Федеральное государственное автономное образовательное учреждение высшего образования
"Национальный исследовательский университет
"Высшая школа экономики"

Факультет компьютерных наук
Научно-учебная лаборатория методов анализа больших данных

Рабочая программа дисциплины
Обучение с подкреплением

для образовательной программы «Прикладная математика и информатика»
направления подготовки 01.03.02 Прикладная математика и информатика
уровень - бакалавр

Разработчик(и) программы
Ратников Фёдор Дмитриевич, к. ф.-м. н., fratnikov@hse.ru

Одобрена на заседании научно-учебной лаборатории методов анализа больших данных
«___»_____ 2017 г.

Зав. лабораторией
Устюжанин А.Е. _____

Утверждена Академическим советом образовательной программы
«___»_____ 2017 г., № протокола_____

Академический руководитель образовательной программы Конушин А. С. _____

Москва, 2017

1. Область применения и нормативные ссылки

Настоящая программа учебной дисциплины устанавливает минимальные требования к знаниям и умениям студента и определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей, ведущих дисциплину «Обучение с подкреплением», учебных ассистентов и студентов направления подготовки 01.03.02 «Прикладная математика и информатика», обучающихся по образовательной программе «Прикладная математика и информатика»

Программа учебной дисциплины разработана в соответствии с:

- Образовательным стандартом ФГАОУ ВО НИУ ВШЭ по направлению подготовки 01.03.02 Прикладная математика и информатика https://www.hse.ru/data/2017/09/04/1321436546/2.03.02%20%D0%9F%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%B0%D1%8F%20%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0%20%D0%B8%20%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0.pdf ;
- Образовательной программой «Прикладная математика и информатика», направление подготовки 01.03.02 «Прикладная математика и информатика»
- Объединенным учебным планом университета по образовательной программе «Прикладная математика и информатика», утверждённым в 2017 г.

2. Цели освоения дисциплины

Основная цель освоения дисциплины «Обучение с подкреплением» - научить студентов использовать методы одноимённой области машинного обучения в практических и исследовательских задачах.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины студент осваивает компетенции:

Компетенция	Код по ОС ВШЭ	Уровень формирования компетенции	Дескрипторы – основные признаки освоения (показатели достижения результата)	Форма контроля уровня сформированности компетенции
Способен выявлять научную сущность проблем в профессиональной области.	УК-2	РБ	Понимает отличия в постановке задачи обучения с подкреплением и обучения с учителем.	Домашняя работа
Способен решать проблемы в профессиональной деятельности на основе анализа и синтеза	УК-3	РБ	Понимает, в каких случаях для решения задачи можно применить обучение с подкреплением (далее RL)	Домашняя работа
Способен описывать проблемы и ситуации профессиональной деятельности, используя язык и аппарат математики	ПК- 1	РБ, СД	Понимает и может реализовать value-based RL алгоритмы (VI, PI, Q-learning, SARSA) для решения задач с конечным количеством состояний	Домашняя работа

Компетенция	Код по ОС ВШЭ	Уровень формирования компетенции	Дескрипторы – основные признаки освоения (показатели достижения результата)	Форма контроля уровня сформированности компетенции
Способен понимать, совершенствовать и применять современный математический аппарат	ПК-3	СД	Понимает отличия между value-based и policy-based алгоритмами RL, знает как использовать policy gradient методы для нахождения оптимальной стратегии в RL задачах.	Домашняя работа
Способен провести сбор, обработку и анализ данных с использованием существующих методов машинного обучения	ПК-7	СД	Понимает и может реализовать алгоритмы RL для стратегии, аппроксимированной с помощью моделей машинного обучения (линейные, нейрончие и т.п.)	Домашняя работа
Способен разработать математическую модель и провести её анализ для поставленной теоретической или прикладной задачи	ПК-8	РБ, СД		Домашняя работа
Способен разработать и реализовать в виде программного модуля алгоритм решения поставленной теоретической или прикладной задачи на основе математической модели	ПК-9	СД	Понимает и может реализовать обучение на базе RL для задач seq2seq.	Домашняя работа

