Applications: project presentation

Please don't Circulate !

Presentation anticipated to provide more time for the project

Presentazione anticipata per fornire maggior tempo per il progetto

Alessio Micheli

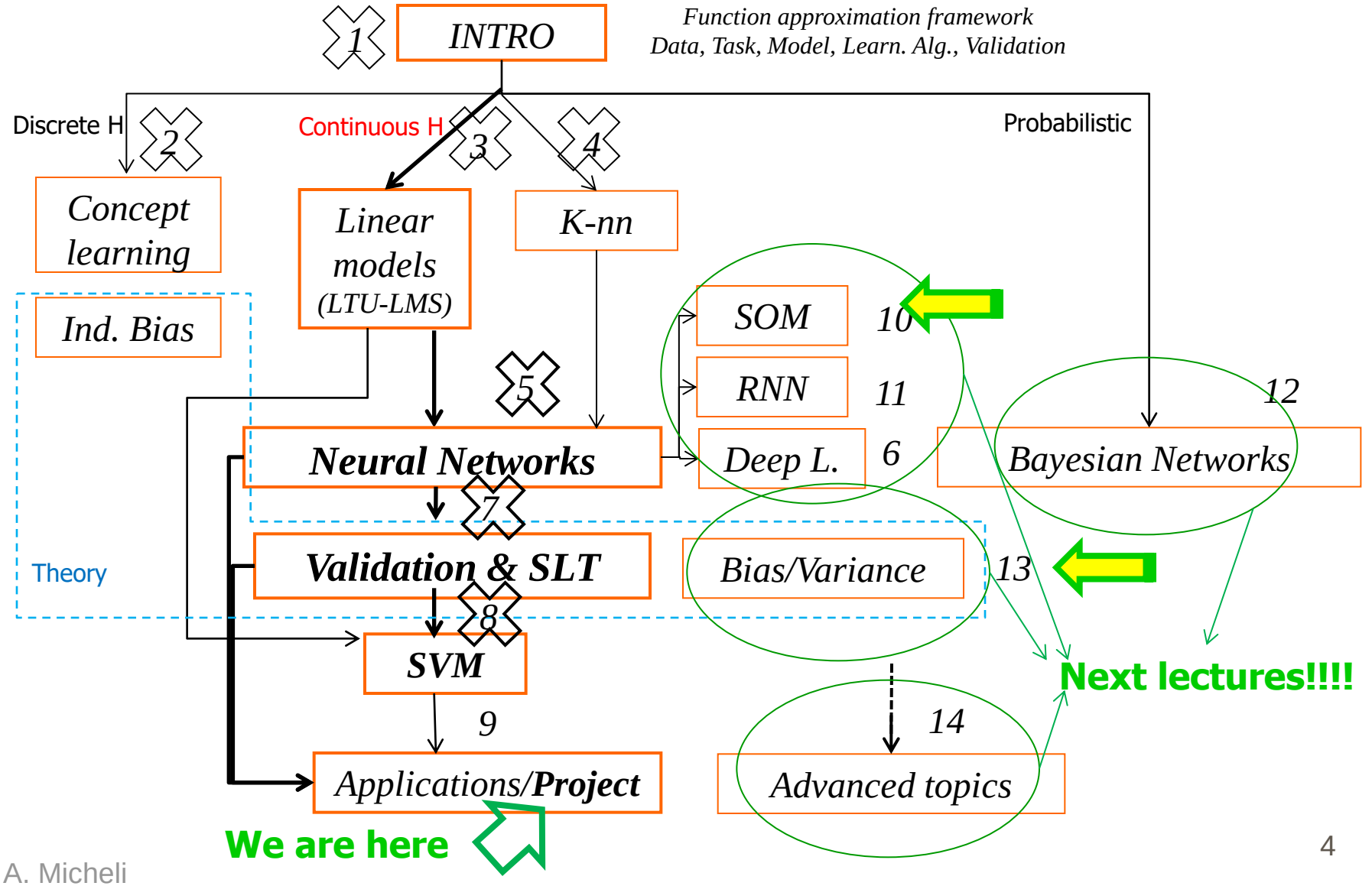
micheli@di.unipi.it

2019

Dipartimento di Informatica
Università di Pisa - Italy

ML Course structure
Where we go

Dip. Informatica
University of Pisa



Notes on the formal exam subscription

Dip. Informatica
University of Pisa

- Date for the exam: esami.unipi.it portal
- Therein you find the date for the oral (beginning)
- The prj must be delivered at least ~10 days in advance: the deadline *typically* oral date -~10 (But see the exact deadline date and hour in the Moodle folder for your session)
  - Of course You CAN always deliver before the deadline!

- Take care also to register your name in the official UNIPI (esami.unipi.it) portal for exams (look the deadline)

Assessment methods (REPETITA from introduction)

Dip. Informatica
University of Pisa

Exam:

- **Project** (*or* specific topic in-depth report) (*or* Written exam)
  - Students have the opportunity to develop a project realizing/applying a learning system simulator (typically a simple neural network) and to validate it through benchmarks. A written report will show the results.
    - Great *opportunity* to apply the concepts by yourself
    - Great *opportunity* to show your concrete understanding and effort for the exam
  - Deadline: ~(around) 10 days before the oral exam session (see the Moodle folder of your session for the exact deadline)
  - **See details in the lecture for project presentation**
  - We include a *competition* with *blind-test*
    - which is part of the benchmark results in the prj
    - 2019: also some joint proposals with CM course
- **Oral exam** (dates according to the exams sessions)

Exam (II): Some details (REPETITA from introduction)

Dip. Informatica
University of Pisa

- **Project**: we will discussed in a specific lecture all the details but from now it useful to know that:
  - It is made by a couple of students (exception MUST be justified and authorized in advance)
  - It is made just one time, i.e. with 1 delivery
  - It is corrected/discussed jointly with the oral exam (at the exam session)
  - It is an implementation of a NN or an application of ML SW (type A or B)
    - It includes code (type A) + results (also for the competition) + report (type A and B)
    - Type A: Programming language is free (we will discuss later)

