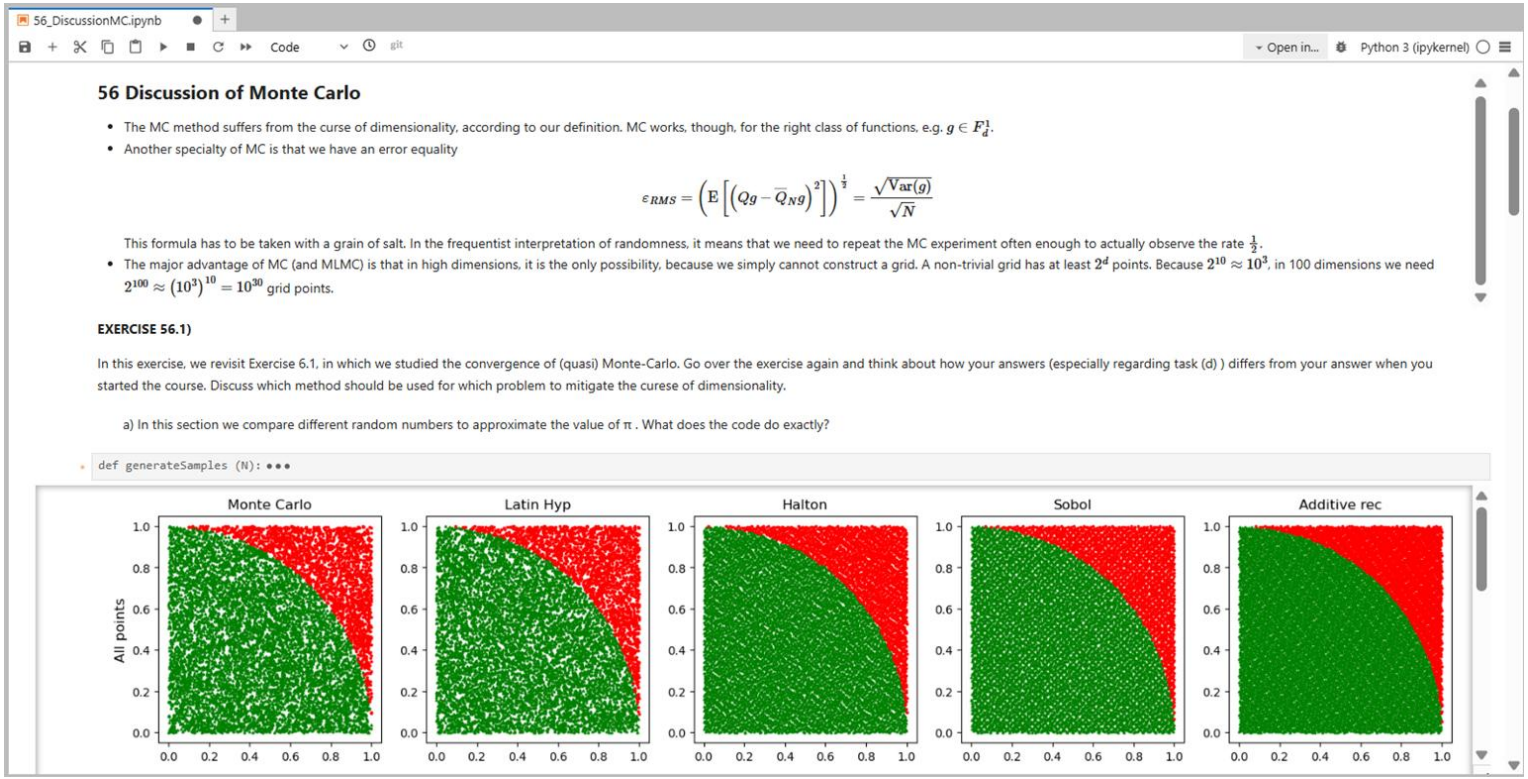
- geschlossene Aufgabenformate
- angeleitetes Arbeiten
- für Lerngruppen mit unterschiedlichen Programmierkenntnissen einsetzbar
- Unterstützung des Lernprozesses auf verschiedenen Ebenen möglich