Виды и задачи профессиональной деятельности

Научно-исследовательские	
Исследование и разработка математических моделей и методов, алгоритмов и программного обеспечения по тематике проводимых научно-исследовательских проектов;	НИД-3
Применение математических методов и наукоемких технологий для изучения и моделирования сложных систем, в частности, в области обработки и анализа данных, экономики, социологии, физики, наук о жизни и др.;	НИД-4
Проектные и производственно-технологические	
Разработка и исследование алгоритмов, вычислительных моделей и моделей данных для реализации элементов новых (или известных) систем информационных технологий	ПД-2
Разработка архитектуры, алгоритмических и программных решений системного и прикладного программного обеспечения	ПД-3
Разработка программного и информационного обеспечения компьютерных систем, автоматизированных систем вычислительных комплексов, сервисов, операционных систем и распределенных баз данных	ПД-4
Изучение и использование различных языков программирования, алгоритмов, библиотек и пакетов программ при разработке программного обеспечения	ПД-5

4. Место дисциплины в структуре образовательной программы

Настоящая дисциплина относится к профессиональному циклу, блоку дисциплин по выбору.

Изучение данной дисциплины базируется на следующих дисциплинах:

- Машинное Обучение 1
- Глубинное Обучение
- Непрерывная Оптимизация
- Основы Теории Игр
- Байесовские методы в машинном обучении
- Параллельные и распределённые вычисления

Основные положения дисциплины могут быть использованы в дальнейшем при изучении дисциплин:

- Машинное обучение 2
- Методы искусственного интеллекта в робототехнических системах

5. Тематический план учебной дисциплины

№	Название раздела	Всего часов	Аудиторные часы		Самостоятельная работа
			Лекции	Практические занятия	
1	Introduction, RL through stochastic optimization	12	4	4	12
2	Value-based RL methods	18	6	6	20
3	Approximate & deep RL	18	6	6	24
4	Policy-based RL	12	4	4	24
5	Practical applications	12	4	4	24
	Итого	152	22	24	104

6. Формы контроля знаний студентов

Тип контроля	Форма контроля	3/4 год				Параметры **
		1	2	3	4	
Текущий	Домашнее задание			6	6	Задания «на закодить» в jupyter-notebook.
	Проект				1	Решение задачи на approx. RL на выбор студента

7. Критерии оценки знаний, навыков

Оценка за курс выставляется по баллам. Сумма баллов за все задания превышает порог максимальной оценки.

По приблизительным подсчётам, оценку 10 (отл) можно получить за выполнение 60-80% заданий, 7(Хорошо) — 40-50% и так далее. В заданиях кроме обязательной части также присутствует дополнительная часть, за которую тоже начисляются баллы. Таким образом, студент может выбрать между углублённым изучением одной области или общими знаниями в нескольких за одну и ту же оценку.

Наконец, дополнительные баллы студент может получить за любую научную или инженерную активность, связанную с обучением с подкреплением: разбор научных статей на тематическом семинаре, воспроизведение работ с последующей публикацией кода, значимый вклад

в open-source проекты, научные статьи и т.п. В итоге, если студенту уже знакомы некоторые темы из курса, он может избежать базовых заданий по этим темам в пользу углублённого изучения других тем.

Всё числа, указанные в предыдущем параграфе, являются предварительными: окончательные критерии оценивания будут опубликованы на первом занятии.

8. Содержание дисциплины

Порядок занятий дан приблизительный, некоторые лекции могут поменяться местами. Примеры заданий можно посмотреть тут - http://bit.ly/2CVsWlx в папке соответствующей недели.

Section 1: Introduction, RL through stochastic optimization

- **Class 1** Intro to Reinforcement Learning
    - Lecture: RL problems around us. Decision processes. Basic genetic algorithms
    - Seminar: Intro to openai gym, RL with supervised learning, basic black box optimization.
- **Class 2** RL as blackbox optimization
    - Lecture: Recap on genetic algorithms; Evolutionary strategies. Stochastic optimization, Crossentropy method. Parameter space search vs action space search.
    - Seminar: Tabular CEM for Taxi-v0, deep CEM for box2d environments.

