

# Data Science Training Getting Started with Python & NumPy



# This training was developed for you by Data Science @ SE GP G QPI QM P Improvement Projects

## We help your employees getting the skills for the digital future:

- Global Data Science Skill Network (with Matt Bryan, LGT R&D)
- Data Science and Machine Learning trainings (with several partners):
  - Currently 11 trainings developed and executed
  - Attended by >500 participants mostly from R&D
- Analytic services in specific improvement projects
- EAI CodeShare (with Thomas Buller, LGN R&D)

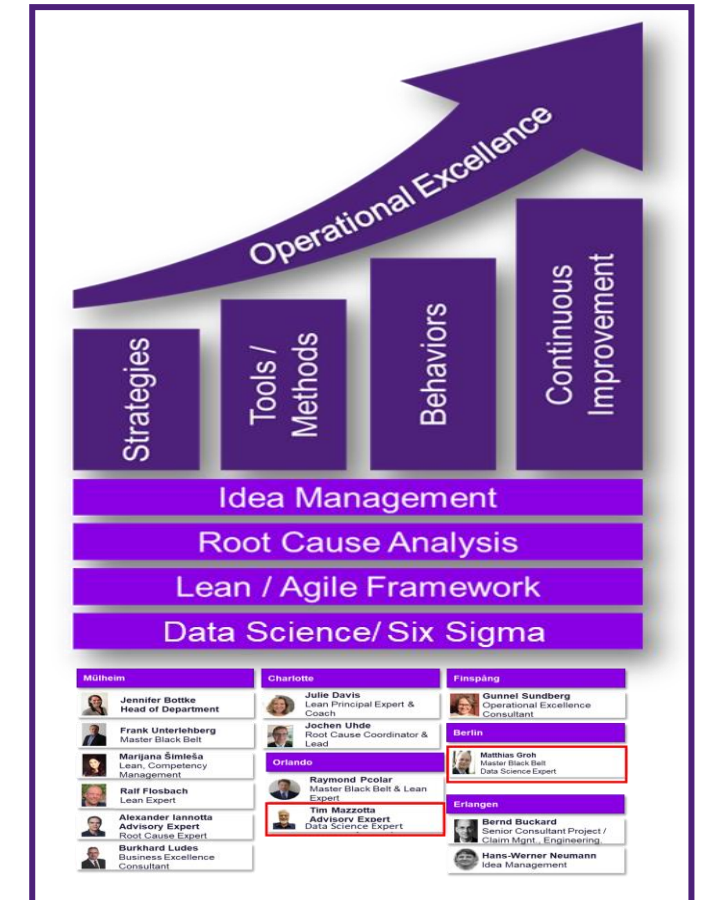
### Data Science Lead Developers @ QPI QM P:



**Matthias Groh**  
Six Sigma Master Black Belt  
Data Science Consultant



**Tim Mazzotta**  
Advisory Expert  
Data Science Expert



We help building the bridge between Data Science and daily engineering business.

# How This Training Works

- Experiment, we are all learning
- DS0-> DS1->DS2->DS3 are prerequisites (enough to read through scripts)
- New For DS2: "Transforming Data" = free chapter 1 of "Data Manipulation with Pandas" <https://learn.datacamp.com/courses/data-manipulation-with-pandas>
- 1x15 Min Break in each 2h block
- Recording: high demand, few training seats
- Based on interaction
  - Discussions, Exercises + Debriefs
  - Harder in Teams
  - If I should not call on you, please tell us now



# Introduction

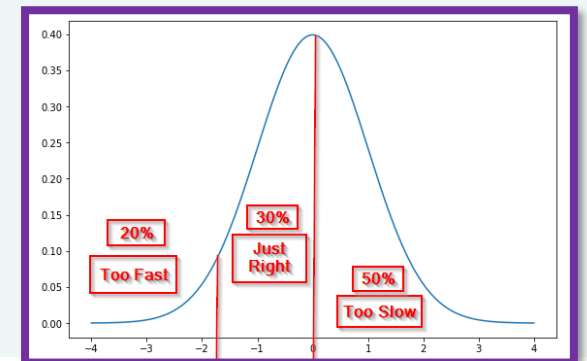
1. Name, Department
2. All prerequisites fulfilled? EnergyAI, knowledge of previous trainings (DS0->DS1->DS2->DS3)
3. Preexisting knowledge program, data science, statistics, Python, R, ...
4. Expectations