- **Oral**: Prj discussion + questions on *all the course content*
  - Including written questions for all at the date of the official exam session
  - And then we distribute the dates of the prj discussion and oral refinement in the following days (for each group)
    - Style: I'll ask answer first by math language (e.g. in the form of *equations* due to clarity, synthesis, …; then we can discuss on them)

Exam: schematic synthesis
(REPETITA from introduction)

Dip. Informatica
University of Pisa

In the following order:

1. **Course lectures** (stay on-line with them!)
2. **Project work** (you can start around in the middle of the semester)
3. **Project package delivery:** results, report etc. at the date specified in the Moodle (around 10 days before the official exam session)
4. **Oral:** at the date of the official exam session (for the questions in written form) (it is in the same session of the prj delivery).
5. **Project discussion and finalization** according to a specific calendar for each group, i.e. as soon as possible in one of the following days (both the students must be present).

See the FAQ in the Moodle!!!

The main hints
(REPETITA from introduction)

Dip. Informatica
University of Pisa

- Follow the lectures and slides as a guide, studying *progressively* during the course
  - Special interactive classes will be used to asses activities and make a M discussion forum (Q&A classes).

NEW

- A major hint *from past students*:

1. **FIRST study the course content**
2. **THEN apply for the project**

- For the PRJ: Develop (implement) a self made NN simulator if you are self-motivated and with good programming skills (type A), else apply existing tools (type B)
  - There is no difference in the grade, but a different effort for implementing (type A), or to make experiments and comparisons (type B)

Good news

Dip. Informatica
University of Pisa

- This year we have a ML course assistant (Marco Podda: marco.podda@di.unipi.it)
  - to help for the project development
- Take the opportunity to interact with him during the course

- Also available via email/at CS department on Wednesday 16-18 (better if fixing an appointment by email in advance)

Project General Rules Material

Dip. Informatica
University of Pisa

- Deliver the report and material according to the deadline in the Moodle folder, with a package including (see also FAQ II):
  - **Code** (if it is developed by you, usually well commented)
  - **The report** resuming of the implementation and the experimental aspects:
    - Details later as: **Max** 10 pages (8 for 1 student group), **font 11** at least ! (a file describing in detail the report is in the folder for the project)
  - **Files** for the ML-CUP: "blind test set" + short abstract

    (see the next slides with details)

- Bring with yourself the code to the oral exam (printed or electronic version)

Project General Rules: How to deliver it

Dip. Informatica
University of Pisa

1. Use the Moodle (elearning platform) for delivery of the Project (PRJ) package: code, report, cup-files, short-abstract
   - See in Moodle the Section **Prj Student Material**

2. After the last upload (until the deadline):
   Send email at both micheli@di.unipi.it, marco.podda@di.unipi.it
   Subject: [ML-2019] Report by <your names>
   Include in the main text:
   - Your name(s) & email contact information in the main text
   - Your degree course (master programme)
   - The name of file used in Moodle (to find it)

It is a must **(!)** to use this **tag**

- Don't 'forget a CC to your colleague (all the group members must be included in the communications).
- Use the Moodle to deliver all the other files, do ***not*** attach them by email!!!
- Projects no announced by an e-mail will not be considered !!!
- (be sure I received it: resend the email if I do not reply with an ack.)

Project General Rules: Groups

Dip. Informatica
University of Pisa

- Autonomy: it is part of the evaluation
- **Groups**: *it is assumed groups of 2 students*
  - 1+1=2.5!
  - Increase autonomy and never discourage
  - Best results (in the general sense) in the past editions

- Can I do the project by myself only?
  No, read the FAQ with a link to "Partners for the project" in the Moodle:
  - Note that no individual projects will be accepted without a preventive agreement with the teacher (motivated by special need).
  - To help you to find a partner use the "Partners for the project" file
  - NEW: I opened it so that you can add or delete your name by yourself!!! (and delete it when you found a partner: it is open to free changes by you as google doc). The link is in the FAQ of Moodle.
  - You can use it also after the first sessions!
  - Making a trial through this document it is a must before ask for a possible individual project

The project

- Premise
- Type A, B, etc.
- The simulators
- The application

First premise
On the empirical approach

Dip. Informatica
University of Pisa

- Experimental method (using ***empirical*** evidence, from experiments) is the basis of the "*scientific method*" per-se.

- In our case, the numerical simulation to test your hypothesis on the model and the quantitative parameters.
  - (you don't know the result in advance, you have to formulate your hypothesis and then to measure the empirical results/observations, to compare and reason on the results etc.)
  - If you don't have any experience with experiments, this is new for you, but it is not negative but REALLY a nice and useful* experience you take the opportunity to gain !!! Exciting or lost? Break the ice !!!

* to build a skill for any field of science.

Second premise: The real aim of the project

Dip. Informatica
University of Pisa

The real aim of the project is not to achieve a state –of –the art result

- The NN or ML models implementation or usage is a mean, your understanding is my/ the objective
- Learning with NN or ML models is a mean, your learning on how such models work is my/the aim

The project

- Type A, B, etc.
- The simulators
- The application

Possible Aims: Type of projects

Dip. Informatica
University of Pisa

A) Realize a ML/NN model simulator and apply it (implementation)

- Programming language is a free choice (C++, Python are popular for ML, or even environments as Matlab or R can be considered)
- A) with CM : a coordinate prj with CM

B) Extensive experimental applications of existing ML/NN simulators (comparison)

- Simulator is a free choice (a list will be discussed in the next slides)

C) Contact me. Only in case there are reasons/impediment not to apply to A and B: we can for instance think to a report developing a ML topic or other written exam or e.g. (special case PhD student) a seminar during the course !

A) or **B)**: participate to the "**ML-2019 cup**" competition.

Let us see details for A) and B) and then the cup details.

