Icep Documentation

Release 1.0.1

Lukas Heumos

Apr 08, 2021

CONTENTS:

1 lcep **1**
- 1.1 Features . . . 1
- 1.2 Credits . . . 1

2 Usage **3**
- 2.1 Setup . . . 3
- 2.2 Training . . . 3

3 Model **5**
- 3.1 Overview . . . 5
- 3.2 Training and test data . . . 5
- 3.3 Model details . . . 5
- 3.4 Evaluation . . . 5
- 3.5 Hyperparameter selection . . . 5

4 Credits **7**
- 4.1 Development Lead . . . 7
- 4.2 Contributors . . . 7

5 Changelog **9**
- 5.1 1.0.1 (2021-04-08) . . . 9
- 5.2 1.0.0 (2021-03-11) . . . 9
- 5.3 0.1.0 (2021-03-11) . . . 9

6 Contributor Covenant Code of Conduct **11**
- 6.1 Our Pledge . . . 11
- 6.2 Our Standards . . . 11
- 6.3 Our Responsibilities . . . 11
- 6.4 Scope . . . 12
- 6.5 Enforcement . . . 12
- 6.6 Attribution . . . 12

7 Indices and tables **13**

CHAPTER

ONE

LCEP

Classifying cancerous liver samples from gene expression data.

- Free software: MIT
- Documentation: https://lcep.readthedocs.io.

1.1 Features

- Fully deterministic machine learning model based on XGBoost and MLflow using the mlf-core framework
- Classify cancerous and healthy tissue samples from gene expression data

1.2 Credits

This package was created with mlf-core using cookiecutter.

CHAPTER

TWO

USAGE

2.1 Setup

mlf-core based mlflow projects require either Conda or Docker to be installed. The usage of Docker is highly preferred, since it ensures that system-intelligence can fetch all required and accessible hardware. This cannot be guaranteed for Mac let alone Windows environments.

2.1.1 Conda

There is no further setup required besides having Conda installed and CUDA configured for GPU support. mlflow will create a new environment for every run.

2.1.2 Docker

If you use Docker you should not need to build the Docker container manually, since it should be available on Github Packages or another registry. However, if you want to build it manually for e.g. development purposes, ensure that the names matches the defined name in the ``MLproject``file. This is sufficient to train on the CPU. If you want to train using the GPU you need to have the NVIDIA Container Toolkit installed.

2.2 Training

2.2.1 Training on the CPU

Set your desired environment in the MLproject file. Start training using `mlflow run .`. You need to disable CUDA to train on the CPU! See parameters.

2.2.2 Training using GPUs

Conda environments will automatically use the GPU if available. Docker requires the accessible GPUs to be passed as runtime parameters. To train using all gpus run `mlflow run . -A gpus=all`. You can replace `all` with specific GPU ids (e.g. 0) if desired.

2.2.3 Parameters

- `training-data` Path to the training data csv file `['train.csv': string]`
- `test-data` Path to the test data csv file `['test.csv': string]`
- `cuda` Whether to train with CUDA support (=GPU) `['True': string]`
- `max_epochs` Number of epochs to train `[1000: int]`
- `general-seed` Python, Random, Numpy seed `[0: int]`
- `xgboost-seed` XGBoost specific seed `[0: int]`
- `single-precision-histogram` Whether to enable single precision for histogram building `['True': string]`

CHAPTER

THREE

MODEL

3.1 Overview

The hereby trained model classifies samples generated from gene expression data into cancerous or healthy.

3.2 Training and test data

Patients with cancer and without cancer were sequenced (RNA-Seq). All samples of the patients were assigned 'cancerous' or 'healthy'. The RNA-Seq experiments generated reads, which are commonly associated with expression values per gene. These expression values were normalized into transcripts per million (TPM) values. Next, all genes were subject to pathway analysis. Any genes not present in any cancer associated pathway were discarded. Finally, the whole dataset was split into 75% training and 25% test data. lcep was trained with the training data and evaluated using the test data.

3.3 Model details

The model is based on XGBoost. Training was conducted using a single GPU . Hence, `gpu_hist` is the training algorithm of choice.