Bsp. 2.1: JNB als Übungsblatt



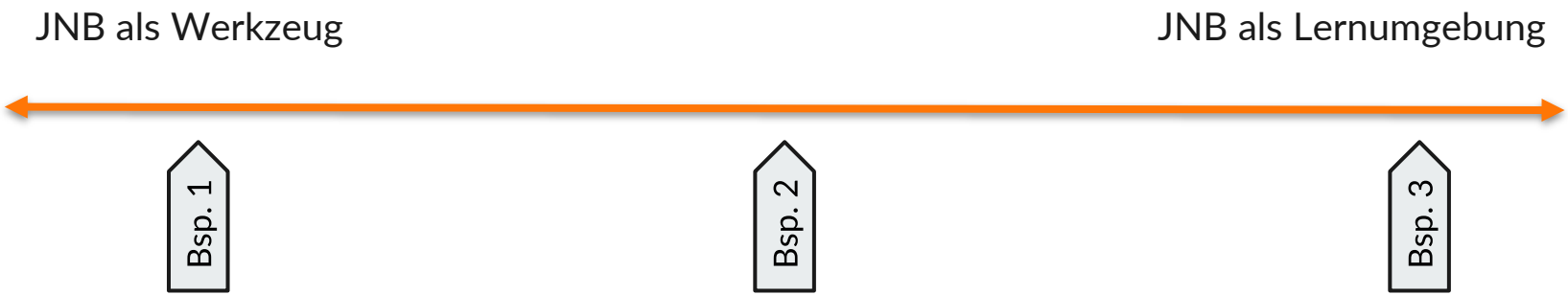
Bsp. 2.1: JNB als interaktives Skript



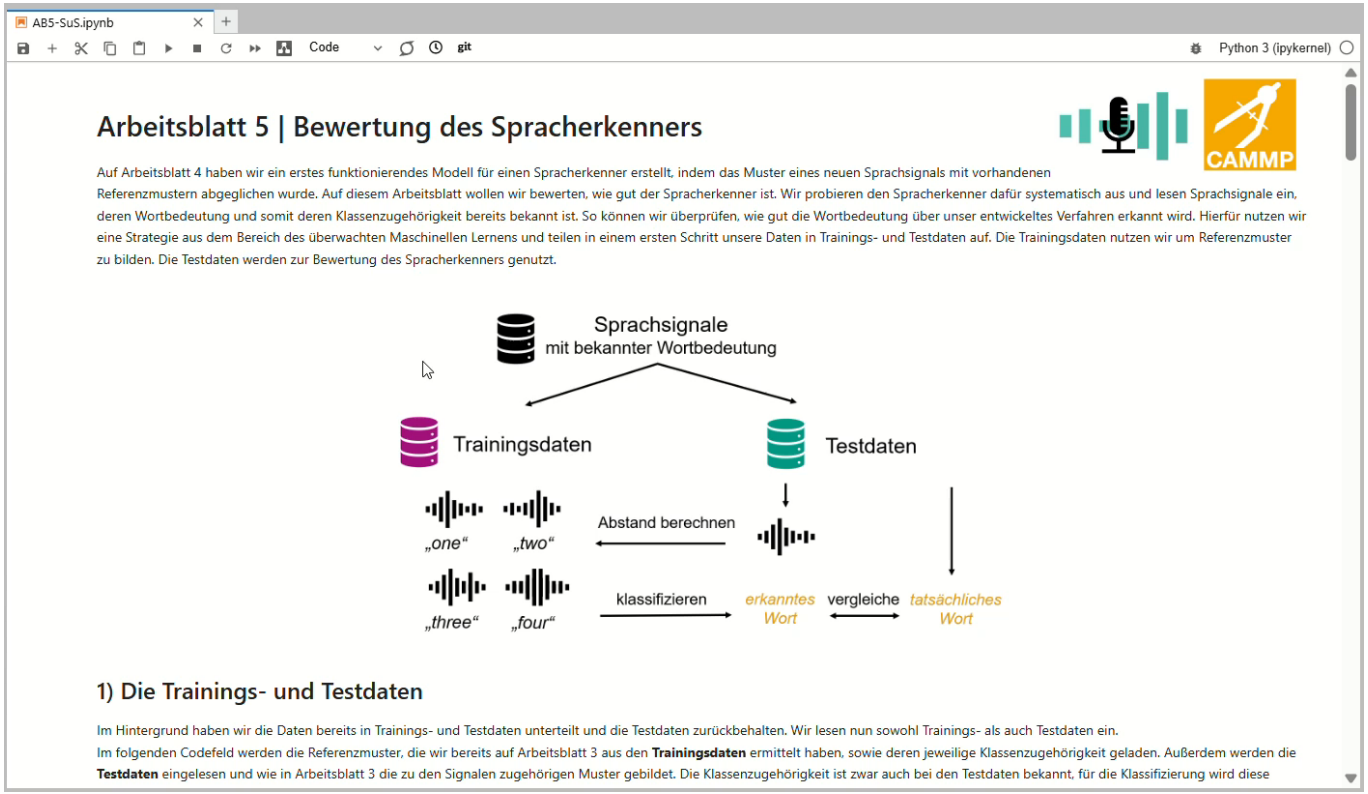
JNB in der Lehre - Vielseitigkeit der JNB

