# Final Project for Exam

Marius Lindauer




Winter Term 2021/22

# Reminder: Final Grading

- Implement a larger project (worth $1-2$ weeks full time)
  - You can propose your own project idea!
    - Hand-in a short summary of the idea and we will provide feedback regarding feasibility
  - Teamwork (at most 3) again possible
    - Larger team $\rightarrow$ larger scope of the project
- "Exam"
  - First $\sim 15$ min: Present your project idea and results
    - Of course, everyone will present the project on their own
  - Second 15min: We will ask further questions about your project and how it relates to stuff you learned in the lecture.
- You will have the choice between a virtual and on-site exam.

# Requirements

- 2 different options for the scope of the project
  - see next slides

# Requirements

- 2 different options for the scope of the project
  - see next slides
- Proposal should include:
  - Scope and main objective(s)
  - Which open-source frameworks you plan to use
    - both for RL agents and for benchmark envs
    - some presented in the next (last) exercise
  - Amount for compute resources you will use
  - Rough time frame / milestones
  - ⇝ At most 1 page

# Requirements

- 2 different options for the scope of the project
  - see next slides
- Proposal should include:
  - Scope and main objective(s)
  - Which open-source frameworks you plan to use
    - both for RL agents and for benchmark envs
    - some presented in the next (last) exercise
  - Amount for compute resources you will use
  - Rough time frame / milestones
  - ⇝ At most 1 page
- Sound scientific workflow
  - Fill out reproducibility check list (see studip)

# Requirements

- 2 different options for the scope of the project
  - see next slides
- Proposal should include:
  - Scope and main objective(s)
  - Which open-source frameworks you plan to use
    - both for RL agents and for benchmark envs
    - some presented in the next (last) exercise
  - Amount for compute resources you will use
  - Rough time frame / milestones
  - ⇝ At most 1 page
- Sound scientific workflow
  - Fill out reproducibility check list (see studip)
- Hand in: source code, ML reproducibility check list and PDF presentation with at most 7 slides (excl. title slide)
  - Proper code documentation; PEP8
  - Send us an invitation to a GitHub Repo
  - To both: eimer@tnt.uni-hannover.de, schubert@tnt.uni-hannover.de

# Option I: New Env

- Propose a new & interesting benchmark environment / application for RL (incl. state, action, reward, transition, ...)
  - Use OpenAI-Gym env API
  - Should be somehow related to preventing climate change
- Minimal requirement: An RL agent (of your choice) can learn something reasonable
  - That is, it performs better than a static or random policy
  - You can use already implemented RL algorithms
  - We will provide a list of recommended RL agents
- Further goals could include:
  - Impact of state or action space (e.g., size or encoding)
  - Reward signal or reward shaping
  - Does the Markov assumption hold true?

$\rightsquigarrow$ Environment should not be too trivial (e.g., a small maze)

# Option II: RL Agent

- Implement an RL agent from scratch
  - Pick a (recent) RL paper and re-implement it
  - Don't use existing implementations!
- Minimal requirement: Your RL agent can learn a reasonable policy on a RL benchmark of your choice
  - That is, it performs better than a static or random policy
  - You can use already implemented RL benchmark envs
  - We will provide a list of recommended RL envs in the next exercise session
- Further goals could include:
  - Variants of different algorithm components (e.g., experience replay)
  - Hyperparameter sensitivity study
  - Comparison against other baselines

# Score Distribution[1]

- Idea, Topic & Results: at most 25
  (sufficiently challenging, interesting beyond lecture, gained insights, results, ...)
- Implementation: at most 25
  (correctness, reproducibility, code documentation, code quality, ...)
- Presentation: at most 25
  (concise slides, clear message, structure, citations, plots, ...)
- Answering of questions: at most 25
  (correct, concise, knowledge, own ideas, ...)

[1] Not the same scoring as in iML

# Score Distribution[1]

- Idea, Topic & Results: at most 25
  (sufficiently challenging, interesting beyond lecture, gained insights, results, ...)
- Implementation: at most 25
  (correctness, reproducibility, code documentation, code quality, ...)
- Presentation: at most 25
  (concise slides, clear message, structure, citations, plots, ...)
- Answering of questions: at most 25
  (correct, concise, knowledge, own ideas, ...)
- 100-90: 1.0; 89 - 85: 1.3; 84 - 80: 1.7
- 79 - 75: 2.0; 74 - 70: 2.3; 69 - 65: 2.7
- 64 - 60: 3.0; 59 - 55: 3.3; 54 - 50: 3.7
- 49 - 45: 4.0; $<$ 45 : failed

[1] Not the same scoring as in iML

# Deadlines

- Proposal Deadline: Jan 27th (AoE[2])
  - Submit earlier to get feedback sooner!
- Feedback for proposal: latest Feb 3rd
- Results Deadline: March 4th (AoE)
- Exam: March 14th - March 18th (in the mornings)
  - We will send you a link for registration to one of the possible slots

[2] Anywhere on Earth

Questions?