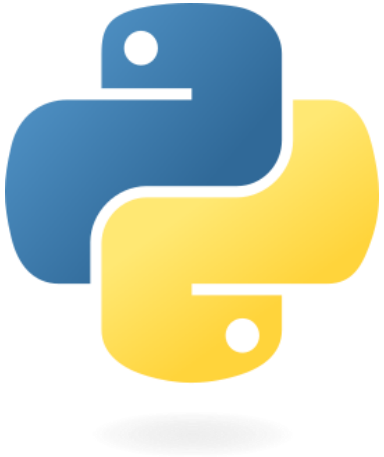
Python Programming for the Movement Sciences



M. Abedinifar, C. Hansen, R Romijnders, J. Welzel, K. Saegner, W. Maetzler

Neurogeriatrics, Department of Neurology, University Hospital Schleswig-Holstein, Kiel

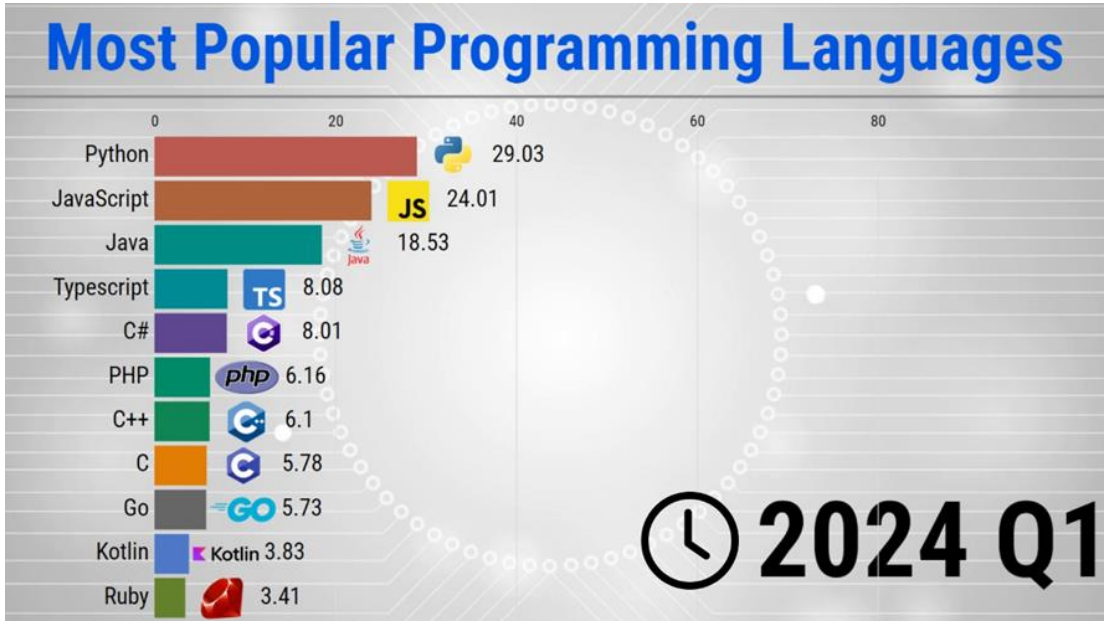
neuro GERIATRICS | KIEL

UKSH UNIVERSITÄTSKLINIKUM Schleswig-Holstein

GAMMA

Brief History of Python

- Invented in the Netherlands, early 90s by Guido van Rossum
- Named after Monty Python (a British comedy troupe)
- Open source and free from the beginning
- Considered a scripting language, but in fact it is much more
- Scalable and functional from the beginning
- Used by renowned corporation such as Google
- Increasingly popular



https://statisticsanddata.org/

Python’s Benevolent Dictator For Life*

“Python is often described as a 'batteries included' language, meaning it has a rich and versatile standard library that is immediately available to users.”

- Guido van Rossum

https://www.flickr.com/photos/niallkennedy/310814303

* It is a title given to a small number of open-source software development leaders!