# The training



# Goal Data Science Workshops: Basic understanding how to use Python and EnergyAI to access power plant data

Workshop Module	Agenda	Goals	Our approach to teaching:
<b>DS 1 Python Basics</b>	<ul style="list-style-type: none"> <li>Jupyter Notebook Intro</li> <li>Lists, Object Localization</li> <li>Loops, Dictionaries, If</li> <li>Definitions</li> <li>Classes/ Objects/ Methods</li> <li>Intro to NumPy/ ND-Arrays</li> </ul>	<b>Target audience (all modules):</b> <ul style="list-style-type: none"> <li>Any engineer who needs to analyze big data</li> <li>In particular engineers verifying assumptions with power plant data</li> </ul>	<b>Handling of questions:</b> <ul style="list-style-type: none"> <li>For understanding questions ask right away</li> <li>For major problems go to separate helpline Teams meeting: one of the teachers will join you as a coach</li> <li>Expert discussions moved to break or afterwards</li> </ul> <b>With a wide range of participant's knowledge:</b> <ul style="list-style-type: none"> <li>Target speed is for 20% too fast / for 50% too slow</li> </ul>
<b>DS2 Python: Pandas and Seaborn</b>	<ul style="list-style-type: none"> <li>Recap</li> <li>Handling Nans</li> <li>Resample and Fill</li> <li>Calculation with dataframes</li> <li>PickleFormat</li> <li>Connecting dataframes</li> <li>Intro Visualization with Seaborn</li> </ul>	<b>Prerequisites:</b> <ul style="list-style-type: none"> <li>Basic programming knowledge or DS0</li> <li>DS1 for DS2, DS2 for DS3 or reading through the scripts</li> <li>New For DS2: "Transforming Data" = free chapter 1 of "Data Manipulation with Pandas"  <a href="https://learn.datacamp.com/courses/data-manipulation-with-pandas">https://learn.datacamp.com/courses/data-manipulation-with-pandas</a> </li> </ul>	
<b>DS 3 Data Pulling in Energy AI</b>	<ul style="list-style-type: none"> <li>Intro Data Pulling</li> <li>Time Interval</li> <li>Plants, Units and Signals</li> <li>Period, Aggregation</li> <li>Fill, Thresholds, Poststresholds</li> <li>Plotting with Seaborn (cont.)</li> <li>Examples</li> </ul>	<b>Teaching goals (all modules):</b> <ul style="list-style-type: none"> <li>First steps to learn the topic is taken.</li> <li>Basic concepts are understood -&gt; just enough to keep learning from online sources.</li> <li>Everybody learns in the workshop, also trainers.</li> <li>This is a teaching "Minimal Viable Product"</li> </ul>	



# Our Focus in the Digitalization World is Data Science



Our Focus

## Data Science

- Focus on quantitative data

For Example:

- Power plant sensor data
- Manufacturing data
- Component design analysis data (FE, CFD, CHT, ...)
- Test rig data