A) Model implementation

Dip. Informatica
University of Pisa

Realize 1 ML model simulator : typically a **MLP Neural Networks (with regularization techniques)**

This is the typical project case, allowing you to realize a simple models by your original code, and to experiments all the variants that you like, and have fun;-)

Examples:

- [**typical case**] **Implement a MLP with backpropagation, momentum, and regularization L2** (possibly multilayers)**:** try the regularization and other hyper-parameters effects

- Backprop with variants for the weights upgrade : Quick-prop, Rprop, … or other gradient based techniques.

Other models (but not for the CUP):

- MLP for classification: LMS versus Cross-entropy*
- Bayesian models: contact me.
- **SVM**: see A) with CM

General note: you can exploit numerical libraries, e.g. NumPy, Armadillo (C++), ..

Special case <A) with CM>

Dip. Informatica
University of Pisa

A) with CM: A coordinate prj between ML and CM

See a detailed list of proposals within the CM course

- You will provide 1 report for ML with the basic A) results + the new result with the CM technique (comparison)
- AND 1 report for CM (according to CM teachers rules)

Categories (examples) [through CM approaches] :

- (improved) NN training by new descent methods algorithms
  - see many examples in the lectures NN2 Heuristics (Conjugate gradients, …)

  [comparing wrt the basic gradient descent with momenutum and L2 regularization]
- Non-differentiable optimization for Piece-Wise linear functions
  - PWL or ReLU activation function or *L1 regularization*

  [comparing wrt the basic gradient descent with momenutum and L2 regularization]
- SVM/SVR implementation (through different approaches), applied with kernels...
  - [for this case NN comparison is not need]

The proposal must be agreed by ML AND CM teachers!!!

You can still compete for the ML-CUP with your results.

<A) with CM> only for group of 2 students.

These are challenging prjs
More (double?) effort!, volunteer choice!

Notes on <A) with CM>:

Dip. Informatica
University of Pisa

Further details on<A) with CM>:

- More effort? It is because you like to try and show something more and not something less! This fully exercises your full understanding of the singular parts (by adapting them to the combined construction)
- For may cases it is useful to refine the literature basis (provide by us) to see previous studies of the impact of the used methods for the ML area.

Notes:

- L2 regularization is the Tikhonov with norm-2 penalty
- L1 uses norm-1 penalty (we will discuss it later)
  (see lecture on linear models and NN-part2)
- Also ask for PWL if you are interested, ReLu will be discussed later

- Note that the aim is not to improve the performance/results obtained with the basic approach, but to critically exercise the use of CM approaches for ML*
- Unforeseen issues? Discussed by a case-based approach*

B) Compare models

Dip. Informatica
University of Pisa

- **Extensive experimental applications of existing ML/NN simulators**
- Compare different models/ existing tools e.g.:
  - SVM or NN by standard library versus basic models (linear, k-nn, naïve bayes, …) implemented by yourself or by standard software tools
  - Compare NN vs SVM vs other models (even not included in the ML program) within the same software tool.
  - Compare 3 or more models even from different sw tools.
  - Compare different software tools for the same model (e.g. SVM or NN).
  - Compare different software tools.
- (also for fairness wrt A) the B case implies a larger effort on the comparison among models (including accurate validation) and to the experimental part
- The report can include also evaluation on the sensitivity to the hyperparameters values (for different models), efficacy, efficiency, predictive performance (of course!) but also issues of tool usage, usability, richness of the set of hyperparameters etc. (for different tools)
- Repetita: you can still apply to the CUP competition just selecting the best model to apply.

For both A) and B) cases

Dip. Informatica
University of Pisa

- For **NN heuristics** see the lecture on "Neural Networks: part 2"
- Try the effects of different configurations/ hyperparameters values according to your experimental schema

  (and explain the schema in the report)
  - In any case include the momentum and a regularization approach (weight decay, early stopping, ….). See all the **(!)** indications

- If you use a library, please not limit yourself to default values!

- You are free to choice the **model selection/assessment-evaluation** strategy:

  whose fitness for the problem at hand is evaluated:
  - Directly from the description in the report
  - Through the results on the blind test set

FAQ: When is a project A or B?

Dip. Informatica
University of Pisa

- If you program the NN model and back-propagation (or any training algorithm), and the CV approach, from scratch and by your-self (independently from the software environment) is type A. E.g.:
  - A NN made by you within the standard Matlab is type A (but see FAQ I)
  - Instead, a NN using Matlab NN tool box is type B
  - The same for TensorFlow, PyTorch, Keras and other frameworks: it is type B.

- Using high level libraries (SciKit learn, Keras etc.) is type B
  - However, if for example you just exploits part of the PyTorch features (e.g. the "automatic differentiation") but you implement the main training loop, the CV etc. you can indicate it in the report.
- Using numerical or graphical libraries in your code maintains it of type A.

The project

- Type A, B, etc.
- The simulators
- The application

Simulators/Software Tools

Dip. Informatica
University of Pisa

- Software / Tools to be used for the **B case**
- Or you can use a tool as an "oracle" to compare with your simulator for the **A case** (helping in assessing its correctness)

- If you use a library you must specify in the report the complete link to the source !!! (for both A and B cases)

- In the following some examples: an exhaustive list is out of our scope and it is even impossible to keep it updated!!!
  - Developers make changes everyday!!! (big companies have a major role)
- The best one is the one that is more useful FOR YOU.
  - Check also the documentation and if it still have maintenance/developments (new releases, support, …)
  - Some of them have a reference in [JMLR]= http://jmlr.csail.mit.edu/mloss/

Simulators/Software Main (ML & NN & Deep)

Dip. Informatica
University of Pisa

- **Keras** is a high-level Neural Networks/Deep Learning library (Python) capable of running on top of either TensorFlow , Microsoft Cognitive Toolokit (formerly CNTK), Theano and others, (+Wrappers versus Scikit-Learn API).
- **PyTorch** is an open-source machine learning library for Python, based on Torch (was in C++). Neural Networks, Deep Learning. It exploits "automatic differentiation"*. Developed by *Facebook AI* research group (and Uber)!
  - **Torch** (NN, SVM, AdaBoost, K-nn, Bayesian, …renewed for deep learning/GPU, coll. with Facebook, Google, Twitter …development stopped in 2018)
- **Theano** (and Pylearn2): also for deep learning, GPU etc. wrappers with scikit-learn, … (stop dev. on 9/17)
- **Caffe** is a deep learning framework, originally developed at UC Berkeley. It is open source, C++, with a Python and MATLAB interfaces
- **TensorFlow** (*Google*), open source, C++/Python like (API, interface). Include deep neural networks. Released as open since November 2015. Now: TensorFlow 2.0.