Introduction to Python

Python is a **high-level**, **interpreted programming** language known for its **simplicity** and **readability**.

Some Key Features:

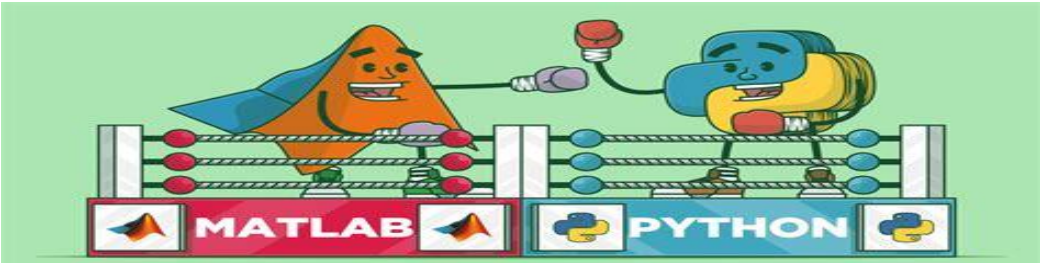
- **Easy to Learn:**

  Python's syntax is clear and easy which makes it an excellent choice for beginners.
- **Extensive Libraries and Frameworks:**

  Python boasts a rich ecosystem of libraries and frameworks that enhance its functionality.
- **Strong Community Support:**

  Python has a large active community which provide resources such as documentation, tutorials, and forums
- **Open-source:**

  Python is cost-effective than proprietary tools such as MATLAB.



Python Packages

Packages are collections of modules providing related functionalities which makes code organization easier and promoting reusability.

Commonly used packages:

- **NumPy:**

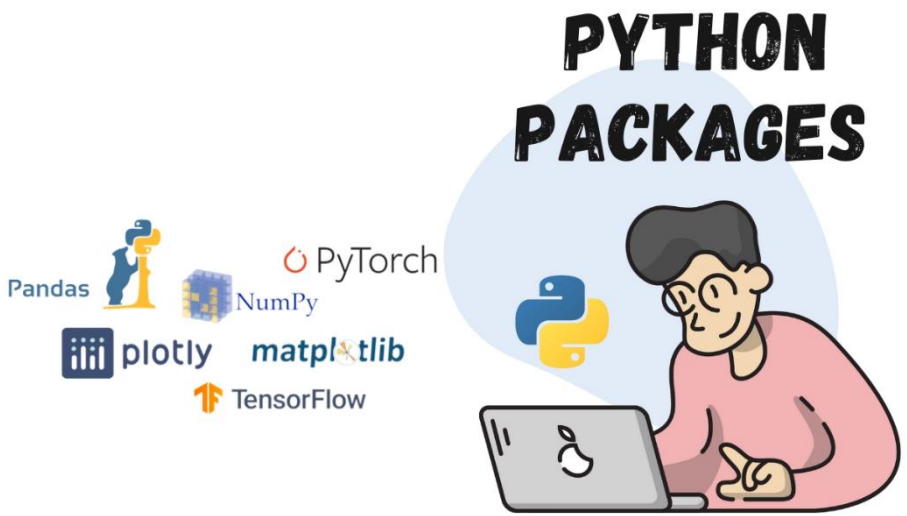
  For numerical computing and handling arrays.
- **Pandas:**

  For data manipulation and analysis, using data frames.
- **Matplotlib and plotly:**

  For creating visualizations like plots and charts.
- **SciPy:**

  For scientific computing with advanced mathematical functions.
- **Scikit-learn:**