## Business Intelligence

- Focus on qualitative data

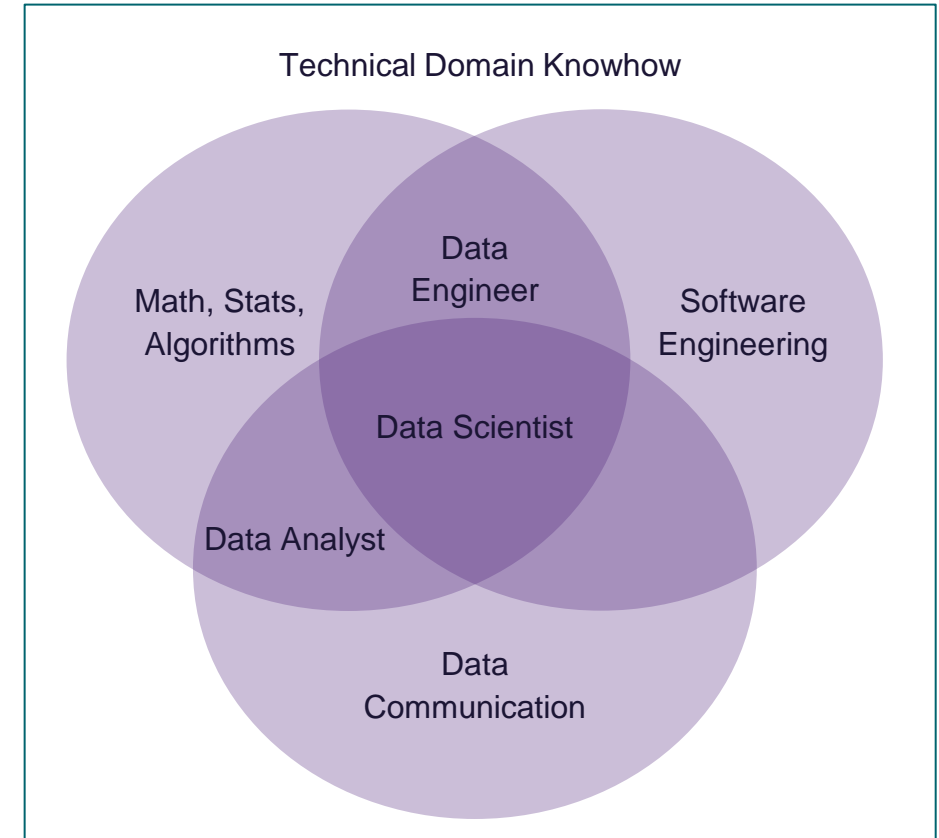
For Example:

- Manufacturing and supply data in SAP
- Qualitative fleet data (Fleet Intelligence)
- PCM

**Of course everybody needs a bit of both data types, but without focusing, no one gets anywhere...**

# Definition Data Science

Data science is a multi-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from structured and unstructured data. [1]



**Data science includes “new” (Machine Learning) and “old” (conventional analytics).**

[1] [en.wikipedia.org/wiki/Data\\_science](https://en.wikipedia.org/wiki/Data_science)



# Python - a short overview



# Basic Elements - Data Structures in Python (Grey), Numpy (Green) and Pandas (Orange)

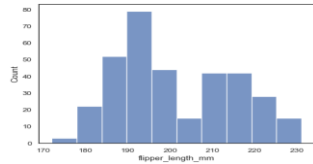
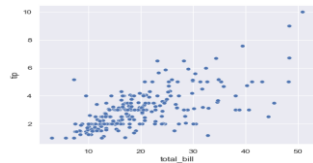
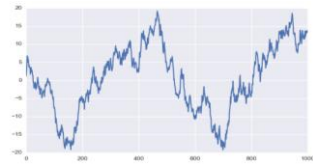
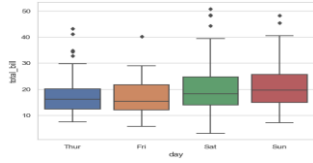
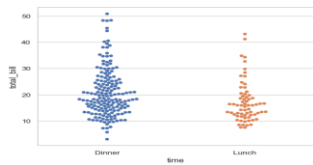
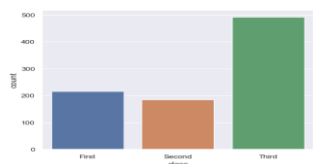
	Description	Useful for...	Example
Lists	<ul style="list-style-type: none"> <li>Store collection of heterogeneous items – immutable and mutable objects</li> </ul>	<ul style="list-style-type: none"> <li>Simple listing</li> <li>Special programming techniques (Stacks, Queue, Graphs, Trees)</li> </ul>	[item, item, item]
Dictionaries	<ul style="list-style-type: none"> <li>Collection of key-value pairs</li> <li>Key: immutable objects</li> <li>Value: heterogeneous items – immutable and mutable objects</li> </ul>	<ul style="list-style-type: none"> <li>Creating loops to do the same operation on many dataset</li> </ul>	{key1:item1, key2:item2}
Files	<ul style="list-style-type: none"> <li>Store and retrieve previously stored information</li> </ul>		.csv, .hdf5, .xlsx
Tuples	<ul style="list-style-type: none"> <li>Tuples are data structures in which the content can not be changed after creation (no deletion, add or edit)</li> </ul>	<ul style="list-style-type: none"> <li>Prevents data manipulation</li> </ul>	(item, item, item)
Sets	<ul style="list-style-type: none"> <li>Collection of unique objects</li> </ul>	<ul style="list-style-type: none"> <li>Creating lists that only hold unique values</li> <li>Helpful when going through huge dataset</li> </ul>	{item, item, item}
Numpy ND-Arrays	<ul style="list-style-type: none"> <li>Store collection of data of the same type</li> </ul>	<ul style="list-style-type: none"> <li>Dealing with large collection of homogeneous data types: easier to use, faster and uses lesser memory than lists</li> <li>Support vectorized operations</li> <li>Work efficiently with large datasets with lots of empty cells</li> </ul>	[[1. 1. 2.] [3. 5. 8.] [5. 3. 2.]]
Dataframes	<ul style="list-style-type: none"> <li>2-dimensional labeled data structure</li> <li>Columns don't have to have the same data type</li> </ul>	<ul style="list-style-type: none"> <li>Data mining / manipulation</li> <li>Labeling</li> <li>Multiindexing</li> </ul>	Like this table
Series	<ul style="list-style-type: none"> <li>One-dimensional labeled array capable of holding any data types</li> </ul>	<ul style="list-style-type: none"> <li>Axis labels</li> </ul>	a 1 b 2 c 3