- Other: See http://deeplearning.net/software_links/

Simulators/Software Main (ML & NN & Deep)

Dip. Informatica
University of Pisa

- **Scikit-learn** Python open source, many tools for preprocessing, model selection, linear models, regularization, SVM, DT, … … NN only from version 0.18 – even if still in a basic form, but wrappers for other sw exists [JMLR 2011]:
  - very good documentation with graphical examples: HAVE A LOOK!

Environments:

- **R** (Statistical Computing language): programming language and free software environment for statistical computing
- **Matlab**: multi-paradigm numerical computing environment and proprietary programming language (available at Unipi),
  - or as a simulator for NN ("Deep Learning Toolbox")
- and **GNU Octave** (free, Matlab compatible)
- **Wolfram Mathematica** (mathematical computation integrated tolls)

Others … (not read in class)

Dip. Informatica
University of Pisa

- **MXNet:** by Apache, is open-source deep learning software framework. Deep neural networks. Multiple programming languages.
  Related to Gluon : Amazon + Microsoft (2017)
- **MLlib** Spark scalable ML library (NumPy, R, Hadoop), NO NN yet [JMLR 2016]
- **Shark** (evolutionary and gradient-based algorithms, NN, kernel-based learning methods, SVM, …[JMLR 2008]): C++ library
- **Dlib-ml** (Bayesian networks and kernel-based methods, clustering, anomaly detection, and feature ranking, … [JMLR Jul 2009]): C++ library
- **Shogun** (SVM, HMM, K-NN, LDA,…[JMLR 2010]): C++/Python
- **Mlpack** (C++, armadillo matrix library, basic models, no NN/SVM, [JMLR 2013])
- **OpenNN** (Open Neural Networks Library) C++ (bugs!)
- Eclipse **Deeplearning4j** is a deep learning programming library written for Java
- Others for Python: e.g. PyMVPA, MLPY , PyML, Plearn, **PyBrain** [JMLR 2010]: **, …** check if still supported)

See also: https://en.wikipedia.org/wiki/Comparison_of_deep-learning_software

Historical:

- **Stuttgart Neural Network Simulator: C, since 1995!**
- **SOM_PAK: C++ /** SOM toolbox (free)

Other examples (cnt)

Dip. Informatica
University of Pisa

And many other…(started as DM tools).

- **Weka** (DM and ML Software in Java)
- RapidMiner
- Orange (JMLR 2013, C++/Python)
- ….
- Many commercial software ! …→ "*predictive analytics*"

Visual workbench approaches are popular for commercial tools:

- E.g. (by students) **Knime**: data analytics platform: DM but also ML/NN (open source/commercial**!**)
- Major sw developers have now a ML branch: azure (Microsoft), google, facebook, IBM, Amazon ….

Just to witness the interest of the big companies

Dip. Informatica
University of Pisa

Just to witness what happens (not for your prj *):

- Google: google prediction (service) / **TensorFlow** *
- IBM: Watson
- Microsoft : Azure (platform and services)/ The Microsoft Cognitive Toolkit (formerly **CNTK)**, is a deep learning framework developed by Microsoft Research.
- Amazon Machine Learning (service) / (old) DSSTNE: Deep Scalable Sparse Tensor Network Engine (open, Deep ML)
- Facebook: FBLearner Flow, PyTorch…

- ……

Simulators for SVM

Dip. Informatica
University of Pisa

- **LIBSVM** (C++)
- SVM light (C)
- Torch (C++)
- JKernelMachines (Java library for learning with kernels [JMLR 2013]).

- Weka (Java)
- mySVM
- SVM in R
- ….
- http://www.support-vector-machines.org/SVM_soft.html

Popular Supporting Libraries

Dip. Informatica
University of Pisa

- **Pandas**: software library written for the Python programming language for data manipulation and analysis
- **NumPy** (*popular for NN!*) (**SciPy**), **Armadillo** (C++): numerical/linear algebra libraries: suggested also for prj of type A!!!

- https://en.wikipedia.org/wiki/List_of_numerical_libraries
- https://en.wikipedia.org/wiki/Comparison_of_linear_algebra_libraries

General note: you can exploit such numerical libraries for your code!

The project

- **Type A, B, etc.**
- **The simulators**
- **The applications**
    1. MONK
    2. CUP

Applications for the PRJ

Dip. Informatica
University of Pisa

Your results will include:

1) MONK benchmarks

2) The CUP data set (competition)

In the report: results for 1) and 2)

1) MONK benchmark [repetita]

Dip. Informatica
University of Pisa

- Difficulty of assessing implementation correctness:
- A first test *("collaudo")* **Monk data set**
  - **The results *must* be reported in the prj report: performance and learning curve plots for the 3 monk tasks**

    http://archive.ics.uci.edu/ml/datasets/MONK's+Problems
  - 3 tasks of binary classification, small artificial data set, "not difficult" (a small NN with few units achieve a very high accuracy, up to 100%, with small time of convergence)
  - There is a report with previous results using MLP (and others ML models)
    - chapter 9 http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.45.2363

Solve it!

Input encoding: 1-of-k → correspond to introduce 17 input units (see also sec 1.1)

  - TS includes TR, which is a bad practice, but does not change here (since 100% accuracy is 100% accuracy also on the test set !!)

- Other sources of data sets for software tests: http://archive.ics.uci.edu/ml/

  UCI Machine Learning Repository (hundres of data sets !)