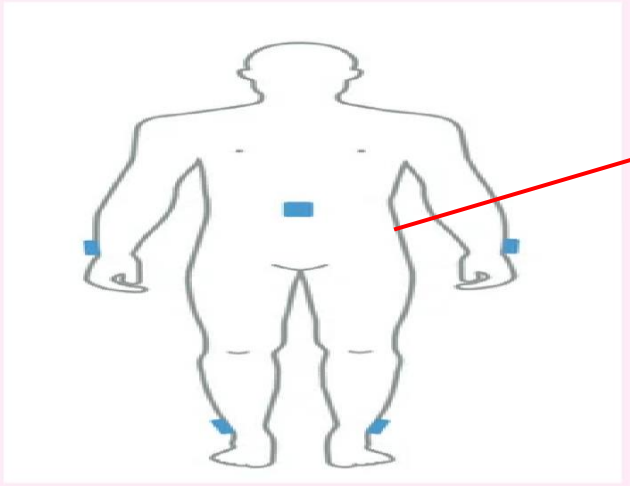
  For machine learning tasks and data analysis.



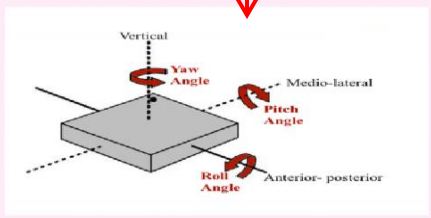
https://www.naukri.com/code360/library/python-packages

Python Applications in Movement Sciences

- Data analysis of biomechanical measurements.
- Simulation of movement patterns.
- Visualization of data to derive insights and communicate findings effectively.
- Development of algorithms for motion tracking and analysis.

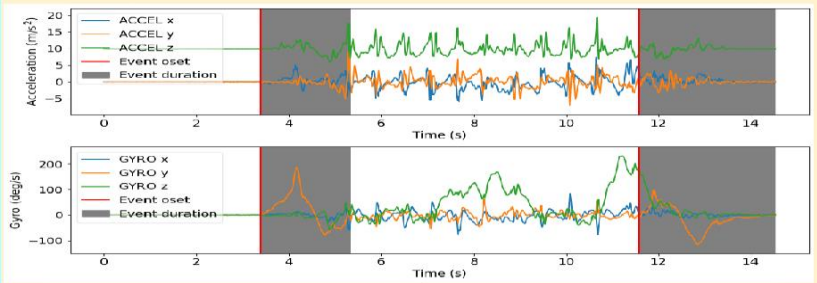


Movement Data Acquisition Platform using Wearable Sensors



Inertial Measurement Unit (IMU)

Data Transmission

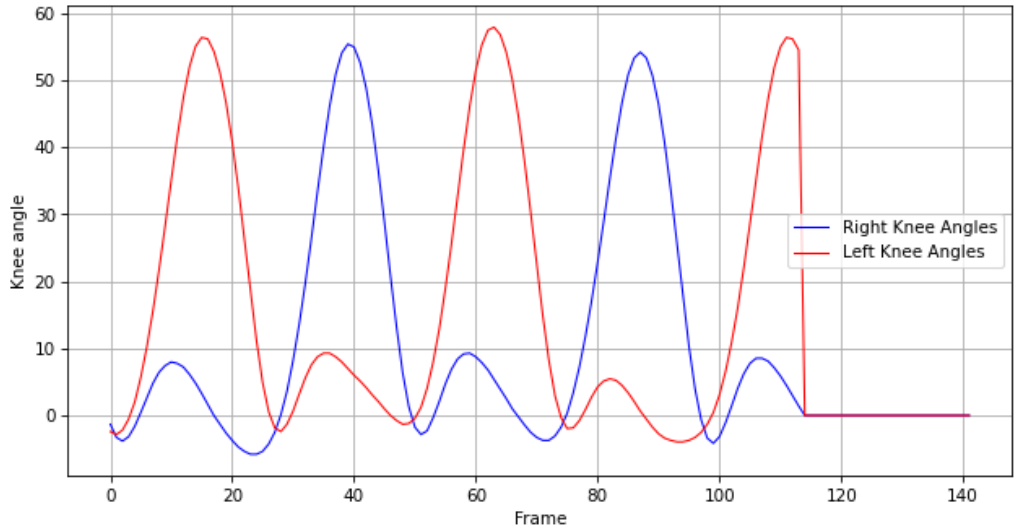


Data Processing and Movement Pattern Detection

What looks like the movement data?

