Algorithm Validation & Deliverables, Project RL

Validation

To measure each algorithm objectively, and to be able to create a small competition between the different Teams, we have created an official environment for the Electric Car which is intended to be used by everyone.

The (Gym) environment for the Electric Car can be found in **TestEnv.py**. An additional file called **main.py** is added to show you how to use **TestEnv.py**.

Notes about the Environment

The environment `Electric_Car` can be simulated using the Gym notation. The action it expects is a value between -1 and 1, where a value of 1 will buy the maximum amount of electricity ($25/0.9kWh$) from the grid at double the market price, and a value of -1 will sell the maximum amount of electricity ($25kWh$) from the battery at market price.

An observation in `Electric_Car` will give you the 7-dimensional tuple: [**battery level, electricity price, hour of day, day of week, day of year, month, year**]. From this observation you can create your own set of states, and decide yourself how much of these states will go to your Q agent.

PS: Unless absolutely necessary, do not change the file **TestEnv.py** !!

Final Code Deliverables

Next to your final report, we expect 1 zipped folder with the following contents before **Thursday February 1st 23:59**:

main.py

This is your own main file, that runs your agent on the official `Electric_Car` environment. We should be able to test your algorithm by running:

```
python3 main.py --excel_file validate.xlsx
```

Where validate.xlsx is the path to the validation file (2 years of price data).

TestEnv.py

This is the official environment, with maybe some slight modifications if absolutely necessary for your Agent. If ill-intentioned modifications are applied, such as trying to create 'fake' profits, penalty points will be subtracted from the grade.

agent.py

This represents your Agent and all its functions. In the file **main.py**, you import your agent and train / validate it on the `Electric_Car` environment.

utils.py (Optional)

This represents any other external or self-coded functions you need for your code to run smoothly.

requirements.txt

This is a document with all the packages needed in the virtual environment, so that we can install your virtual environment on our own machine using the command:

```
pip install -r requirements.txt
```

The final zip file we need from every group, with **X** representing your group number, should thus look like this:

```
GroupX.zip
├── main.py
├── TestEnv.py
├── agent.py
├── utils.py
└── requirements.txt
```

Before sending, make sure that your code runs in the structure listed above. If everything works, send the zip file to **j.e.kooi@vu.nl**.

Good luck!