Monk data set results

Dip. Informatica
University of Pisa

- Please see examples of plots (learning curves for MSE and Accuracy) in previous lecture on NN:

  the "Neural Networks: part 2" lecture (file: ML-**-NN-part2-…pdf)

  - See "Other suggestions for your first trials" in the same lecture
  - and examples of Hyperparameters were specified in the slides

    - Good results on the MONK benchmark does not guarantee the simulator correctness
    - Bad results on the MONK benchmark for sure require revision of the code/setting

- Also: the results on Monk should be achieved using a small network (2-4 units): The test works if you achieve the state of the art results with few units (you have to not to improve the accuracy).
- For the CUP, on the other hand, there are no constraints, because instead you have to do the best performance ...

2) ML CUP !

Dip. Informatica
University of Pisa

ML CUP 2019

Dip. Informatica
University of Pisa

I provide to you 1 data set for the cup

<u>For the data set:</u>

- I provide a training set and a blind test set (examples without target values)
- Apply the models that you are investigating selecting the final one that you think it is more accurate through the training set (used for training and validation, and your *internal test **) (*see the FAQ).
- Report in the "report document" the TR/VALIDATION and your internal TEST errors, in the original scale i.e. MEE for the 2019 cup (see next slides).
- Apply your final model to the blind test producing an output for each example of the blind test set and record them into an output file
- Provide with the report the output file containing the blind test results
- Test result: the accuracy on the blind test set will be automatically computed
- The final results will be summarized on the web site using your nicknames
- Glory to the winner! ;-)

Tasks and Data

Dip. Informatica
University of Pisa

- **Regression** on 2 target variables, i.e. x,y coordinates in a 2D space
  - FAQ: regression = linear output units
  - 2 target variables: 1 NN with 2 output units, or 2 NNs, 2 SVR, ….

- 1765 training examples
- Column 1: pattern name (id)
- Central columns: 20 variables with continuous values (from a noise source, real sensor data)
- Last 2 column: target of two continuous value variables : x and y.

- Blind Test set: 411 patterns, with the same input format (of course without the 2 target columns)

Task & Errors 2019

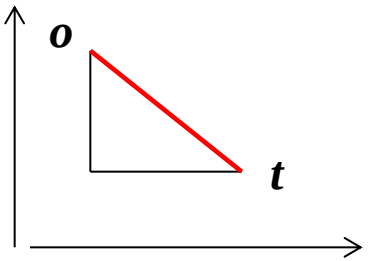
Dip. Informatica
University of Pisa

Report in the original scale the following error measure (**Euclidian distance**) :

- **Mean Euclidian Error** (it is the error used for the competition performance evaluation), *where N=number of data, p=pattern, o=output, t=target*

$$E_{MEE} = \frac{1}{N}\sum_{p=1}^{N}\left\|\boldsymbol{o}_p - \boldsymbol{t}_p\right\|_2 = \frac{1}{N}\sum_{p=1}^{N}\sqrt{(o_{p,x}-t_{p,x})^2+(o_{p,y}-t_{p,y})^2}$$

Distance between 2 points in 2 dim space



- Typically you will also observe the (root) mean squared error (that is the typical loss used for the LMS training approach)

$$E_{MSE} = \frac{1}{N}\sum_{p=1}^{N}(\boldsymbol{o}_p - \boldsymbol{t}_p)^2 = \frac{1}{N}\sum_{p=1}^{N}\left((o_{p,x}-t_{p,x})^2+(o_{p,y}-t_{p,y})^2\right)$$

MEE≠RMSE since sum(root(a), root(b)) ≠ root (sum(a,b))

What is provided 2019

Dip. Informatica
University of Pisa

- Training set
  - ML-CUP19-TR.csv
- Blind test set (without target)
  - ML-CUP19-TS.csv

!!!!!

WHERE:

- **ML section on Moodle: https://elearning.di.unipi.it/**
- **Folder** ML-19-PRJ lecture & package
- With files for data, info (slides and txt), report-demo, results-demo

- Note: editions 2010-2018 **are NOT used this edition/ year**

What to produce 2019 (I)

Dip. Informatica
University of Pisa

- Output file in a simple text/*txt* format. The name must be:
  - *team-name*_ML-CUP19-TS.csv
- Using the following format
  - First 4 rows are for comments :
    - # your name/names
    - # team (nickname max 8-10 char) for the web results
    - # data set name (ML-CUP19 v1)
    - # date (e.g. 20 Dec 2019)
  - Table with 411 rows and 3 columns (comma separated values):
    - id, output_x, output_y
    - id ordered from 1 to 315 (exactly as for the file ML-CUP19-TS.csv)

    See the demo file: ***output_template_example-with-random-output-ML-CUP19-TS.csv*** (which is filled with random-output values for demo)

PLEASE, double check the output file format!!!!

If it is not OK we cannot evaluate it automatically→ jump to the the bottom of the rank !

Repetita: you can send only 1 *team-name*_ML-CUP19TS.csv file (assuming it is your best result)

What to produce (II)

Dip. Informatica
University of Pisa

- File: team-name_abstract.txt, in a simple text/*txt* format
- With a very short (5 rows) description of the used model and validation technique.

- HENCE, you have to send (for the cup):
  - *team-name*_ML-CUP19-TS.csv
  - *team-name*_abstract.txt

- Along with (see the "Project general rules" slide at the beginning of this lecture)
  - Your code
  - Your written report (with results on 3 MONKs and the CUP)

Results

Dip. Informatica
University of Pisa

- Initially I will personally communicate the result to the single participant during each oral
  (for fairness with participants following in time)
- At the end of next year it is possible to publish the final ranking
- And the winner ;-)

- Criteria for the winner on the task: accuracy (MEE) and possibly the quality of final plot of the results (at the discretion of the jury ;-))

Finally

- **The report**
- **Other hints/request for the prj**
- **FAQs**

The report

Dip. Informatica
University of Pisa

- See the DEMO/Template file: **ML-19-Report-template-v*.doc**
- Check the last version (vx.y)

- The demo file includes descriptions of the information to be provided by the report:
  - these are mandatory to accept the report as valid for the exam
  - **double check you reported all the needed information**

