Teaching Machine Learning to a Diverse Audience: the **Foundation**-based Approach

Hsuan-Tien Lin, National Taiwan University

Malik Magdon-Ismail, Rensselaer Polytechnic Institute

Yaser S. Abu-Mostafa, California Institute of Technology

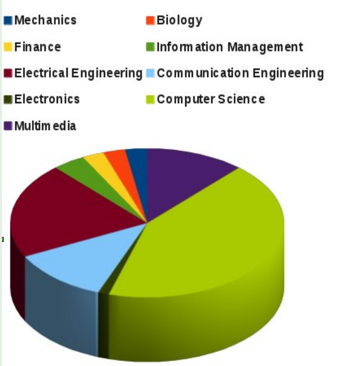
Teaching Machine Learning Workshop @ ICML 2012

June 30, 2012

Diversity in ML classes

NTU ML 2011 Fall (77 students)

- background diversity



- "maturity" diversity
  - junior: 8
  - senior: 20
  - master: 44
  - phd: 5
- similarly diverse in RPI and in Caltech (online course)[1]
- **challenge**:
  serving CS students while accommodating the needs of diverse non-CS audience

mindset of the audience?

[1] `http://work.caltech.edu/telecourse`

Observed Mindsets of the Diverse Audience

- highly motivated to learn
  —not satisfied with only shallow comic-book stories
- often with minimum but non-empty math/programming background
  —capable of downloading and trying the latest packages

words of a student from industry (Caltech online course 2012)[2]

While it's easy to pick up a couple of algorithms from the many text-books and online materials out there, it is the solid foundation, both mathematical and practical, as well as this better intuition that I would have missed studying alone without this class. Also, the cadence of the lectures and the

demand: **solid foundation** (and better intuition)!

[2] `http://book.caltech.edu/bookforum/showthread.php?p=3107`

Our Proposed Teaching Approach

- foundation-based, and foundation-**first**
- then, compensate foundation with **a couple of** useful algorithms/techniques

comparison to techniques-based

- techniques-based:
  hops through the forest of many latest and greatest techniques
- **foundation-based**: illustrate the map (core) first to prevent getting lost in the forest

foundation-based:
prepare students for easy learning of untaught/future techniques

Our Proposed Teaching Approach [Cont.]

- foundation-based, and foundation-**first**
- then, compensate foundation with **a couple of** useful algorithms/techniques

comparison to foundation-later

- foundation-later:
  - first, techniques to raise interests
  - then, foundations to consolidate understanding
- **foundation-first**: build the basis (core) first to perceive the techniques from the right angle

foundation-first:

let students know when and how to use the powerful tools before getting addicted on the power

Our Proposed Foundation: Three Concepts

understand learnability, approximation and generalization

- when can we learn and what are the tradeoffs?
- conducting machine learning properly

use simple models first

- the linear model coupled with some nonlinear transforms is typically enough for most applications
- conducting machine learning safely

deal with noise and overfitting carefully

- how to tackle the “dark side” of learning?
- conducting machine learning professionally

our experience: worth starting with those foundations,
even for a diverse audience

learnability, approximation & generalization —conducting machine learning properly

$$
\begin{aligned}
& \text{good learning (test performance)} \\
= \; & \text{good approximation (training performance)} \\
+ \; & \text{good generalization (complexity penalty)}
\end{aligned}
$$

- a must-teach key message
- can be illustrated in different forms (e.g. VC bound, bias-variance, even human-learning philosophy)
- make learning non-trivial and fascinating to students

learnability, approximation & generalization —conducting machine learning properly [Cont.]

wrong use of learning (beginner's mistakes)

ensure good approximation, pray for good generalization
—praying for something out-of-control

right use of learning

ensure good generalization, try best for good approximation
—trying something possibly in-control

We cannot guarantee learning. We can "guarantee" no disasters. That is, after we learn we will either declare success or failure, and in both cases we will be right.

linear models
—conducting machine learning safely

linear models
$=$ good generalization
with established optimization tools for good approximation

- after knowing approximation/generalization:
  a good stage for learning safe techniques
- sufficiently useful for many practical problems (Yuan et al., 2012)
- building block in sophisticated techniques through feature transforms
- make learning concrete to students

linear models —conducting machine learning safely [Cont.]

wrong use of learning (beginner's mistakes)

start with the "greatest" techniques first — a point of no return

right use of learning

start with the simplest techniques first — and yes, it can work well

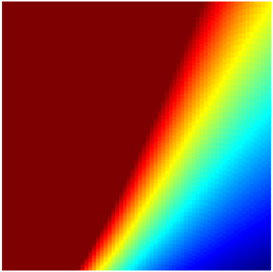
a rich and representative family of linear techniques

- classification: approx. combinatorial optimization (perceptron-like)
- regression: analytic optimization (pseudo-inverse)
- logistic regression: iterative optimization (SGD)

Students coming from diverse backgrounds not only get the big picture, but also the finer details in a concrete setting.

deal with noise and overfitting —conducting machine learning professionally

- overfit = difficult to ensure good generalization/learning with stochastic or deterministic noise on finite data
- regularization = tools for further guaranteeing good generalization
- validation = tools for certifying good learning



overfit(data size, noise level)

- turn amateur students to professionals
- make learning artistic to students

deal with noise and overfitting —conducting machine learning professionally [Cont.]

wrong use of learning (beginner's mistakes)

apply all possible techniques and choose by best approximation result —high risk of overfitting

right use of learning

apply a reasonable number of well-regularized techniques and choose by best validation result —relatively immune to noise and overfitting

Complex situations call for simpler models.

Teaching/Learning Life After the Foundations

Support Vector Machine

generalization	large-margin bound
approximation	quadratic programming
linear model	basic formulation
feature transform	through kernel
regularization	large-margin
validation	#-SV bound

Neural Network

#-neuron bound
gradient decent et al.
neurons
through cascading
weight-decay or early-stopping
for choices in regularization

```
[libsvm-2.9]$ ./svm-train -t 2 -g 0.05 -c 100 heart_scale
optimization finished, #iter = 1966
Total nSV = 113
```

- good approximation (by choosing kernel and optimization)
- good generalization (by regularization)
- good learning (by using #SV as validation indicator)

Teaching/Learning Life After the Foundations [Cont.]

- Caltech 2012: (mixed) 7 weeks of foundations, 0.5 week of NNet, 0.5 week of RBF Net, 1 week of SVM
- NTU 2011: (sequential) 10 weeks of foundations, 2.5 weeks of SVM, 2.5 weeks of bagging/boosting
  —with an in-class data mining competition[3] where students exploited taught/not-taught techniques with ease

often incremental efforts to teach/learn a new technique after solid foundations

[3] http://main.learner.csie.ntu.edu.tw/php/ml11fall/

Conclusion

foundation-based, foundation-**first**
—works well in our experience

- learnability: philosophical understanding, make learning non-trivial, conduct learning properly
- linear models: algorithmic modeling, make learning concrete, conduct learning safely
- overfitting: practical tuning, make learning artistic, conduct learning professionally

Thank you. Questions?