Course Syllabus

Course Title (in English) | Machine Learning and Applications
Course Title (in Russian) | Машинное обучение
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 work in progress (draft)
Contact Person | Evgeny Burnaev
Contact Person's E-mail | E.Burnaev@skoltech.ru

1. Annotation

Course Description

The course is a general introduction to machine learning (ML) and its applications. It covers fundamental modern topics in ML, and describes the most important theoretical basis and tools necessary to investigate properties of algorithms and justify their usage. It also provides important aspects of the algorithms’ applications, illustrated using real-world problems. The course starts with an overview of canonical ML applications and problems, learning scenarios, etc. Next, we discuss in depth fundamental ML algorithms for classification, regression, clustering, etc., their properties as well as their practical applications. The last part of the course is devoted to advanced ML topics such as Gaussian processes, neural networks, active learning. Within practical sections, we show how to use the methods above to crack various real-world problems. Home assignments include application of existing algorithms to solve applied industrial problems, development of modifications of ML algorithms, as well as some theoretical exercises. The students are assumed to be familiar with basic concepts in linear algebra, probability and real analysis.

Course Prerequisites / Recommendations

Course prerequisites are Numerical Linear Algebra and Optimization Methods. Nevertheless, the course is self-contained and can be taken by any student even if none of the courses above have been taken before. We suppose an attendee be fluent with linear algebra, probability and real analysis.