- Please follow such basic organization for your document

A zoom for Screening phase:

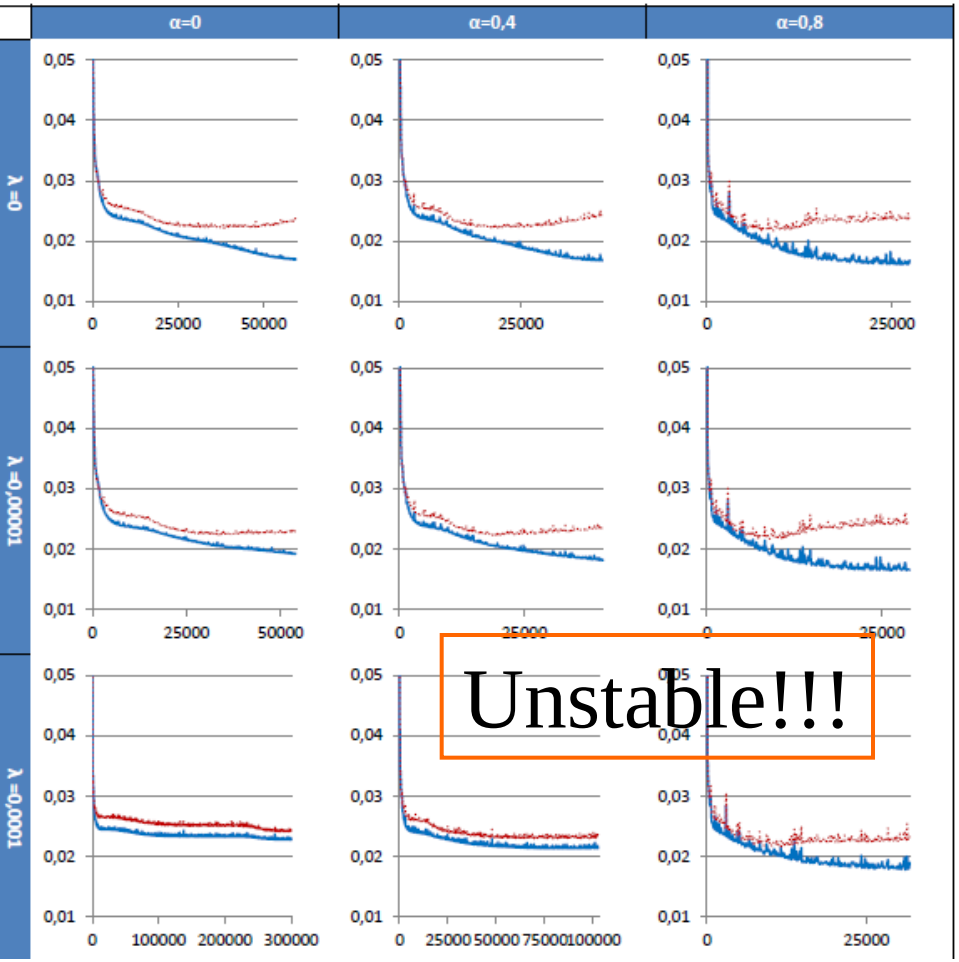
Dip. Informatica
University of Pisa

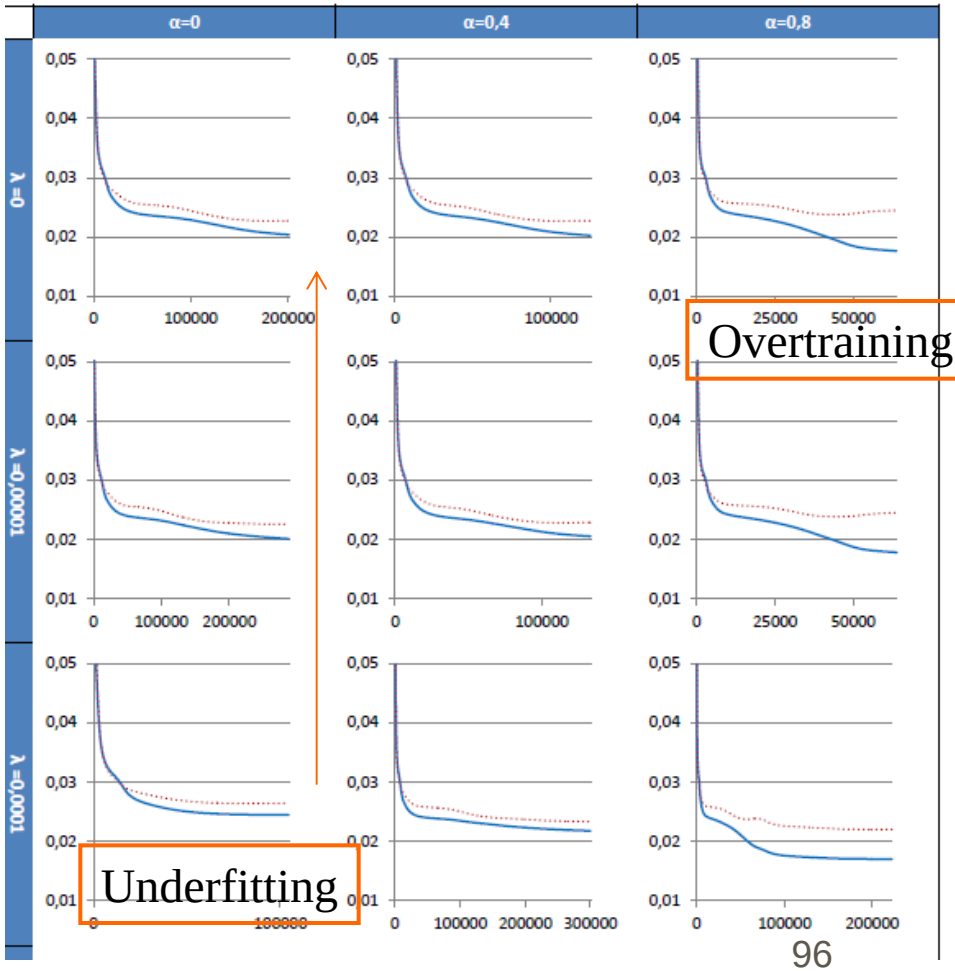
- Initially you will try different values of hyper-parameters
- (if you think interesting) significant cases of leaning curves can be reported in the report
- **but for yourself** plot them for various combination* →the screening phase is essential for yourself to learn from this experience [see next slide demo]
- Results of this kind (also from the grid search) could be part of the appendix