Sample recording from a motion capture system:

- A crucial starting point is the ability to load standard motion capture file types such as c3d files.
- A c3d file is a data file saved in the Coordinate 3D format, which contains biomechanical information.
- The c3d file stores three-dimensional motion capture data in a compact binary structure
- We need an easy way to read (and maybe write) these files for further analysis

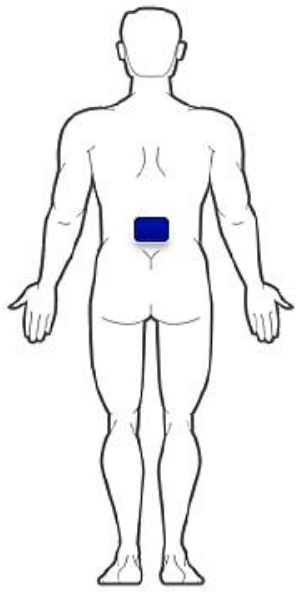


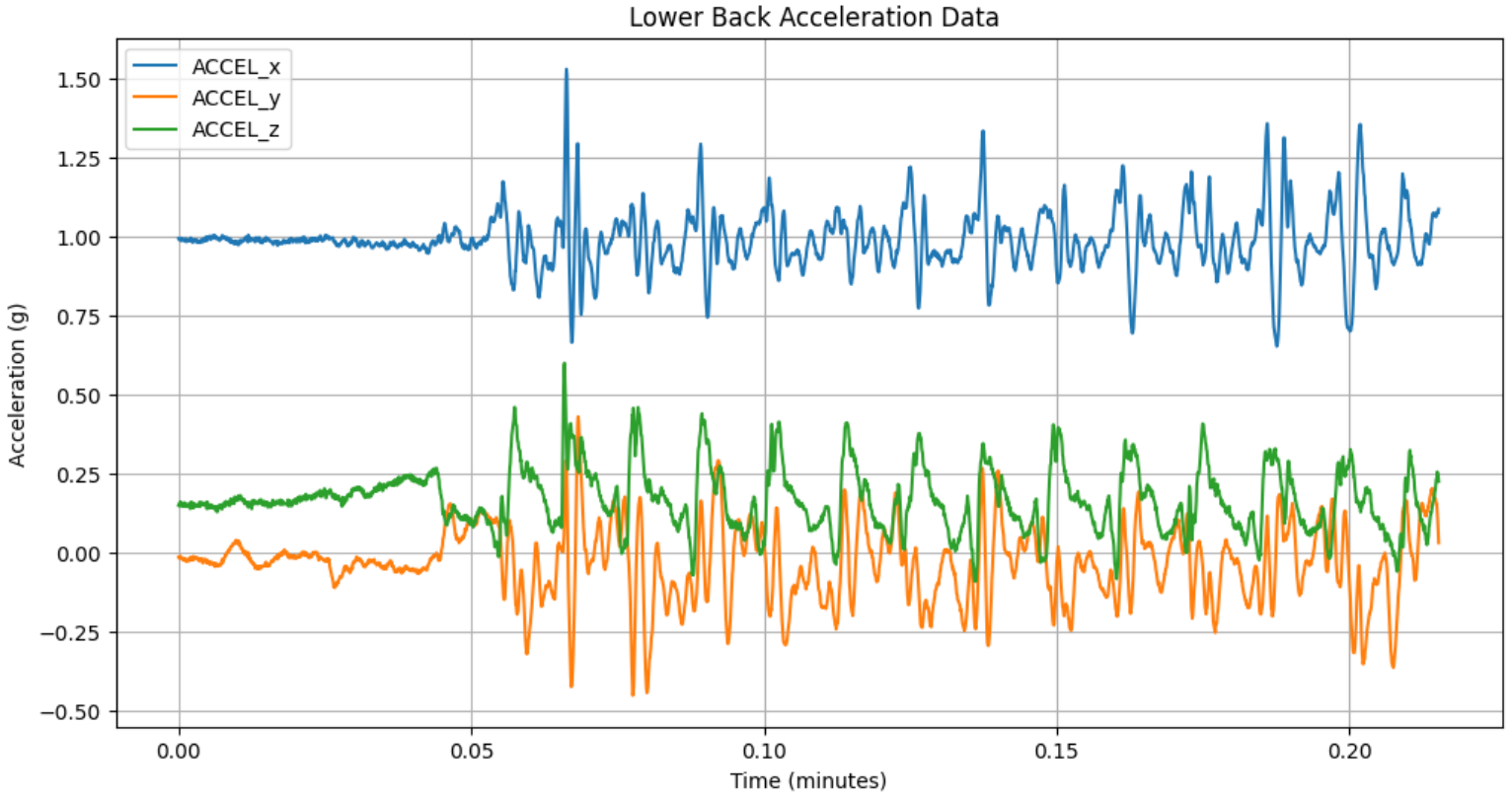
Data loaded using *c3d* package in Python

