ML IN HEALTHCARE
BIOMEDICAL ENGINEERING

Contents

1. Introduction

With billions of mobile devices worldwide and the low cost of connected medical sensors, recording and transmitting medical data has become easier than ever. However, this ‘wealth’ of physiological data has not yet been harnessed to provide actionable clinical information. This is due to the lack of smart algorithms that can exploit the information encrypted within these ‘big databases’ of biomedical time series and images and take individual variability into account. Exploiting these data necessitates an in depth understanding of the physiology underlying these biomedical time series and images, the use of advanced digital signal processing and machine learning tools (e.g. deep learning) to recognize and extract characteristic patterns of health function, and the ability to translate these patterns into clinically actionable information. The creation of intelligent algorithms combined with existing and novel wearable and portable biosensors offer an unprecedented opportunity to monitor patients remotely (i.e. outside of the traditional clinical setting) and support the management of their condition.

In this course you will learn about aspects of information processing including data preprocessing, visualization, regression, dimensionality reduction (PCA, ICA), feature selection, classification (LR, SVM, NN) and their usage for decision support **in the context of healthcare**. The course aims to provide an overview of computer tools and machine learning techniques for dealing with **medical datasets** (time series and images). The course is practical with computer based tutorials and assignments. The necessary theory will be covered. The lectures are divided in three sets: ML basis, Popular classifiers and Deep Learning.

Each session is structured by two lectures and two hours of tutorial. All students will be expected to keep a digital log book of their code and results on their GitHub account for each tutorial session and share the results with the instructors by the end of the session – this will be used to track the class progress. Include your Python code, figure plots and explanations. Please label your figures clearly: parameters and units on both axes in a font large enough to be readable, with a legend describing each line and symbol you plot.

2. Course Summary

Course title:	Machine Learning in Healthcare (MLH)
Short title:	ML in Healthcare
Course ref. no.	336546
Number of credits:	3
Number of weeks: - **Weekly lectures** - **Weekly tutorials**	13 2 hours (total 26 hours) 2 hours (total 26 hours) + 3 workshops of 2 hours (total 6 hours).
Course assessment:	3 assignments (20%, 30%, 30%) Challenge (20%) Course attendance (bonus of 5%)
Capacity:	12 Work station
Computer requirements:	Six GPU available from department cluster. Software: PyCharm, jupyter notebook, Git, Atom. Libraries: Numpy, Panda, Keras.
Lecturer(s):	Joachim Behar (JB), PhD
Teaching assistants:	Alon Begin (AB), MSc candidate Moran Davoodi (MD), MSc candidate
Guests Lecturers:	Anne Weill (AW), PhD, Technion-BME Doron Shaked (DS), PhD, GE Healthcare Danny Eytan (DE), MD-PhD, Rambam Hospital Uri Shalit (US), PhD, Technion-IE Kamal Masarweh (KM), MD, Rambam Hospital Aviv Rosenberg (AR), PhD-candidate, Technion-CS
Teaching objectives:	Students will acquire the following skills: • Python for data science. • Main classifiers, intuition and mathematical background. • Deep Learning. • Structuring machine learning projects. • ML in healthcare.
Important deadlines:	Assignments: • A1: W06 • A2: W10 • A3: W16 Challenge: W12
Workshops dates:	W01, W06, W10.

“W” stands for week, “A” for assignment.

3. Syllabus

1.1 Part I: ML Basis

Week	Lecture	Subjects covered
1 **29/10/2019**	#C01 Introduction	- Course objectives and settings - Introduction to ML in healthcare - Supervised, unsupervised and deep learning - Polynomial curve fitting - Main concepts in ML - Notations features and variables
	#C02 Data Exploration and Preprocessing	- Exploratory data analysis - Data visualization - Abnormality detection and handling - Features scaling
2 **05/11/2019**	#C03 Technion-MLH Challenge	- Presentation of the Technion-MLH Challenge - Medical problem and dataset
	#C04 Linear Models for Regression	- Intuition - Calculus proof - Probabilistic proof - Sequential learning - Cost function - Gradient descent
3 **12/11/2019**	#C05 Linear Models for Classification	- Classification versus regression - LR hypothesis representation - LR Cost function - Gradient descent - Multiclass classification - Odds ratio - Linear discriminant analysis
	#C06 Regularization	- Overfitting - Cost function - Regularized linear regression - Regularized logistic regression - Ridge, Lasso regression - Geometrical interpretation
4 **19/11/2019**	#C07 Training a Classifier I	- Evaluating a model (train, validation, test sets) - Model selection, learning curves and error analysis - Bias-variance tradeoff - Cross validation approaches - Stratification - Information leakage - Generalization performance
	#C08 Training a Classifier II	- Performance statistics - Receiver operative curve - Training the final ML model.