Screening demo

Dip. Informatica
University of Pisa

- Some instances of screening phase (don't mind of the specific demo hyper. values, look to behavior changes…and make the same with your values)





How to evaluate your work (I)

Dip. Informatica
University of Pisa

- **Autonomy**
- Soundness and quality of the (code) simulator (including any characteristics of modularity, efficiency, ...)
- Proper operation and behavior:
  - e.g. see the regularity of learning behavior on the plots (see the FAQ), …
- **Pertinence of the choice** for the model selection and evaluation
- Extension and/or depth of your investigation
  - Model and/or algorithms variants
  - Hyperparamters range
  - Validation and assessment …(**accuracy/rigorousness of the validation**)

How to evaluate your work (II)

Dip. Informatica
University of Pisa

- **Quality of the report**:

1. completeness (adherence to requests, see the template file),
2. correctness, 3. synthesis, 4. depth.
And in general: organization, rigor, soundness and synthesis (no repetitions); motivations suitability; choices made; breadth and depth of the experimental investigation (although expressed in a concise way); **clarity of the validation schema**.

- In case: difficulty of the challenge (if it is well done) can be a plus (e.g. <A) with CM>)

- CUP: The **blind test** result is just one of the possible parameters (not the most important one): however, very very often it strongly depends on the quality of the model and of the validation approach.

- The ultimate aim is to evaluate the **maturity** of your approach.

FAQ 0

Dip. Informatica
University of Pisa

Typical serious matters to make possible the correction:

- Respect the number of pages limits: the pages out of the limit will be not read and therefore not evaluated!!!
- Respect the minimum font (11)
- Respect the format of the CUP results (and range values): see the file "output_template_example_with_random_output-ML-CUP19-TS.csv"

FAQ I

Dip. Informatica
University of Pisa

- **Deadline**: ~ (around)10 days in advance to the oral session is the deadline, before is better (e.g. sometimes additional info are required), **and later is indeed in advance for the next session.**
  - See the **exact deadline** for the project material delivery in the Moodle folder for your session, i.e. in Prj Student Material --> Session xx (your session of interest)

- **Groups** of 2 people is neeed (see previous slide)
  - The evaluation (final mark) can be different for the two students (oral is different)

- What change using **MATLAB/OCTAVE/R?**
  - If you code by yourself → prj type A,
    but exploit your advantage (less time for the coding phase) to use more time in exploration/usage of advanced Matlab numerical computing functions/ possibly a comparison with other available models in the environment/extensive cross-validation/ impressive graphical results/….
  - If you use the NN toolbox → prj type B

FAQ II

Dip. Informatica
University of Pisa

- Is there any **CHECK LIST** for material delivery?
  - Previous slides on <Project general rules> and < What to produce> (2 slides)
  - For the **report**: see the template file with [*] for mandatory info on the results
    - Template file: ML-19-Report-template-v*.doc
  - Check that cup **result file** has the proper name and format
  - Provide the following files:
    - Your code (if original, else use a link to libraries in the report)
    - Your written report (with results on 3 MONKs and the CUP)
    - *team-name*_ML-CUP19-TS.csv
    - *team-name*_abstract.txt

- **CSV** format for the CUP results: see the files, it is a comma-separated values without spaces after the comma. Each pattern is a row. The header of input file has some rows of description beginning with #.
  - ***Example:*** output_template_example-with-random-output-ML-CUP19-TS.csv (which is filled with random-output values for demo)

FAQ III

Dip. Informatica
University of Pisa

- What if my model **does not work well**?
  - Check the learning curve to find clues
  - HINT: Try to compare with respect to a known tool in the same condition
  - Take advantage of the Q&A class (with also the assistant)
  - But…: Autonomy is part of the evaluation

- **How many trials** ? How many do you feel useful in order to do a good model selection. Such choices are part of the quality of your work.

- **Which trials** should I report? All the significant cases with details (experimental evidences), the other can be mentioned/synthetized in the text. Such choices are part of the quality of your work.

- **MSE** or **MEE**? You can use MSE (LMS) for training and MEE to evaluate

- Is it need an "**internal test" result**? Yes, it is, so we can compare your results with the blind test results (for our didactics aims)
  See also the template demo file for the report.

FAQ IV

Dip. Informatica
University of Pisa

- Plots , graphics etc.: the printer is **Black&White**: it is mandatory to distinguish lines in the plots also using different lines symbols/style (to see them also in B/W)

- Report Format: free style (but **font** >= 11), typically **PDF** (include a pdf copy in any case).
- Italian or English? The language more easy for you ;-)

- Executable programs, libraries, … (**large files**): until the package is small size try to include everything in the package.
  If you have problems include (in the report and in the READ-ME file) a link to download the *** accessories files *** of great dimension

FAQ V

Dip. Informatica
University of Pisa

Even more QUESTIONS?

Please read the following slides (other FAQ) before the next Q&A lecture!!!!

FAQ (NOT read in class I)

Dip. Informatica
University of Pisa

Please, read them by yourself :

- It is important to report the final Training, Validation and the internal Test error by the MEE measure?
  Yes, it is a must. I cannot accept a report without these values.
- It is important to report how we obtained the final model (how selected)?
  Yes, it is a must. I cannot accept a report without this information.
- Can you provide to me an idea of the basic result for the CUP? Should I have near 0 errors like Monk?
  No, zero errors is ONLY for Monks, an exception. For reasons of equity, no indications can be given on the reference performance for the CUP.
- How many layers do you suggest for the CUP?
  A single hidden level is sufficient, but I strongly suggest you to try also two (or more) and compare. It is a plus for the evaluation.
  This holds also for student with 6 credits exam (AA1) without following the final part of the course.
- The error to be reported in the tables is without the penalty term? Yes, without it
- General note on plots in the report: to be comparable, use a point for each epoch (defined as the total number of training data): if you are using on-line or mini-batch provide the mean over an epoch for each point in the plot.