- offene Aufgabenformate
- exploratives, authentisches wissenschaftliches Arbeiten
- Programmierkenntnisse notwendig

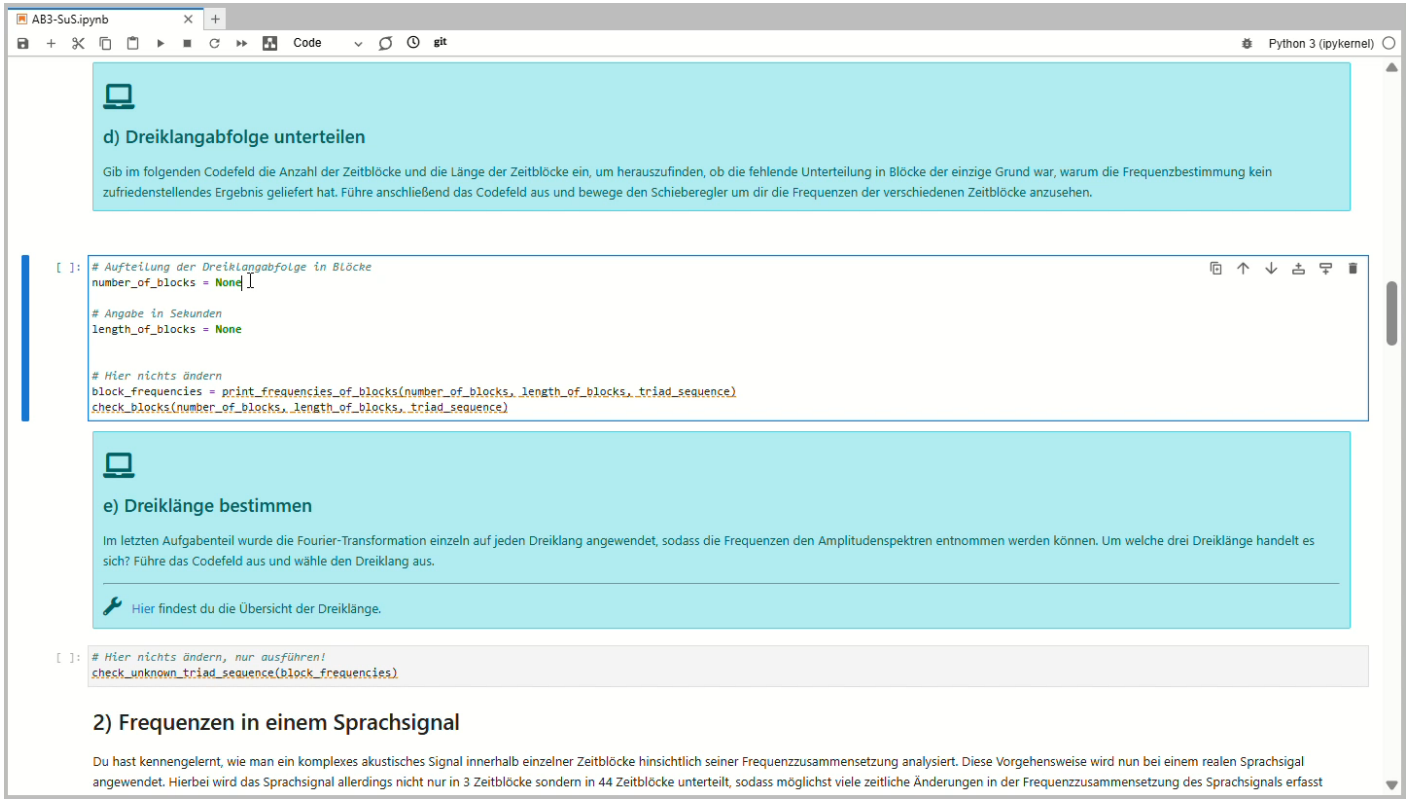
- geschlossene Aufgabenformate
- angeleitetes Arbeiten
- für Lerngruppen mit unterschiedlichen Programmierkenntnissen einsetzbar
- Unterstützung des Lernprozesses auf verschiedenen Ebenen möglich