What looks like the movement data?

Sample recording from a IMU sensor:

Data acquired using wearable IMU sensor placed on lower back, performing slow walking (Warmerdam et al., 2022)





Working With Movement Data

Analysis of data using some open-source packages

- The gait sequence detection algorithm was applied using Python-based Kiel Motion Analysis Toolbox (KielMAT)
- The algorithm pre-processes the acceleration data (magnitude calculation, filtering, etc.) and detect the gait sequences within signal

Home

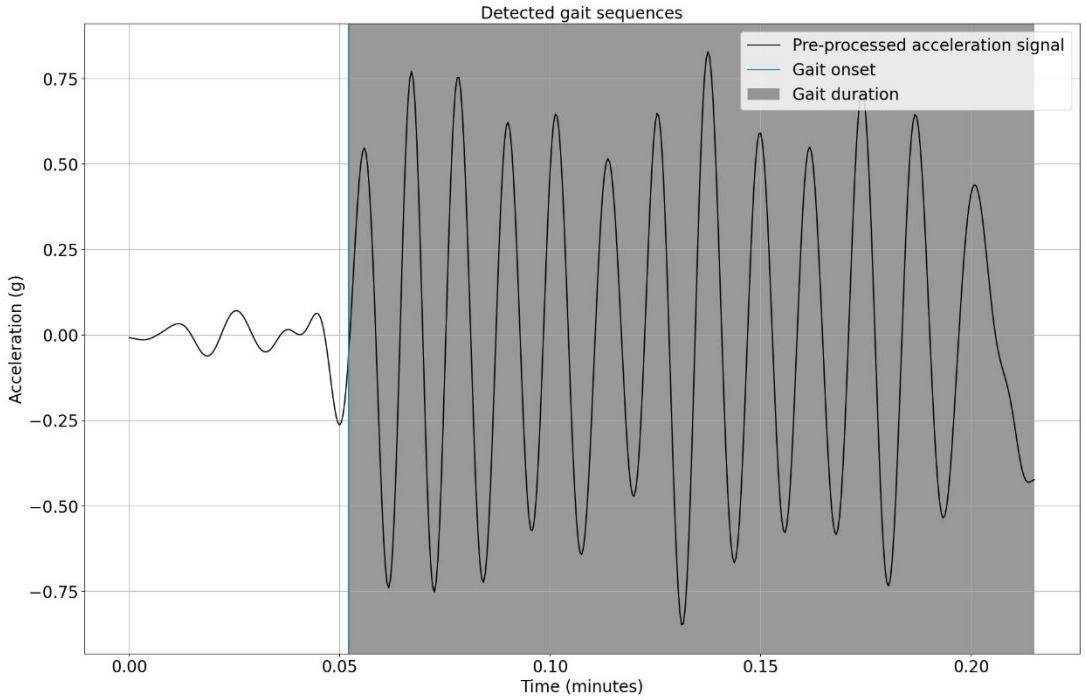
codecov 87% | build docs passing | code style black | open issues 9 | contributors 4

lint-and-test passing | python 3.10 | 3.11

KIELMAT

Welcome to the KielMotionAnalysisToolbox (KielMAT). We are a Python based toolbox for processing motion data.