2. Structure and Content
<table>
<thead>
<tr>
<th>Topic</th>
<th>Summary of Topic</th>
<th>Lectures (&lt;# of hours&gt;)</th>
<th>Seminars (&lt;# of hours&gt;)</th>
<th>Labs (&lt;# of hours&gt;)</th>
</tr>
</thead>
</table>
| **Introduction Lecture** | - Introduction. Some canonical applications and problems. Definitions and terminology  
- Types of problems: Supervised, Semi-supervised, Empirical Risk Minimization, Cross-validation | | | |
| **Regression, Kernel Trick** | - Regression (problem statement)  
- Linear Regression. Closed form Solution  
Ridge Regression. Closed-form Solution. Direct Dual Solution  
- LASSO. L1 Sparsity. ElasticNet | | | |
| **Classification** | - Binary classification, loss curve, ROC/AUC, precision and recall  
- Learning a classifier. Surrogate Loss  
Empirical risk minimization (ERM), Overfitting.  
- Regularization, Log Loss  
- Two class and Multiclass Logistic Regression  
- k-Nearest Neighbour Classifier. Compactness and continuity hypotheses. Distance functions  
- Classification and regression trees. Ensembles  
- Naive Bayes Classifier | | | |
| **Support Vector Machines** | - Convex Optimization. Lagrangian. KKT Conditions.  
- Classification Task. SVM Optimization Problem. Non-separable case. Support Vectors  
- Non-linear separable case. SVM with Kernels  
- SVR. Quadratic loss case. | | | |
- Design choices for decision tree learning: choice of root; purity, entropy, information gain and gini index; loss function.  
| **Advanced classification. Imbalanced and Multilabel cases** | - Imbalanced classification. Imbalance ratio.  
- Resampling methods: random oversampling (ROS); random undersampling (RUS); synthetic minority oversampling technique (SMOTE).  
- Kernel density estimation (KDE). Multidimensional KDE.  
- Feature Selection. LASSO Feature Selection.  
- Regularization. Bayesian View on Regularization.  
- Mallows' Cp Statistic. AIC. BIC.  
- Sensitivity Analysis. Elementary Effects. Sobol Indices. EASI and CSTA.  |
| Adaboost | - Ensembles of classifiers. Bagging.  
- Stacked generalization.  
- Naive boosting for regression.  
- Gradient Boosting Machines  
- Gradient Boosting Decision Trees.  |
- Reproducing Kernel Hilbert Space.  
- SVM with kernels  
- Representer Theorem.  
- Closure Properties of PDS Kernels.  
- Negative Kernels.  |
- Probabilistic vs. Frequentist view. Evidence. MAP and MLE estimates. MAP estimate as a regularization.  
- Bayesian approach to regression (Curve Fitting). MAP and L2 penalized regression. Predictive distribution.  
- Linear basis function regression. Bayesian View. Predictive distribution for LBFR.  
- Model selection for Bayesian Regression.  
- Gaussian Kernel as Covariance function. Interpretability of Kernel Parameters.  
- GP optimization.  
- GP classification. Sigmoidal likelihood.  
- Non-stationary GP.  |
| Shallow Artificial Neural Networks | - Motivation. Real-life Neurons.  
- Perceptron algorithm and SGD. Convergence.  
- Feed-forward Neural Networks. Activation function.  
| Deep Artificial Neural Networks | - Old School approach to feature engineering. Feature construction magic.  
- Neural network as a computational graph.  
- Examples of practical applications. Causes of Deep NN breakthrough  
- Universal Approximation  
- Training computational graphs via backprop.  
- Transfer learning via fine-tuning.  
- Recurrent nets (Brief).  
- Batchnorm. Weight Norm. Dropout. Data Augmentation |
| Dimensionality Reduction | - Problem Statement. Examples: Faces, Airfoils, MNIST.  
- PCA  
- Multidimensional Scaling (MDS)  
- Replicative Neural Networks (Autoencoders)  
- Graph based on nearest neighbours: ISOMAP, LLE  
- t-Stochastic Neighbour Embedding |
| Anomaly Detection | - Anomaly detection.  
- Nearest neighbours based methods  
- One-class SVM. Kernel choice  
- Other approaches to anomaly detection |
| Clustering | - Hierarchical clustering (agglomerative/divisive models).  
- K-means  
- Cluster validity. External Measures: Entropy, Mutual Information, Jaccard Index, Rand Index, Silhouette Coefficient  
Mixture models, etc.  
- Hard and soft assignment with K-means, Gaussian Mixture Models (GMM)  
- Expectation-Maximisation algorithm. EM convergence. K-means in comparison to Learning GMM |
- Sampling criteria: maximum variance criterion, minimization of mean prediction error, integrated mean squared error gain, combination of ImseGain and MaxVar, MaxMin, etc.  
- Active Learning for Classification.  
- Active Heuristic Learning.  
- Types of active learning: stream-based and pool-based. Biased sampling.  
<p>| 3. Assignments |  |</p>
<table>
<thead>
<tr>
<th>Assignment Type</th>
<th>Assignment Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework</td>
<td>HW 1 includes theoretical and practical tasks on topics of the first 6 lectures.</td>
</tr>
<tr>
<td>Homework</td>
<td>HW 2 includes theoretical and practical tasks on topics of the second 6 lectures.</td>
</tr>
<tr>
<td>Homework</td>
<td>HW 3 includes theoretical and practical tasks on topics of the second 6 lectures.</td>
</tr>
<tr>
<td>Homework</td>
<td>(Bonus) HW 4 includes additional theoretical and practical tasks.</td>
</tr>
<tr>
<td>Final Project</td>
<td>Group project for 3-5 students.</td>
</tr>
</tbody>
</table>

### 4. Grading

**Type of Assessment**
- Graded

**Grade Structure**

<table>
<thead>
<tr>
<th>Activity Type</th>
<th>Activity weight, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework Assignments</td>
<td>30</td>
</tr>
<tr>
<td>Homework Assignments</td>
<td>5</td>
</tr>
<tr>
<td>Final Exam</td>
<td>20</td>
</tr>
<tr>
<td>Midterm Exam</td>
<td>15</td>
</tr>
<tr>
<td>Projects</td>
<td>35</td>
</tr>
</tbody>
</table>