Section 2: Value-based RL methods

- **Class 1** Value-based methods
    - Lecture: Discounted reward MDP. Value-based approach. Value iteration. Policy iteration. Discounted reward fails.
    - Seminar: Value iteration. Policy iteration.
- **Class 2** Model-free reinforcement learning
    - Lecture: Q-learning. SARSA. Off-policy Vs on-policy algorithms. N-step algorithms. TD(Lambda).
    - Seminar: Qlearning Vs SARSA Vs Expected Value SARSA , experience replay.
- **Class 3** deep learning recap
    - Lecture: Deep learning 101
    - Seminar: Simple image classification with convnets

Section 3: Approximate & deep RL

- **Class 1** Approximate reinforcement learning
    - Lecture: Infinite/continuous state space. Value function approximation. Convergence conditions. Multiple agents trick; experience replay, target networks, double/dueling/bootstrap DQN, etc.
    - Seminar: Approximate Q-learning with experience replay. (CartPole, Atari)

- **Class 2** Intro to recurrent neural nets
  - Lecture: Recurrent neural networks, backprop through time, LSTM
  - Seminar: character-level language models with RNN
- **Class 3** Partially observable MDPs
  - Lecture: POMDP intro. POMDP learning (agents with memory). POMDP planning (POMCP, etc)
  - Seminar: Deep kung-fu & doom with recurrent A3C and DRQN

Section 4: policy-based reinforcement learning

- **Class 1** Policy gradient methods I
  - Lecture: Motivation for policy-based, policy gradient, logderivative trick, REINFORCE, variance reduction with baselines, advantage actor-critic (incl. GAE)
  - Seminar: REINFORCE, advantage actor-critic
- **Class 2** Policy gradient methods II
  - Lecture: Trust region policy optimization. NPO/PPO. Deterministic policy gradient. DDPG. Bonus: DPG for discrete action spaces.
  - Seminar: Approximate TRPO for simple robotic tasks.

Section 5: Practical applications

- **Class 1** Exploration in reinforcement learning, appliations I
  - Lecture: Contextual bandits. Thompson Sampling, UCB, bayesian UCB. Exploration in model-based RL, MCTS. "Deep" heuristics for exploration.
    Seminar: bayesian exploration for contextual bandits. UCB for MCTS.
- **Class 2** Applications II
  - Lecture: Reinforcement Learning as a general way to optimize non-differentiable loss. G2P, machine translation, conversation models, image captioning, discrete GANs. Self-critical sequence training.
  - Seminar: Simple neural machine translation with self-critical sequence training

1 Образовательные технологии

Практические задания выполняются студентами в формате jupyter notebook. В курсе используется стандартный набор библиотек машинного обучения (numpy, scipy, sklearn, pandas, matplotlib), вычислительные графы в pytorch / tensorflow / theano (поддерживаются все 3 фреймворка). Стандартные среды для отладки алгоритмов взяты из openai gym / openai universe. Проверка домашних заданий осуществляется в системе anytask.

1.1 Методические рекомендации преподавателю

Не требуются

1.2 Методические указания студентам

Студентам, заранее не знакомым со стэком библиотек python for data science (numpy, scipy, sklearn, matplotlib, pandas) рекомендуется их изучить до начала курса. Короткий ликбез по ним лежит тут: http://bit.ly/2ATt4jd .

Рекомендованная траектория прохождения курса — выбор интересующих тем, сдача заданий по ним в течение семестра и как следствие — относительно небольшая нагрузка в предсессионный период. Этому также способствует то, что многие задания в первой половине курса имеют более выгодное соотношение усилий за 1 балл.