3.4 Evaluation

The model was evaluated on 20% of unseen test data. The reported root mean squared error origins from the test data. The full training history is viewable by running the mlflow user interface inside the root directory of this project: `mlflow ui`.

3.5 Hyperparameter selection

The hyperparameters of this model were selected using a grid search approach.

1. `single-precision-histogram` was enabled for faster training
2. `subsample` was set to 0.7
3. `colsample_bytree` was set to 0.6
4. `learning_rate` was set to 0.2

5. `max_depth` was set to 3
6. `min_child_weight` was set to 1
7. `eval_metric` was set to `logloss`
8. `objective` was set to `binary:logistic`

CHAPTER

FOUR

CREDITS

4.1 Development Lead

- Lukas Heumos <lukas.heumos@posteo.net>
- Steffen Lemke <steffen.lemke@uni-tuebingen.de>

4.2 Contributors

None yet. Why not be the first?

CHAPTER

FIVE

CHANGELOG

This project adheres to Semantic Versioning.

5.1 1.0.1 (2021-04-08)

Added

Fixed

- Fixed train/test dataset float point variation of some samples. The new dataset was generated by the nextflow-lcep pipeline.

Dependencies

Deprecated

5.2 1.0.0 (2021-03-11)

Added

- Added new train and test dataset based on TCGA-LIHC & GTEx (liver)
- Added optimized hyperparameters to the model

Fixed

Dependencies

Deprecated

5.3 0.1.0 (2021-03-11)

Added

- Created the project using mlf-core

Fixed

Dependencies

Deprecated

CHAPTER

SIX

CONTRIBUTOR COVENANT CODE OF CONDUCT

6.1 Our Pledge

In the interest of fostering an open and welcoming environment, we as contributors and maintainers pledge to making participation in our project and our community a harassment-free experience for everyone, regardless of age, body size, disability, ethnicity, gender identity and expression, level of experience, nationality, personal appearance, race, religion, or sexual identity and orientation.

6.2 Our Standards

Examples of behavior that contributes to creating a positive environment include:

- Using welcoming and inclusive language
- Being respectful of differing viewpoints and experiences
- Gracefully accepting constructive criticism
- Focusing on what is best for the community
- Showing empathy towards other community members

Examples of unacceptable behavior by participants include:

- The use of sexualized language or imagery and unwelcome sexual attention or advances
- Trolling, insulting/derogatory comments, and personal or political attacks
- Public or private harassment
- Publishing others' private information, such as a physical or electronic address, without explicit permission
- Other conduct which could reasonably be considered inappropriate in a professional setting

6.3 Our Responsibilities

Project maintainers are responsible for clarifying the standards of acceptable behavior and are expected to take appropriate and fair corrective action in response to any instances of unacceptable behavior.

Project maintainers have the right and responsibility to remove, edit, or reject comments, commits, code, wiki edits, issues, and other contributions that are not aligned to this Code of Conduct, or to ban temporarily or permanently any contributor for other behaviors that they deem inappropriate, threatening, offensive, or harmful.

6.4 Scope

This Code of Conduct applies both within project spaces and in public spaces when an individual is representing the project or its community. Examples of representing a project or community include using an official project e-mail address, posting via an official social media account, or acting as an appointed representative at an online or offline event. Representation of a project may be further defined and clarified by project maintainers.

6.5 Enforcement

Instances of abusive, harassing, or otherwise unacceptable behavior may be reported by opening an issue. The project team will review and investigate all complaints, and will respond in a way that it deems appropriate to the circumstances. The project team is obligated to maintain confidentiality with regard to the reporter of an incident. Further details of specific enforcement policies may be posted separately.

Project maintainers who do not follow or enforce the Code of Conduct in good faith may face temporary or permanent repercussions as determined by other members of the project's leadership.

6.6 Attribution

This Code of Conduct is adapted from the Contributor Covenant, version 1.4, available at https://www.contributor-covenant.org/version/1/4/code-of-conduct.html

CHAPTER

SEVEN

INDICES AND TABLES

- genindex
- modindex
- search