Bsp. 3: JNB als geführte Lernumgebung



Bsp. 3: JNB als geführte Lernumgebung



JNB in der Lehre - Benefits

Einsatz von JNB in unterschiedlichen Lernszenarien kann dazu beitragen…

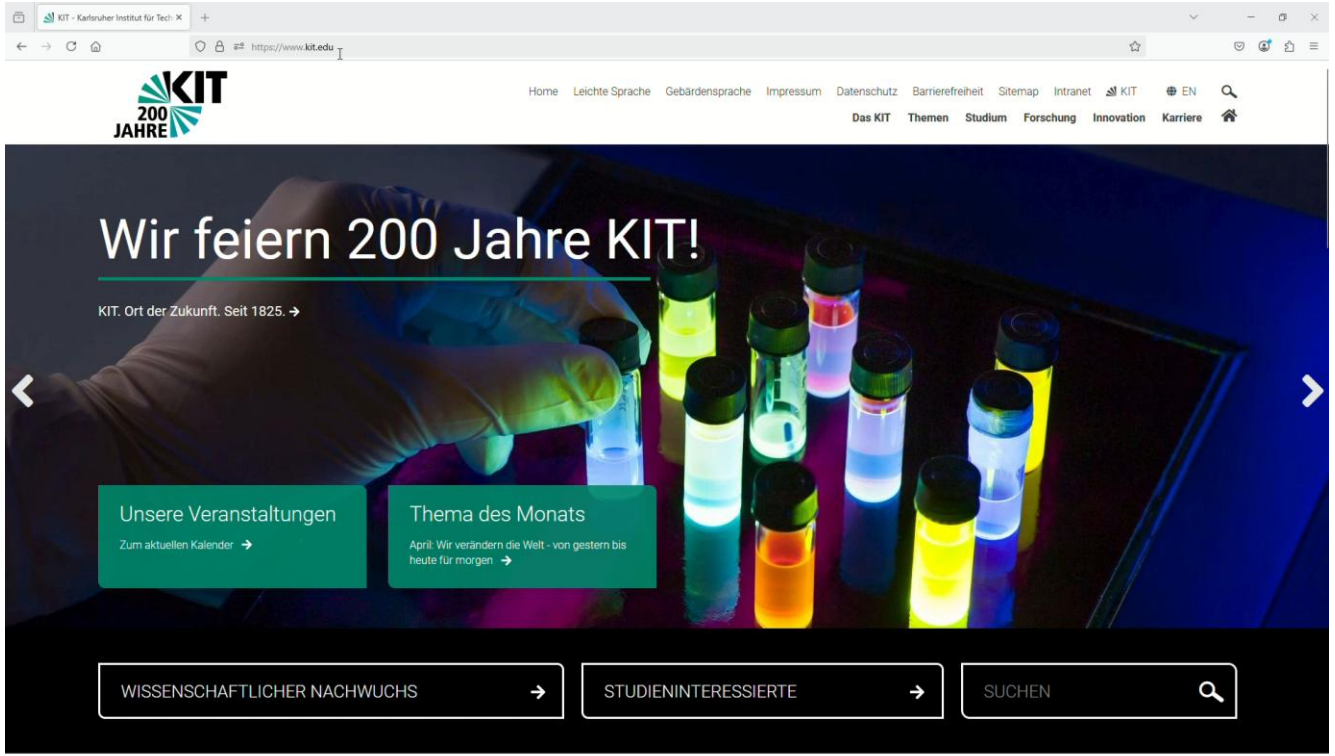
- die Beteiligung zu steigern,
- das Verständnis zu fördern,
- die Leistung zu verbessern,
- eine praxisnahe Vorbereitung auf das Berufsleben zu ermöglichen.