FAQ (NOT read in class II)

Dip. Informatica
University of Pisa

Please, read them by yourself :

- Can we use other methods beyond grid search?
  Yes, but first grid search is mandatory. E.g. Random search : see Validation 1 (Valid1) lecture. It is needed to teach to you the meaning of the hyperparameters!
- Is it needed an advanced validation approach for Monks?
  No, Monk dataset is a benchmark to asses your code and system. You can also asses the validation process with Monk, but it is not needed to address this task by a complex CV approach.
- Making a prj with SVM, which plot can I do?
  See E.g. Slide 6 <ML-17-SVM-other inform> lecture (results w.r.t. hyper-parameters values, including epsilon-tube parameter)
- Plots with MEE or MSE?
  MEE for the tables (and this is mandatory!), for plots up to you (typically the quality of the learning behavior does not change)
- Usage of Early Stopping:
  See Validation 3 (Valid3) lecture
- Prj A with CM: comparison also on Monks?
  The results on MONK are still needed (at least summarized if the not interesting for the basic or the advanced algorithms), and of course you have to apply to the CUP dataset.

FAQ (NOT read in class III)

Dip. Informatica
University of Pisa

Please, read them by yourself:

- Does Monk converge always with few hundred of epochs"?
  The model converge easily in "most of the case" with the suggested setting (and also with other settings). But not "always" due to the sensitivity to the random initialization using a large range of the initial weights. Use a very small range! Please see in the lecture NN2.
- So what accuracy have to be reported for the Monk, just the best over different trials/initializations?
  No, the mean accuracy over that trials. See the discussion above.
- How many units for Monk? Follow the manual as baseline: see a slide in this lecture!
- How many units for the CUP? You have to search by yourself, but it is completed unrelated with respect to the Monk task!!!
- What if my learning curve is unstable? Try to solve the issue! A smooth learning curve can be obtained for these tasks by a correct implementation and a suitable setting ! Else, I have to consider it as a not good result.

FAQ (NOT read in class IV)

Dip. Informatica
University of Pisa

Please, read them by yourself:

- Can we collaborate with other groups for specific issues? We encourage to exchange ideas and to solve some common issues with other students (e.g. is the W update rule correct in my code)? However, of course, the project will be completely independent, and you must show your autonomy in developing it.

- Project B: Do we have to compare the model on both MONK and the CUP data sets? Yes, you have to compare the models on both the Monk datasets & on the CUP dataset.

- What is the time to be reported when yoyu say «Provide an estimation of the (training) computing time (and of your HW resources)" in the template doc for the report? A measure of the time like the mean time (e.g. in second) for an epoch of training (and the description of the computer that you are using to make this training).

Motivational hints by past students

Dip. Informatica
University of Pisa

Selection of general nice hints/motivations by past students reports conclusions:

- "The powerful libraries available nowadays provide a high level of abstraction and while this is useful for quickly getting things done, a naive use of a user with no solid knowledge of **the theoretical ground** may lead to a disastrous outcome. […] In conclusion, by implementing our own library, we feel that we are now much more aware of whats happening behind the scenes."

- "The limited time and computing resources **has pushed us to think** more about reasonable choices or shortcuts to speed up the searching for a fine model selection for the Cup dataset, showing us how necessary is the integration between artificial and human intelligence."

- "Last consideration is on how important is to **visualize results in the best way**. Not only it is fundamental for presentation itself, often it is the fastest way to understand how the model behaves in the hyperparameter space."

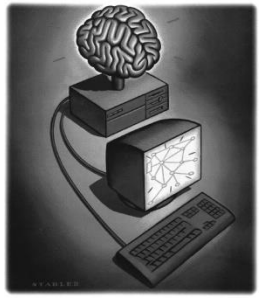
- "Lo sviluppo di questo progetto è stato sicuramente un'ottima motivazione per mettere in pratica quanto affrontato nel corso, sia in termini di implementazione e comprensione degli algoritmi alla base delle tecniche più moderne di Machine Learning, sia in termini di applicazione di una rigorosa metodologia per la valutazione e soprattutto la validazione di un modello, indispensabile per raggiungere risultati plausibili e confrontabili con quanto studiato in teoria. **La coordinazione che un lavoro condiviso come questo richiede è inoltre un ottimo 'training test' per lo sviluppo in gruppo**, ed il confronto continuo durante le diverse fasi dell'implementazione e validazione ha benificiato entrambi nella comprensione del lavoro svolto."

- "I **concetti teorici trattati a lezione** di cui fanno parte ad esempio le tecniche di regularization […] e le nozioni di underfitting/overfitting di un modello sui dati sono stati visti in un ambito più pratico e è stato molto utile ai fini didattici. Le richieste ci hanno permesso di scoprire più a fondo il mondo del Machine Learning e in particolare quello delle Neural Network, infatti abbiamo avuto l'occasione di mettere le mani su dinamiche e funzionamenti che dall'esterno risultano trasparenti."

Enjoy !

Dip. Informatica
University of Pisa

- … Now you can **enjoy** with ML !



Remember to have
FUN !!!

Future – just ahead

Dip. Informatica
University of Pisa

- CIML@Pisa
- Didactics: other courses (ISPR, CNS,…)
- General topics for ML applications
- CIML@Pisa research

2019 Note: THIS PART WILL BE PRESENTED LATER,
If you are a AA1 student (6 credits) but you like to see them, please contact me to know the date of the lecture

DRAFT, please do not circulate!

For information

Alessio Micheli

micheli@di.unipi.it

www.di.unipi.it/groups/ciml

Dipartimento di Informatica
Università di Pisa - Italy

Computational Intelligence & Machine Learning Group