renku Intro

Rok Roškar, Swiss Data Science Center, ETH Zürich

September 6, 2023
Renku enables “sustainable” data science

1. Done today with tomorrow in mind
2. Individual work benefits institution and community
3. All components form a functioning ecosystem
a story...

Set up the project environment... more than once

Remembering how to re-run code to update a figure...

Many practical and technical hurdles along the way from data to results...

Development

Inefficient sharing of computational resources...

Publication

1 year later, trying to share project materials...
What is Renku?

Knowledge Graph
Connecting Code, Datasets, Workflows, and Computational environments.

Versioned Code
The foundation of collaborative machine learning and data science.

Reusable Data
Configurable data packages with rich metadata.

Comprehensible Workflows
Encode relationships between code and data.

Containerized Environments
Easy access to consistent cloud or local execution environments.
Datasets
Create (choose storage), assemble, annotate, publish

- Create datasets to easily reuse and share data across projects
- Use various backends: git-LFS, S3, Azure blob, local or network storage sources
- Combine with compute environments and analysis examples to ensure data can easily be used and reused
- Record pipelines that yield or consume datasets for full traceability
Workflows

renku run my-analysis.sh

Capture workflow

Record as KG

Reuse on various backends

Optimized storage

toil on HPC

Additional via plug-ins
Automatically record pipelines

**Legend**

- **Code files**
  - code_file.py
  - Code in a programming language like python or R

- **Script execution**
  - python filename.py
  - The command which is run, typically taking input arguments and producing output results.

- **Data files**
  - data_file.ext
  - A file or folder containing data.

- **Datasets**
  - dataset name
  - A collection of files and/or folders that contain data and metadata describing information like authorship, licensing, etc.

**Acquire raw data**

- raw_data_downloader.py
- util_setup_params.py
- raw_data

**Preprocess**

- data_preprocessing.py
- python data_preprocessing.py
- preprocessed_data

**Train**

- ml_train_autoencoder.py
- python ml_train_autoencoder.py
- logging/autoencoder

**Extract Features**

- ml_extract_features_autoencoder.py
- python ml_extract_features_autoencoder.py
- feature_representation
Environment

- Easy access to shared compute and storage
- Containers for reproducibility and portability, templates for consistency
- Maintained library of images to keep things up-to-date; install apps, dashboards, desktops etc.
- Configurable access to resources
- Shared project data sources
For the user, there is NO vendor or technology lock-in
apart from git + docker
The Renku Stack

https://renkulab.io

 renku

jupyter
R Studio
MATLAB
GitLab
Python
Julia
C++

renku
command-line interface (CLI)
docker
kubernetes

The Renku Stack
Where is Renku used?

- Public instance at renkulab.io; several other smaller instances
- Primary use-cases:
  - Teaching (courses, workshops)
  - Small teams working on data analysis projects
  - Showcasing of derived datasets and results
  - Improving reusability of data products
- Used as a “connecting” piece (e.g. enabling collaborative access to data products from MMODA)
What’s ahead

- Renku as the “middle layer”/connector of code, compute and data
- More comprehensive overviews of where and how data is used
- Better integration with data providers and institutional repositories
- Our goal is to make data “alive” – how can we do better? What would you imagine to be useful for your community?
- High-level organizational, group, and topical views based on the knowledge graph
We want to hear from you!

👋 Try out Renku
- renkulab.io - Public

📝 Renku Docs

❓ Run into a problem?
- Post on Discourse (our forum)
- Submit a bug report

💡 Feature Request?
- Discourse!