Course Syllabus

Course Title (in English)  Foundations of Data Science
Course Title (in Russian)  Основы Наук о Данных
Lead Instructor(s)  Burnaev, Evgeny

Is this syllabus complete, or do you plan to edit it again before sending it to the Education Office?
The syllabus is a final draft waiting for approval (once approved the syllabus will be published on the public web-site and other systems)

Contact Person  Evgeny Burnaev
Contact Person's E-mail  E.Burnaev@skoltech.ru

1. Annotation

Course Description
In this course, we introduce the forefront of modern research in data science and familiarize Ph.D. students with state of the art in those areas. In particular, we introduce cornerstone subjects that are not commonly discussed in undergraduate or graduate Machine Learning classes. This course is intended to serve as an introduction to the basics of everyday industrial software engineering. Also, this course explores extensively four novel areas in Machine Learning, namely Causality, Sequential Data, Geometric Computer Vision and Reinforcement Learning.

Over multiple weeks, we will investigate how these methods and algorithms can be used for analyzing scientific data, social networks or time-series data, mining sequences, carrying out text/web analysis, topic modeling, and pattern mining. We explore how these concepts are applied for dimensionality reduction and manifold learning, combinatorial optimization, relational and structured learning, classification and regression methods, semi-supervised learning, unsupervised learning including anomaly detection and clustering, kernel methods, compressed sensing and sparse modeling, Bayesian methods, deep learning, hyper-parameter, and model selection.

The course aims to bring all students on the same page regarding the nature and orientation of state-of-the-art work in their field so that they acquire both depth and breadth of knowledge.

Course Prerequisites / Recommendations
The prerequisites for the course are the same as prerequisites for admission to the PhD program in Information Science and Technology.
## 2. Structure and Content

<table>
<thead>
<tr>
<th>Course Academic Level</th>
<th>PhD-level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of ECTS credits</td>
<td>6</td>
</tr>
<tr>
<td>Topic</td>
<td>Summary of Topic</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Software Engineering for DS               | - Unix fundamentals (shell and command line, scripting, filesystem, streams and pipes, parameter parsing, remote machine and ssh, etc.).  
- Software engineering in teams (code review and version control, reproducibility and containers, testing and test-driven development, learning great codestyle, software deployment and APIs).  
- Software design (team organization, software specifications, software project management, and software design methodologies).                                                                 | 9                     | 4.5                   | 4.5               |
| Causality                                 | - What is Causal Inference?  
- Association vs Causation  
- Learning From Data With Statistical Models  
- Estimator Reliability And Regression Models  
- Introduction To Counterfactuals, Randomization Based Inference, And Missing Data Problems  
- Dealing With (Simple) Confounding  
- Instrumental Variables  
- Mediation Analysis  
- Statistical Models Of A Directed Acyclic Graph  
- Undirected Models  
- Directed Models  
- Causal Models Of A Directed Acyclic Graph  
- Causal Models With Hidden Variables  
- Causal Decision Theory  
- Learning The Structure Of A Directed Acyclic Graph  
- Structure Learning With Scoring Methods                                                                 | 9                     | 4.5                   | 4.5               |
| Sequential Data                           | - Neural network architectures for time series classification: are CNN good?  
- Neural networks perspective from RNN and other guys: LSTM, GRU  
- Attention mechanism and transformer architecture for sequential data: seq2seq models  
- Adversarial attacks for sequences with seq2seq models  
- Anomaly detection in time series using DL                                                                 | 7.5                   | 3.75                  | 3.75              |
- 3D surface generation and mesh extraction. Poisson surface reconstruction. Marching cubes.                                                                 | 6                     | 3                     | 3                 |
| Reinforcement Learning                    | - Tabular methods: Markov decision processes and value functions, tabular planning and model-free RL.  
- Function approximation and deep RL: semi-gradient evaluation and control, linear function approximation and least-squares temporal-difference, deep RL.  
- Policy gradients and model-based RL: actor-critic, policy gradients and natural gradients, tabular and deep-model-based methods, smart exploration.                                                                 | 4.5                   | 2.25                  | 2.25              |
3. Assignments

<table>
<thead>
<tr>
<th>Assignment Type</th>
<th>Assignment Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test/Quiz</td>
<td>They will be multiple choice quizzes on canvas evaluating the material covered that same day</td>
</tr>
<tr>
<td>Team Project</td>
<td>Group project for 2-3 students. This deliverable must be a solid logical high-quality text describing your project and obtained results. The quality of the submission should be equal to the quality of a publication that can go into a minor venue.</td>
</tr>
</tbody>
</table>

4. Grading

<table>
<thead>
<tr>
<th>Type of Assessment</th>
<th>Grade Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pass/Fail</td>
<td>Activity Type</td>
</tr>
<tr>
<td></td>
<td>Class participation</td>
</tr>
<tr>
<td></td>
<td>Test/Quiz</td>
</tr>
<tr>
<td></td>
<td>Team Project</td>
</tr>
</tbody>
</table>

Grading Scale

Pass: 46

Attendance Requirements: Mandatory with Exceptions

5. Basic Information

<table>
<thead>
<tr>
<th>Course Stream</th>
<th>Science, Technology and Engineering (STE)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course Term (in context of Academic Year)</td>
<td>Term 4</td>
</tr>
<tr>
<td>Course Delivery Frequency</td>
<td>Every year</td>
</tr>
</tbody>
</table>