Source: [www.datacamp.com](http://www.datacamp.com), [pandas.pydata.org](http://pandas.pydata.org); Also see: <https://www.datacamp.com/community/tutorials/data-structures-python>

# A Selection of Important Libraries in Python

Category	Name	Description	Link
Data Handling Libraries	Numpy	Basic Library for scientific computing in Python (linear algebra, numerical functionalities etc.)	<a href="https://numpy.org/">https://numpy.org/</a>
	Pandas	Package especially built for data analysis	<a href="https://pandas.pydata.org/">https://pandas.pydata.org/</a>
	Scipy	Mathematical and engineering functions (e.g. optimization and fits, numerical integration...)	<a href="https://www.scipy.org/">https://www.scipy.org/</a>
Plotting Libraries	Matplotlib	2D plotting library for basic plotting	<a href="https://matplotlib.org/">https://matplotlib.org/</a>
	Seaborn	Is built on top of matplotlib, thus offers more functionalities	<a href="https://seaborn.pydata.org/">https://seaborn.pydata.org/</a>
	Plotly	Is built for interactive graphs	<a href="https://plot.ly/python/">https://plot.ly/python/</a>
Machine Learning Libraries	TensorFlow	For machine learning applications (e.g. voice recognition)	<a href="https://www.tensorflow.org/">https://www.tensorflow.org/</a>
	SciKit-Learn	For machine learning applications (e.g. regression models, clustering, vector-machines...)	<a href="https://scikit-learn.org/stable/">https://scikit-learn.org/stable/</a>
Other Useful Libraries	Time	Provides all time-related functions	<a href="https://docs.python.org/3/library/time.html">https://docs.python.org/3/library/time.html</a>
	Sys	For interaction with interpreter (advanced)	<a href="https://docs.python.org/3/library/sys.html">https://docs.python.org/3/library/sys.html</a>

# Overview: Essential Graphs of Seaborn

Name	In SNS	Useful for	Plot	Relative to ...	Target variable
Histogram	<code>.histplot()</code>	<ul style="list-style-type: none"><li>▪ Central tendency</li><li>▪ Spread</li><li>▪ Distribution</li><li>▪ Outliers</li></ul>			Continuous data
Scatterplot	<code>.relplot()</code> X-Axis is continuous	<ul style="list-style-type: none"><li>▪ Correlation</li><li>▪ Outliers</li></ul>		... other continuous data	
Lineplot		<ul style="list-style-type: none"><li>▪ Time effect</li><li>▪ Spread</li><li>▪ Outliers</li></ul>		... time	
Boxplot	<code>.catplot()</code> X-Axis is categorical	<ul style="list-style-type: none"><li>▪ Group effects</li><li>▪ Central tendency</li><li>▪ Spread</li><li>▪ Distribution</li><li>▪ Outliers</li></ul>		... categorical data	
Swarmplot					
Countplot			<ul style="list-style-type: none"><li>▪ Counts group effects</li></ul>		... categorical data