**Grading Scale**

<table>
<thead>
<tr>
<th>Grade</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>86</td>
</tr>
<tr>
<td>B</td>
<td>76</td>
</tr>
<tr>
<td>C</td>
<td>66</td>
</tr>
<tr>
<td>D</td>
<td>56</td>
</tr>
<tr>
<td>E</td>
<td>46</td>
</tr>
<tr>
<td>F</td>
<td>0</td>
</tr>
</tbody>
</table>

**Attendance Requirements**
- Mandatory

### 5. Basic Information

**Maximum Number of Students**
Maximum Number of Students

<table>
<thead>
<tr>
<th>Overall:</th>
<th>135</th>
</tr>
</thead>
<tbody>
<tr>
<td>Per Group (for seminars and labs):</td>
<td></td>
</tr>
</tbody>
</table>

Course Stream: Science, Technology and Engineering (STE)

Course Term (in context of Academic Year): Term 3

Course Delivery Frequency: Every year

Course Tags: Math, Programming

6. Textbooks and Internet Resources

<table>
<thead>
<tr>
<th>Required Textbooks</th>
<th>ISBN-13 (or ISBN-10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bishop, C.M. Pattern Recognition and Machine Learning. Springer, 2007</td>
<td>9780387310732</td>
</tr>
<tr>
<td>Barber, D. Bayesian Reasoning and Machine Learning. Cambridge University Press, 2012</td>
<td>9780521518147</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Recommended Textbooks</th>
<th>ISBN-13 (or ISBN-10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shai Shalev-Shwartz, Shai Ben-David. Understanding Machine Learning: From Theory to Algorithms. Cambridge, 2014.</td>
<td>9781107057135</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Web-resources (links)</th>
<th>Description</th>
</tr>
</thead>
</table>
7. Facilities

<table>
<thead>
<tr>
<th>Software</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Colab</td>
</tr>
</tbody>
</table>

8. Learning Outcomes

**Knowledge**

- Obtain a big picture of practical problems exploiting ML methods; applications include anomaly detection in complex multicomponent systems, churn prediction, scoring and fraud detection, predictive modeling of engineering systems, etc.
- Know main ML problem statements;
- Know available standard ML methods and areas of their applications;
- Know functionality and constraints of existing ML algorithmic software libraries (Scikit-learn, TensorFlow, LibSVM, Vowpal Wabbit, etc.);
- Know the theoretical basis and conceptual tools needed for the investigation and justification of algorithms;

**Skill**

- Be able to formulate in mathematical terms a real-world problem, identify the corresponding type of ML problem, select an appropriate ML method;
- Be able to apply existing ML algorithmic software libraries (Scikit-learn, TensorFlow, LibSVM, Vowpal Wabbit, etc.) and interpret obtained results in subject domain terms;
- Be able to implement ML methods into efficient programming code;
- Be able to exploit internal problem/data structure and, if necessary, to take it into account when modifying an ML method or developing a new one;
- Ability to read and discuss research papers in ML and applications;

**Experience**

- Obtain a sufficient experience during practical exercises and project activities to become a qualified user of ML methods.

9. Assessment Criteria

**Input or Upload Example(s) of Assignment 1:**

<table>
<thead>
<tr>
<th>Select Assignment 1 Type</th>
<th>Problem Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Or Upload Example(s) of Assignment 1</td>
<td><a href="https://ucarecdn.com/a3ac299d-03e6-40c0-b070-54ebd213ead1/">https://ucarecdn.com/a3ac299d-03e6-40c0-b070-54ebd213ead1/</a></td>
</tr>
<tr>
<td>Input or Upload Example(s) of Assignment 2:</td>
<td></td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Input or Upload Example(s) of Assignment 3:</td>
<td></td>
</tr>
<tr>
<td>Input or Upload Example(s) of Assignment 4:</td>
<td></td>
</tr>
<tr>
<td>Input or Upload Example(s) of Assignment 5:</td>
<td></td>
</tr>
</tbody>
</table>

10. Additional Notes