JupyterHub: Bereitstellung der Entwicklungsumgebung

Bisheriger Workflow:

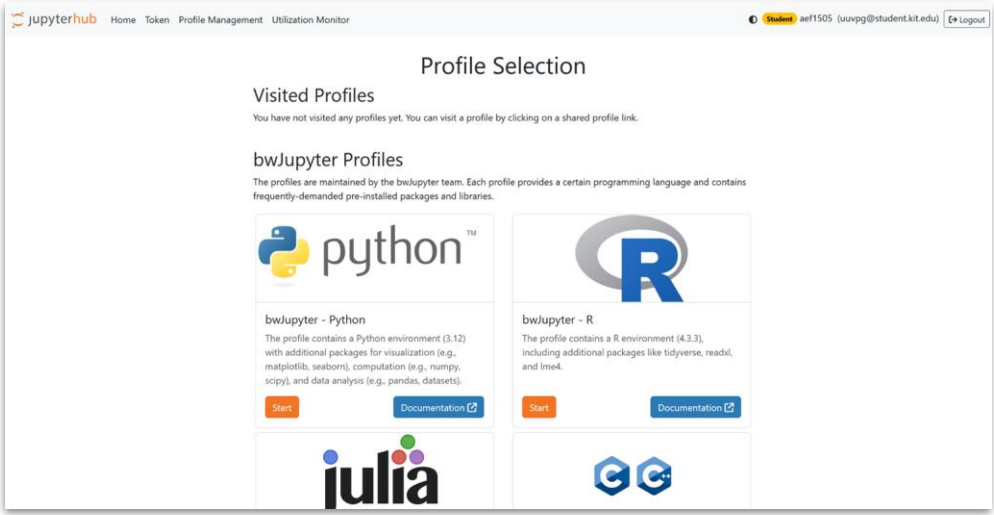
1. Python-Interpreter (Programmiersprache) auf dem PC installieren.
2. Ausführungsumgebung “Jupyter Lab” installieren.
3. Beim Lernmanagementsystem (LMS) anmelden und die Notebooks herunterladen (z.B. Übungen).
4. Benötigte Bibliotheken und Abhängigkeiten nachinstallieren.
5. Jupyter Lab starten.
6. Mit der Bearbeitung beginnen.