# If you want to use Python at SE after this class...

Power and Gas – Large Gas Turbines, Generators



# The Classical Package for Python: Anaconda (not an official SE-software for download, but often just downloaded anyway)

<https://www.anaconda.com/distribution/>



Individual Edition

## Your data science toolkit

With over 20 million users worldwide, the open-source Individual Edition (Distribution) is the easiest way to perform Python/R data science and machine learning on a single machine. Developed for solo practitioners, it is the toolkit that equips you to work with thousands of open-source packages and libraries.

Download



### Open Source

Anaconda Individual Edition is the world's most popular Python distribution platform with over 20 million users worldwide. You can trust in our long-term commitment to supporting the Anaconda open-source ecosystem, the platform of choice for Python data science.



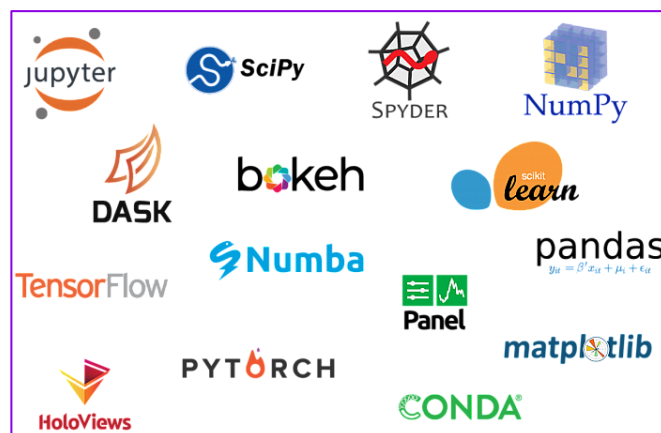
### Conda Packages

Search our cloud-based repository to find and install over 7,500 data science and machine learning packages. With the conda-install command, you can start using thousands of open-source Conda, R, Python and many other packages.



### Manage Environments

Individual Edition is an open source, flexible solution that provides the utilities to build, distribute, install, update, and manage software in a cross-platform manner. Conda makes it easy to manage multiple data environments that can be maintained and run separately without interference from each other.



## Build machine learning models

Build and train machine learning models using the best Python packages built by the open-source community, including scikit-learn, TensorFlow, and PyTorch.

## Anaconda Installers

### Windows

Python 3.8  
64-Bit Graphical Installer (457 MB)  
32-Bit Graphical Installer (403 MB)

### MacOS

Python 3.8  
64-Bit Graphical Installer (435 MB)  
64-Bit Command Line Installer (428 MB)

### Linux

Python 3.8  
64-Bit (x86) Installer (529 MB)  
64-Bit (Power8 and Power9) Installer (279 MB)

# A good and “correct” way around Anaconda at ES is EnergyAI:

Apply at: <https://energyai.siemens-energy.cloud/>

For problems contact: [contact@energyai.siemens-energy.cloud](mailto:contact@energyai.siemens-energy.cloud)