https://neurogeriatricskiel.github.io/KielMAT/



What is KielMAT?

An *open-source* Python-based toolbox for processing human motion data

- It is published in Journal of Open Source Software (JOSS)
- Comprises comprehensive suite of algorithms such as gait sequence detection, etc.
- Can be used for identifying patterns in motion data across various time scales
- Can be applied both in clinical and in home assessment environments
- Includes examples and documentations demonstrating functionality of algorithms

KielMAT: Kiel Motion Analysis Toolbox - An Open-Source Python Toolbox for Analyzing Neurological Motion Data from Various Recording Modalities

Masoud Abedinifar[*1], Julius Welzel[†1], Clint Hansen[1], Walter Maetzler[1], and Robbin Romijnders[1¶]

1 Neurogeriatrics, Department of Neurology, University Hospital Schleswig-Holstein (UKSH), Kiel Germany ¶ Corresponding author

DOI: 10.21105/joss.06842

Software
- Review
- Repository
- Archive

Editor: Kevin M. Moerman
Reviewers:
- @aaa34169
- @davidpagnon

Submitted: 12 April 2024
Published: 02 October 2024

Summary

The Kiel Motion Analysis Toolbox (KielMAT) is an open-source Python-based toolbox designed for processing human motion data, following open-science practices. KielMAT offers a range of algorithms for the processing of motion data in neuroscience and biomechanics and currently includes implementations for gait sequence detection, initial contact detection, physical activity monitoring, sit to stand and stand to sit detection algorithms. These algorithms aid in identifying patterns in human motion data on different time scales. The KielMAT is versatile in accepting motion data from various recording modalities, including IMUs that provide acceleration data from specific body locations such as the pelvis or wrist. This flexibility allows researchers to analyze data captured using different hardware setups, ensuring broad applicability across studies. Some of the toolbox algorithms have been developed and validated in clinical cohorts, allowing extracted patters to be used in a clinical context. The modular design of KielMAT allows the toolbox to be easily extended to incorporate relevant algorithms which will be developed in the research community. The toolbox is designed to be user-friendly and is accompanied by a comprehensive documentation and practical examples, while the underlying data structures build on the Motion BIDS specification (Jeung et al., 2024). The KielMAT toolbox is intended to be used by researchers and clinicians to analyze human motion data from various recording modalities and to promote the utilization of open-source software in the field of human motion analysis.

Statement of need

Physical mobility is an essential aspect of health, as impairment in mobility is associated with reduced quality of life, falls, hospitalization, mortality, and other adverse events in many chronic conditions. Traditional mobility measures include patient-reported outcomes, objective clinical assessments, and subjective clinical assessments. These measures are linked to the perception and capacity aspects of health, which often fail to show relevant effects on daily function at an individual level (Maetzler et al., 2021). Perception involves surveys and patient-reported outcomes that capture how individuals feel about their own functional abilities, while capacity refers to clinical assessments of an individual's ability to perform various tasks. To complement both patient-reported (perception) and clinical (capacity) assessment approaches, digital health technology (DHT) introduces a new paradigm for assessing daily function. By using