1.2 Part II: Popular classifiers

Week	Lecture	Subjects covered
5 **26/11/2019**	#C09 Case study: Obstructive sleep apnea detection using Lasso	- Practical machine learning.
	#C10 Support vector machines	- Getting nonlinear (XOR), QDA - Maximum margin classifiers - Dual representation - Kernel trick - Grid search and random search
6 **03/12/2019**	#C11 Principal component analysis (Unsupervised Learning)	- Blind source separation - Principal component analysis - Change of basis - Mathematical proof - PCA in machine learning
	#C12 Independent component analysis (Unsupervised Learning)	- Independent component analysis - Statistical independence versus correlation - Whitening - Beyond ICA: t-SNE
7 **10/12/2019**	#C13 K-means and GMM (Unsupervised Learning)	- K-nearest neighbor - Probabilistic data analysis: GMM
	#C14 Causal inference	- Causal inference (US)
8 **17/12/2019**	#C15 Feature selection	- Lasso - mRMR - Genetic algorithms
	#C16 Machine Learning in Healthcare	- AI at the hospital, - Opportunities and ethical challenges (DE)

1.3 Part III: Deep learning

Week	Lecture	Subjects covered
9 **22/12/2019** **(instead of ג)**	#C17 Artificial Neural Network I: introduction	- Revisiting logistic regression - Introduction to NN - Notations - Representation learning - Forward propagation - Backward propagation - Activation functions - Multiclass classification (softmax)
	#C18 Artificial Neural Network II: training a NN	- Revisiting train-validation-test split - Weight initialization - Optimization algorithms - Revisiting bias-variance tradeoff - Batch normalization
10 **31/12/2020**	#C19 Artificial Neural Network III: hyperparameters tuning	- Grid search - Random search - Bayesian optimization - Vanishing and exploding gradient
	#C20 Deep Learning CNN	- Foundation - Convolution - CNN architecture - Striding, padding, pooling.
11 **07/01/2020**	#C21 Deep Learning CNN	- Popular CNN and image segmentation (DS)
	#C22 Deep Learning CNN	- Popular CNN and image segmentation (DS)
12 **14/01/2020**	#C23 Deep Learning	- High Performance Computing (AW)
	#C24 Deep Learning	- High Performance Computing (AW)
13 **21/01/2020**	#C25 Recurrent Neural Network	- RNN, LSTM, GRU - Autoencoders
	#C26 Challenge Final	- Challenge top 3 presentations and announcement of winning team.

3. Assignments

Assignment	Dataset type	Number of instances (n) and attributes (p)	Learning objectives
#A1 Cardiotography Data Set	The dataset consists of measurements of fetal heart rate (FHR) and uterine contraction (UC) features on cardiotocograms labelled by expert obstetricians. The aim is to automate the analysis of the FHR and approach the obstetricians' labels. http://archive.ics.uci.edu/ml/datasets/Cardiotocography	n=2126 p=23	- Feature engineered from physiological time series - Classification
#A2 ICU Mortality prediction	Features derived from physiological time series and demographics for the purpose of predicting what patients will die in the intensive care unit (ICU).	n=3000 p=35	- Classification-predicting what patients will die in the intensive care unit (ICU).
#A3 X-ray	Medical images.	n=400	- Deep Learning

4. Workshops

Workshop	Learning objectives
#WS1 Crash course on Python	Basics of Python and working environment.
#WS2 Supervised and unsupervised learning	Hands on supervised and unsupervised classification tasks.
#WS3 Deep Learning	Hands on deep learning classification tasks.

6. Challenge

The Technion-MLH Challenge

The Technion-MLH Challenge, denoted ‘Challenge’ in the following, aims to stimulate **rapid progress on an unsolved question of practical clinical significance** and that may benefit **improving healthcare outcome for the citizen of Israel and globally**. For that purpose, data will be provided to the students at the beginning of the course together with the medical question of interest. The dataset consists of a collection of variables and/or physiological signals that need to be analyzed toward a defined goal. The students are provided with a subset of the data, the “training set” that will be used by each team to develop their algorithm and a “test set” that will not be provided to the team and which will be used to analyze the student’s algorithm performance. The data and question we ask are novel and you will be the first to research it!