## Why EnergyAI?

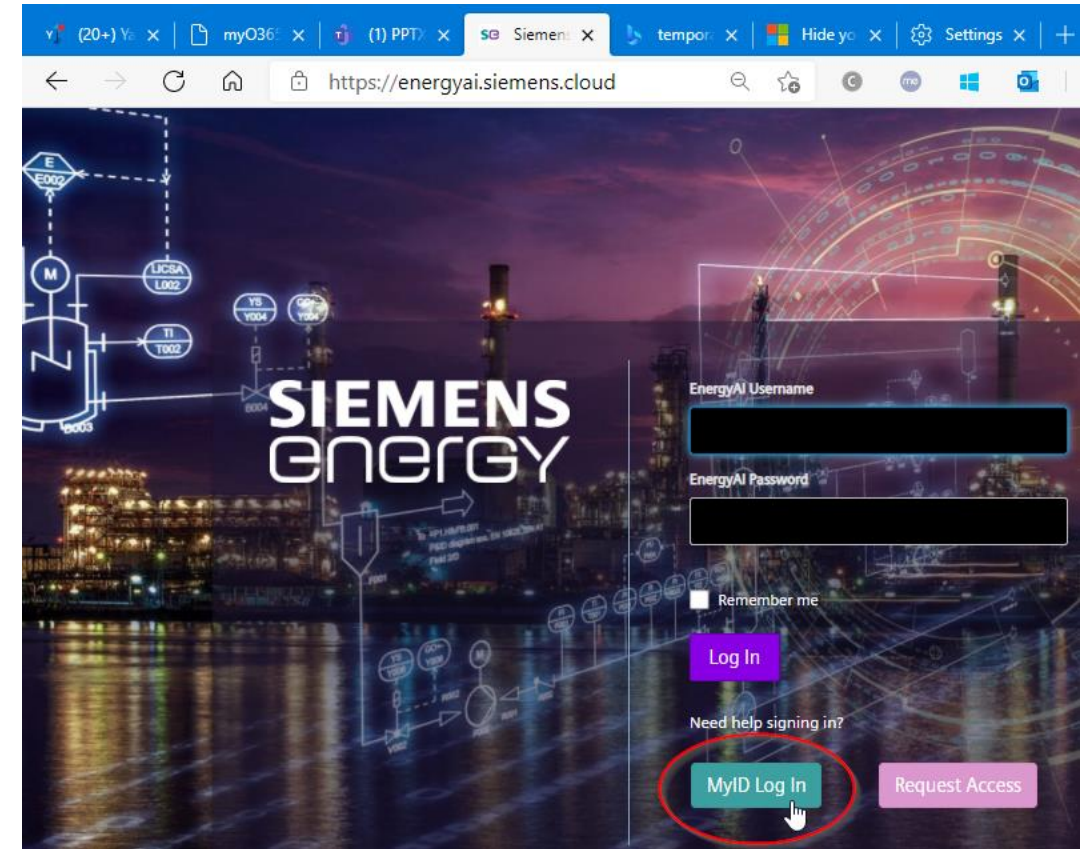
The EnergyAI - Analysis platform...

- ...offers an efficient interface to handle large data requests on multiple plants or units

- ...enables the user to perform postprocessing on-the-fly, without further data handling steps

- ...provides a standardized GPN System to address signals across all plants and units

- ...executes all operations outsourced on a server and makes results available for client