*co-first author
†co-first author

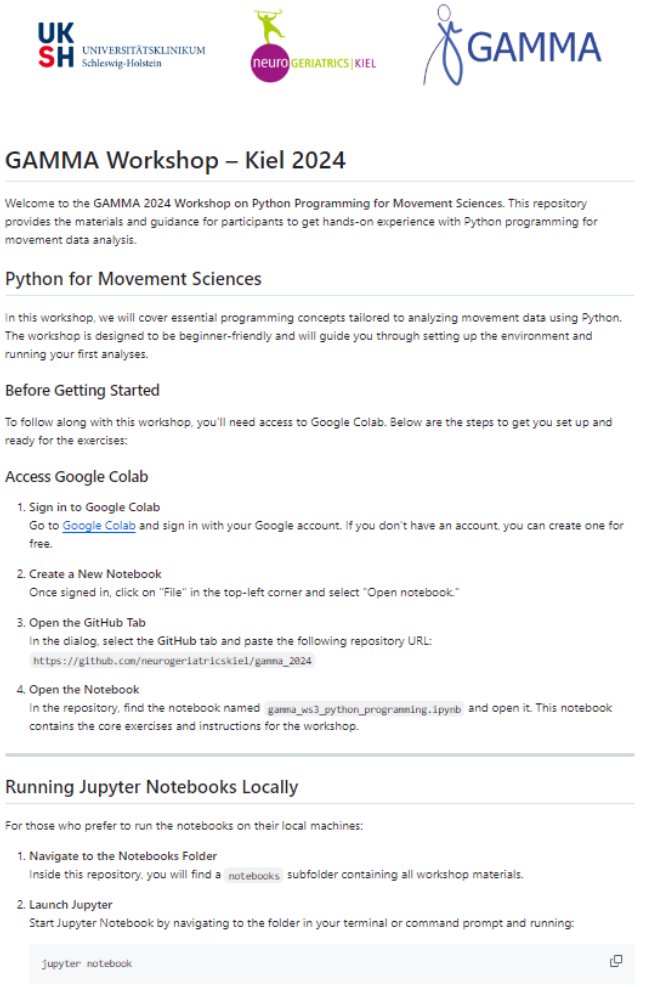
https://neurogeriatricskiel.github.io/KielMAT/

What to expect next:

- Basics of Python scripting
- Primary data types, collections, control structures, functions, etc.
- Essential packages for movement data analysis
- Loading and visualizing different movement data types such as data from motion capture systems, and wearable sensors such as IMUs.
- Analysis of movement data using open-source Python packages such as KielMAT, Gaitmap, etc.

To reach out the materials, you may visit the **GAMMA** workshop repository:

https://github.com/neurogeriatricskiel/gamma_2024



GAMMA Workshop – Kiel 2024

Welcome to the GAMMA 2024 Workshop on Python Programming for Movement Sciences. This repository provides the materials and guidance for participants to get hands-on experience with Python programming for movement data analysis.

Python for Movement Sciences

In this workshop, we will cover essential programming concepts tailored to analyzing movement data using Python. The workshop is designed to be beginner-friendly and will guide you through setting up the environment and running your first analyses.

Before Getting Started

To follow along with this workshop, you'll need access to Google Colab. Below are the steps to get you set up and ready for the exercises:

Access Google Colab

1. Sign in to Google Colab
   Go to Google Colab and sign in with your Google account. If you don't have an account, you can create one for free.
2. Create a New Notebook
   Once signed in, click on "File" in the top-left corner and select "Open notebook."
3. Open the GitHub Tab
   In the dialog, select the GitHub tab and paste the following repository URL:
   `https://github.com/neurogeriatricskiel/gamma_2024`
4. Open the Notebook
   In the repository, find the notebook named `gamma_ws3_python_programming.ipynb` and open it. This notebook contains the core exercises and instructions for the workshop.

Running Jupyter Notebooks Locally

For those who prefer to run the notebooks on their local machines:

1. Navigate to the Notebooks Folder
   Inside this repository, you will find a `notebooks` subfolder containing all workshop materials.
2. Launch Jupyter
   Start Jupyter Notebook by navigating to the folder in your terminal or command prompt and running:

```
jupyter notebook
```