Readme

Authors: Lia Bifano, Filippa Bång, Roman Bachmann

Architecture

- **./data/**: Folder where training and testing data should be placed.
- **config.py**: Contains environment variables, like the path to the data.
- **data.py**: Contains functions for loading and storing CSV files.
- **feature_eng.py**: Contains functions that can be used to enhance the feature set of the input data.
- **helpers.py**: Contains helper functions that are called by the 6 functions in the implementations.py file.
- **implementations.py**: Contains the 6 regression functions needed for Project 1.
- **performance.py**: Contains functions to evaluate the performance and quality of the training.
- **run.py**: Code that trains a polynomial regression and will produce the same results for the given test.csv data as our best score on Kaggle.

How to run

- Download the train.csv file from the epfml-higgs Kaggle competition and place it inside the ./data/ folder.
- Place the test.csv file in the ./data/ folder.
- Make sure you are in an environment running python 3.5 and that numpy is installed
- Execute the following command in the directory where run.py is:

```
python run.py
```

Description of run.py

Running run.py, the user can replicate out team's best result. Run.py achieves this in the following way:

- Load the training data from train.csv into numpy arrays
- Replace the missing values (-999) by the column mean
- Create the full polynomial base expansion with 5 degrees, using the 11 columns that we found to produce the best results
- Evaluate how well the training is doing by printing the number of correctly classified predictions in the training data.
- Load the test.csv data and apply the same feature engineering pipeline to it

- Predict the test labels and save them in a csv file
- After running run.py, the submission csv file can be found as submission.csv

The training takes approximately 20 minutes on a Macbook Pro. To get a sense of the progress, the programme prints out information about the status of execution.