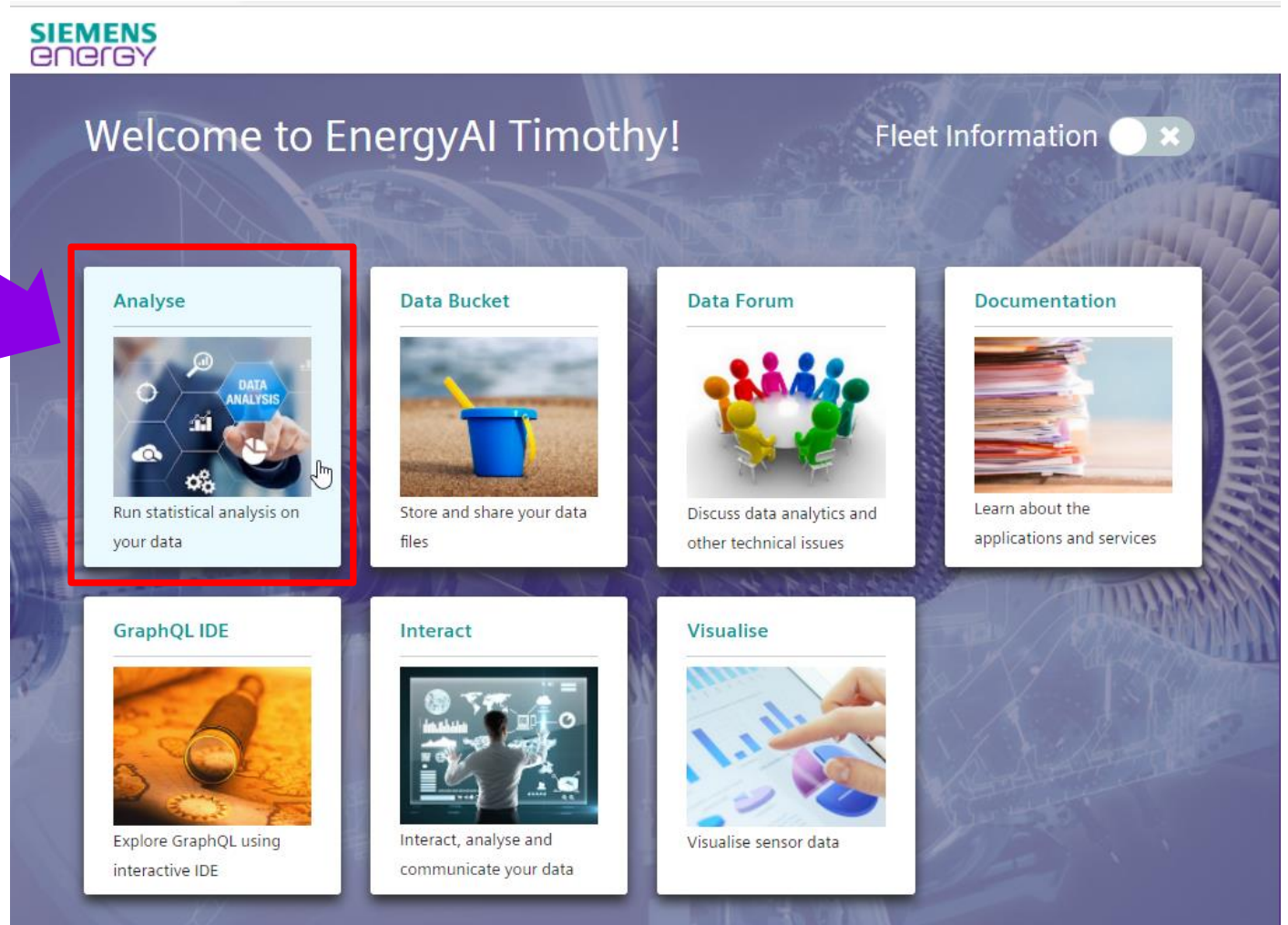
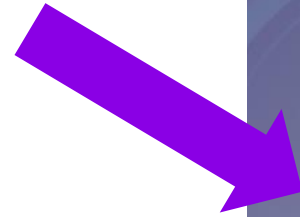




# Go to “Analyse”

<https://energyai.siemens-energy.cloud/>

<https://analyse.energyai.siemens-energy.cloud/>





# Analyse Layout

The screenshot displays the JupyterLab interface. On the left is the **File System** sidebar, which includes a search bar and a file list. The main area is divided into three horizontal sections: **Notebook**, **Console**, and **Other**. The **Notebook** section contains icons for Python 3 (ipykernel) and R. The **Console** section also contains icons for Python 3 (ipykernel) and R. The **Other** section contains a row of icons for various tools: Terminal, Text File, Markdown File, Python File, R File, Scheduler, Shiny, Dash, Bokeh, TensorBoard, and MLf. Red annotations with arrows point to the Python and R icons in the Notebook and Console sections, labeled "Script-based (JupyterLab)" and "Console-based" respectively. Another red annotation with an arrow points to the Scheduler icon in the Other section, labeled "more functionalities".

File System

Script-based (JupyterLab)

Console-based

more functionalities

# First Steps – File System

- Hierarchical File System as on Windows
- In *'/shared/groups'* , it can be worked in groups (group folder has to be requested)
- In *'/shared/users'* some scripts of other users are visible (copy script in your shared folder to make it visible for others)
- Go to: *'/shared/demos'* for several demos on the usage of the datahub with R or Python
- Go to: *'/shared/spaces/CommunityCodeShare'* for EnergyAI CodeShare projects
- Repositories from GitHub can be embedded in folders to simultaneously work on scripts (recommended)