Challenge rules

1. Work is performed in teams of three students.
2. You are free to choose your teammates.
3. You are encouraged to collaborate between different teams and exchanges tips, observation and even code. This is why we call this a “Challenge”, the end goal is to solve a clinical question. However, each group must present their **unique** approach by the deadline.
4. Source code submitted should be in Python.
5. It is forbidden to transfer the complete or part of the dataset to someone outside the course. These are medical data and should be handled safely.
6. Publication of results in conference or journals based on the data provided by the Challenge is forbidden without prior consent of the Challenge organizing committee.

Evaluation

The marking of this work will be independent of your final score on the test set and will be based on the evaluation of your approach in tackling the research question (rational, experimental design etc.). Each of the following four parts will be assessed and each be worth 25% of the mark for the Technion-MLH Challenge assignment:

- Data exploration and preprocessing (e.g. summary statistics, data cleaning).
- Methods (e.g. standardization, classifiers, evaluation of generalization).
- Results (e.g. presentation of the key performance statistics).
- Discussion. In particular, discussion of the air pollution features in explaining the final classifier prediction (feature importance) and conclusion on the importance of these variables.

Technion-MLH Challenge prizes

The top-3 scoring teams on the hidden test set will be invited to the final and present their work during the last week of the course. A panel of judges will, based on the scoring of the teams and their presentation performance, vote for the Challenge winning team. The winning team will get the following benefits:

- Visit at the Rambam Children hospital.
- A prize certificate.
- Be invited to pursue their research up to publication.

5. Mathematical notations and terminology

Some notations used in this course are adapted from the notations of the Stanford CS230 course. Reference: https://cs230.stanford.edu/files/Notation.pdf

General notations:

(i)	Example number.
m	The number of examples in the dataset.
n_x	Number of features or input samples (input size).
n_y	Number of classes (output size).
$X \in \mathbb{R}^{n_x \times m}$	Input matrix i.e. matrix with input features n_x for all examples m.
$x^{(i)} \in \mathbb{R}^{n_x}$	Column vector of the i^{th} example.
$x_j^{(i)}$	Scalar value of the j^{th} feature for example i^{th}.
$Y \in \mathbb{R}^{n_y \times m}$	Target matrix i.e. matrix with targets n_y for all examples m.
$y^{(i)} \in \mathbb{R}^{n_y}$	Target label for the i^{th} example.
$\hat{y}^{(i)} \in \mathbb{R}^{n_y}$	The predicted output vector from the classifier.
$\underline{y} \in \mathbb{R}^{m}$	A vector of scalar targets for all examples m.
h	The hypothesis function.
f	Target function i.e. the function we aim to learn.
$\hat{h}$	The estimated target function using the hypothesis function h.
J	Cost function i.e. cost function for all m examples.[1]
E	Error i.e. for a single example.
$\mathcal{N}(\mu, \sigma)$	Normal distribution with mean μ and standard deviation σ.
$w \in \mathbb{R}^{n_x}$	Weights vector in linear and logistic regression.

[1] The function that we aim to minimize or maximize is called the objective function. As we are minimizing it is often called equivalently the cost function, loss function, or error function. The term “cost function” usually refers to an optimization problem and “loss function” usually refers to parameter estimation.

Notations specific for Neural Networks:

Hyperparameters in NN:

α	Learning rate.
β	Momentum
p	Mini batch size
K	Number of iterations for gradient descent.
$n_h^{[l]}$	Number of hidden units of the l^{th} layer.
L	Number of layers in a neural network.
$g^{[l]}$	Activation function for layer l.
k	Learning rate decay
	Features scaling method
	Other model specific hyperparameters (e.g. convolution kernel width in CNN.)

NN variables:

$W^{[l]} \in \mathbb{R}^{n_h^{[l]} \times n_h^{[l-1]}}$	Weight matrix for layer l.
$w_j^{[l]} \in \mathbb{R}^{n_h^{[l]}}$	Weight vector for j^{th} activation at layer l.
$w_{jk}^{[l]} \in \mathbb{R}$	k^{th} weight coefficient for j^{th} activation at layer l i.e. element of $W^{[l]}$ at (j,k)
$b^{[l]} \in \mathbb{R}^{n_h^{[l]}}$	Bias vector at layer l.
$b_j^{[l]} \in \mathbb{R}$	j^{th} bias activation at layer l.
$a^{[l]} \in \mathbb{R}^{n_h^{[l]}}$	Activation vector at layer l.
$a_j^{[l]} \in \mathbb{R}$	j^{th} activation at layer l.

Terminology

Example	Refers to a set of features describing an observation.
Target	The label we are aiming to learn to predict.
Hypothesis class	A space of possible hypotheses for mapping inputs to outputs.
Hypothesis function	An instance of the hypothesis class that maps inputs to outputs.

Acronyms

SGD	Stochastic gradient descent.
BGD	Batch gradient descent.