Students of Which Programs do You Recommend to Consider this Course as an Elective?
6. Textbooks and Internet Resources

<table>
<thead>
<tr>
<th>Recommended Textbooks</th>
<th>ISBN-13 (or ISBN-10)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Papers</th>
<th>DOI or URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jared Heinly, Johannes L. Schönberger, Enrique Dunn, Jan-Michael Frahm. &quot;Reconstructing the World* in Six Days &quot;(As Captured by the Yahoo 100 Million Image Dataset)&quot;, CVPR 2015.</td>
<td></td>
</tr>
</tbody>
</table>

7. Facilities
Software

Python 3.4+
Jupyter
Open source python libraries including machine learning ones: Scikit learn, XGboost, Pandas, Matplotlib, Seaborn

8. Learning Outcomes

Knowledge

Obtain a big picture of applied problems in Data Science domain specially in the fields of Causality, Sequential Data, Geometric Computer Vision and Reinforcement Learning; applications include anomaly detection in complex multicomponent systems, churn prediction, scoring and fraud detection, predictive modeling of engineering systems, topics modeling, etc.

Know main Data Science problem statements, available standard Data Science methods and areas of their applications, functionality and constraints of existing algorithmic software libraries, realizing methods of Data Science;

Know the theoretical basis and mathematical tools needed for the development, investigation and justification of Data Science methods and algorithms;

Skill

Be able to formulate in mathematical terms a real-world problem, identify the corresponding type of Data Science problem, utilize existing research results if any to solve the problem;

Be able to identify novel and practically relevant research problems, motivated either by some application, or by the logic of development of Data Science;

Be able to exploit internal structure of the problem and, if necessary, to take it into account when modifying a Data Science method or developing a new one;

Be able to implement methods of Data Science into efficient programming code for considered production system;

Experience

Ability to read, discuss and present research results, novel problems, their potential solutions in Data Science and applications, as well as provide written materials on these matters;

Understanding of the research process, research ethics and academic integrity;

9. Assessment Criteria

Input or Upload Example(s) of Assignment 1:

Select Assignment 1 Type: Team Project
To successfully pass this Team Project, it is necessary to do the following:
(1) Provide a project report. This deliverable must be a solid logical high-quality text describing your project and obtained results. The quality of the submission should be equal to the quality of a publication that can go into a minor venue.
(2) Submit the PDF presentation of your project. It is the file that will be discussed by you in the video.
(3) Provide a link to cloud storage (Dropbox or similar) containing the video presentation of your project.
The video presentation has to be an 8-10 minute video of one of the following forms:
(a) Student(s) record presentation via a screen recording program (+microphone).
(b) Student(s) record presentation on their mobile phone/camera. In this case, the video frame should include both the presenter and clearly visible presentation slides.
All team members do not need to participate in the presentation. Moreover, we highly recommend not to get together in groups (because of COVID-19 DANGER). Thus, only one student may be involved in the video presentation.
The video should be at least semi-HD (720p). Please be informed that all videos will be uploaded to a public video platform such as YouTube.
(4) Link to a GitHub repository with a fully reproducible code for all experiments.
Students must commit all the code required to reproduce the experiments from their report before the project report submission deadline. Additionally to the code, the repository must contain brief descriptions of the project, a quick overview of some results and instructions on how to run the code. This should be provided as a README.MD file.
Reproducibility means that a reviewer should be able to run all the code provided by following the instructions in the README and with relatively low effort. Unless otherwise stated (i.e., projects with high GPU demand), projects should be able to be run directly on Google Colab.
Here are some examples of the good/ideal repositories:
https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix
https://github.com/locuslab/icnn
We clearly understand that it is not straightforward to produce repositories of this quality, yet we state that this is a good target to orient.

The project grading will be done according to the following rules:
(1) If you fail to submit at least one of the four mentioned bullet points in time, your final grade for all the project activities will be zero. We do not accept late submissions. Plan your workload ahead.
(2) The project presentation represents 10% of the final mark. The grade for the presentation will be assigned based on your presentation skills, quality of the presentation, overall exposition and presented results. The grade for the presentation will be the same for all members of the team.
(3) The final project report equals 40% of the final mark. The preliminary grade will be assigned based on the grading scheme approved in your project proposals. Depending on the quality of your final report (quality, structure, completeness, logic), this grade will be decreased.
Please be aware that even if your grade is 100% (according to the grading scheme), you may finally end with a zero if your experiments are incomplete or the results are non-reproducible or they lack proper explanations and code. The grade for the final report will be the same for all members of the team in most cases.
In some extreme cases, we can modify the grades of each team member proportional to the number of contributions of each team member. Yet you should not expect this to happen and should distribute the workload equally between the team members.
<table>
<thead>
<tr>
<th>Input or Upload Example(s) of Assignment 2:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input or Upload Example(s) of Assignment 3:</td>
</tr>
<tr>
<td>Input or Upload Example(s) of Assignment 4:</td>
</tr>
<tr>
<td>Input or Upload Example(s) of Assignment 5:</td>
</tr>
</tbody>
</table>

10. Additional Notes