Студентам, уже знакомым с азами обучения с подкреплением, рекомендуется обсудить с преподавателями темы для проектов и заниматься ими всчёт не интересующих студента заданий. Проектом может быть что угодно по теме курса: (исследования, выступления на научных семинарах, воспроизведение статей, научно-популярные статьи, open-source разработка, и т. п.). Единственное требование — тема проекта и разбалловка должна быть согласована с преподавателем как можно раньше.

2 Оценочные средства для текущего контроля и аттестации студента

2.1 Оценочные средства для оценки качества освоения дисциплины в ходе текущего контроля

Текущий контроль требует от студентов решать задания, в основном оформленные в виде jupyter-notebooks (примеры: http://bit.ly/2B9dpkT , http://bit.ly/2BtrTJg , http://bit.ly/2kGMlyk).

2.2 Примеры заданий промежуточной аттестации

Все задания имеют практический характер. Пример тематики заданий: “Реализовать Value Iteration и Policy Iteration для данной среды”, “Создать такую среду в которой Value Iteration сходится максимально долго” “Дообучить модель машинного перевода на максимизацию BLEU с помощью policy gradient” и т.п.

3 Порядок формирования оценок по дисциплине

Информация про формирование оценок содержится здесь - **http://bit.ly/2CVsWlx** .

Оценка за курс ставится исключительно на основе баллов, накопленных за семинарские/домашние задания и, опционально, проекты. Экзамен по данной дисциплине не предусмотрен.

Полностью верно сделанный семинар оценивается в 10 баллов; более сложные задания на несколько недель имеют вес в 30-40 баллов в зависимости от сложности. Оценки за дополнительные задания указаны в самих заданиях и также зависят от сложности.

Формула оценивания, официально требуемая в программе учебной дисциплины, выглядит следующим образом:

Итоговая оценка получается делением накопленной оценки на 20 с арифметическим округлением. Формула выставления оценки может несколько отличаться в каждой итерации курса.

4 Учебно-методическое и информационное обеспечение дисциплины

4.1 Базовый учебник

Reinforcement Learning: An Introduction (by R.S. Sutton). http://bit.ly/2v0NDbO .

4.2 Основная литература

Дополнительные материалы по каждой теме можно найти в репозитории (http://bit.ly/2kBEy3I) папках ./week*/ в разделе Materials. Например, http://bit.ly/2BvV5PJ .

4.3 Дополнительная литература

Дополнительные материалы по каждой теме можно найти в репозитории (http://bit.ly/2kBEy3I) папках ./week*/ в разделе More Materials. Например, http://bit.ly/2oDrotr .

4.4 Справочники, словари, энциклопедии

Справочников, словарей и энциклопедий по предмету не предусмотрено.

4.5 Программные средства

Для успешного освоения дисциплины, студент использует следующие программные средства:

Data Science Stack: python 2/3, numpy, scipy, sklearn, matplotlib, pandas, jupyter-notebook
Deep Learning Stack: Theano + Lasagne ИЛИ Tensorflow ИЛИ Pytorch на выбор студентам
Reinforcement Learning environments: OpenAI Gym

Студент может выбрать для выполнения заданий другой фреймворк глубинного обучения с согласия преподавателей. Проходить курс на другом языке программирования не рекомендуется и допускается только в исключительных случаях.

4.6 Дистанционная поддержка дисциплины

Студент может найти записи лекций, слайды, семинарские и домашние задания и форму сдачи этих заданий в репозитории (http://bit.ly/2kBEy3I). Также в Readme репозитория можно найти ссылку на чат курса (обычно это telegram), в котором можно задавать вопросы по курсу.

5 Материально-техническое обеспечение дисциплины

Чтобы сдать курс студенту потребуется иметь что-либо из

- **Минимальный вариант:** вем-браузер с поддержкой javascript (для Jupyter), и экстраординарное упорство.
- **Комфортный вариант:** ПК (desktop/laptop/server) с 64-битной ОС, 4+ Gb RAM и 2+ физических ядра CPU
- **Максимальный вариант:** доступ к машине с >= 8 Gb RAM и high-end процессором или дискретной видеокартой с поддержкой CUDA computa capaility >=3.0 и >2GB памяти GPU.