JupyterHub: Anmeldeprozess

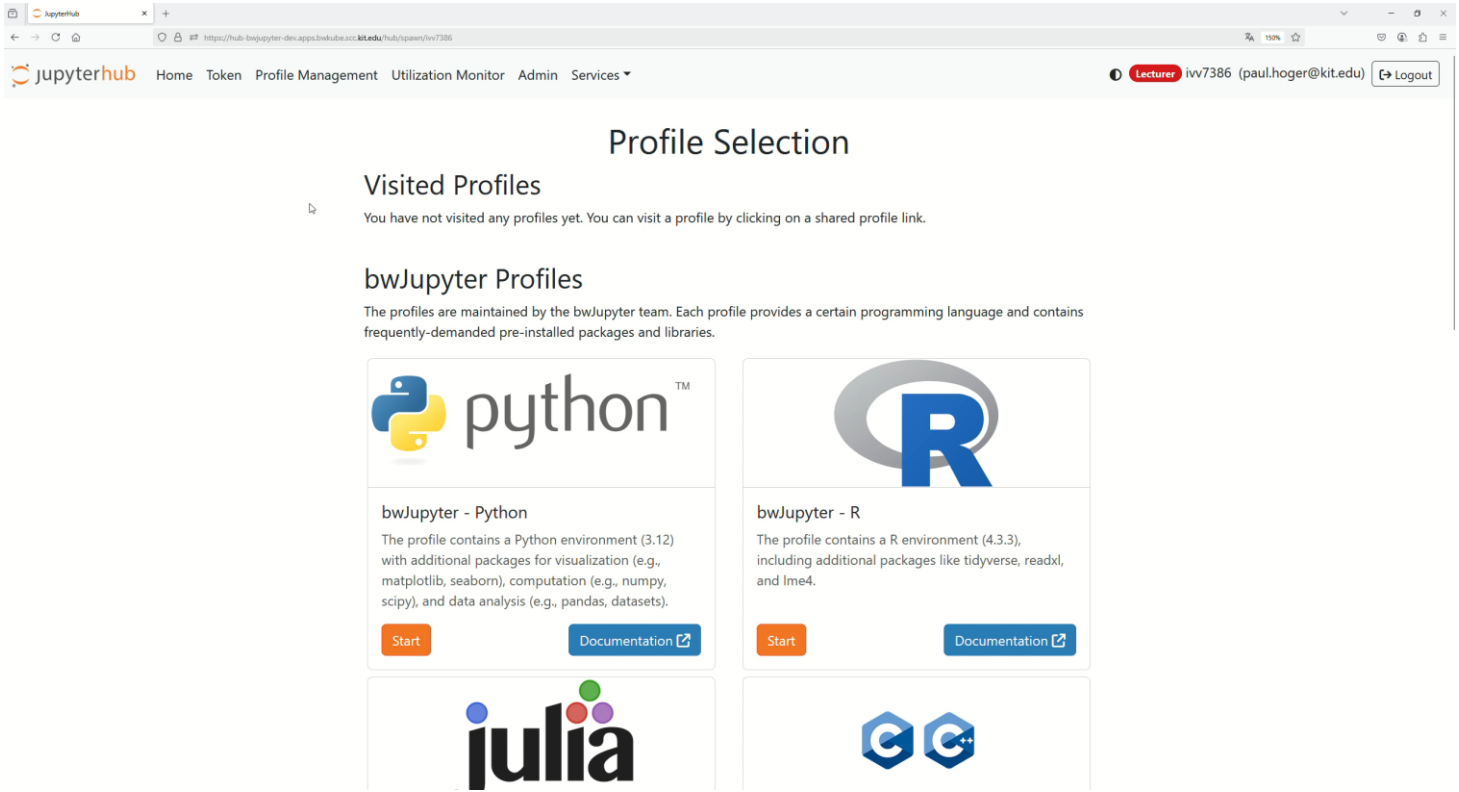


JupyterHub: Einführung in Profile

- JupyterHub nutzt sog. Profile
  - Ausführungsumgebung für Jupyter Notebooks
  - Bereitstellung der Rechenkapazitäten
- bwJupyter-Profile
  - Für die Programmiersprachen Python, R, Julia, C/C++, Java
  - Vorinstallierte Bibliotheken und Module
- Dozierende können eigene Profile anlegen
  - Vorhandenes Profil auswählen oder eine eigene Umgebung angeben
  - Nutzung einer Grafikkarte möglich
  - Jupyter Notebooks hochladen
  - Profil über einen Link mit Studierenden teilen



JupyterHub: Ein eigenes Profil anlegen



JupyterHub: Bereitstellung der Entwicklungsumgebung

Bisheriger Workflow:

1. Python-Interpreter (Programmiersprache) auf dem PC installieren.
2. Ausführungsumgebung “Jupyter Lab” installieren.
3. Beim Lernmanagementsystem (LMS) anmelden und die Notebooks herunterladen (z.B. Übungen).
4. Benötigte Bibliotheken und Abhängigkeiten nachinstallieren.
5. Jupyter Lab starten.
6. Mit der Bearbeitung beginnen.

Workflow mit bwJupyter:

1. Dozierende legen ein Profil an und laden die Notebooks hoch.
2. Studierende öffnen den geteilten Link.

Zusammenfassung:

+ Leicht zugänglich
+ Geräte- und Plattformunabhängig
+ Ressourcenverfügbarkeit
+ Geteiltes Dateisystem
+ Eigene Umgebungen

Ausblick

- Integration in die Lernmanagementsysteme (LMS)
- Übungsbetrieb (Erstellen, Ausgeben, Bearbeiten, Einsammeln, Korrigieren, Feedback verteilen) über bwJupyter abdecken
- Entwicklung und Bereitstellung von Begleitmaterialien, Vorlagen und Beispielen

Vielen Dank für Ihre Aufmerksamkeit!

bwJupyter - der JupyterHub für die Hochschullehre in BW

- Jupyter Notebooks in der Lehre einsetzen
- Einfacher Anmeldeprozess ohne manuelle Freischaltung
- Nutzung bereitgestellter Rechenressourcen
- Eigene Profile für Dozierende

https://bwjupyter.de

support@bwjupyter.de (Support, Fragen, Feedback)

Registrieren Sie sich jetzt!

https://hub.bwjupyter.de