Want to learn more about ML

• there are a number of other relevant courses on campus (those shown in red are being offered in Spring 2016)
  • CS 761 *Advanced Machine Learning* (Profs. Nowak & Willett)
  • Statistics 479 *Statistical Machine Learning* (Prof. Balasubramanian)
  • BMI/CS 576 & 776 *Bioinformatics* (Prof. Gitter)
  • ECE 532 *Theory and Applications of Pattern Recognition*
  • ECE 539 *Intro to Artificial Neural Nets* (Prof. Hu)
  • ECE 830 *Estimation and Decision Theory* (Profs. Rob Nowak, Willett)
  • Statistics 840 *Statistical Model Building and Learning*
  • Statistics 860 *Estimation of Functions from Data*
  • Statistics 761 *Decision Trees for Multivariate Analysis*

Some Advice on Applying Machine Learning in Practice

Mark Craven
Computer Sciences 760
Fall 2015

www.biostat.wisc.edu/~craven/cs760/
Much of this lecture is based on:  
A Few Useful Things to Know about Machine Learning  
Pedro Domingos  
CACM 2012

It’s generalization that counts

“To generalize is to be an idiot”
-- William Blake

“Stick to poetry and painting, Blake”
-- Mark Craven
It’s generalization that counts

• the fundamental goal of machine learning is generalize beyond the instances in the training set
• you should rigorously measure generalization
• use a completely held-aside test set
• or use cross validation

It’s generalization that counts

• but be careful not to let any information from test sets leak into training

• be careful about overfitting a data set, even when using cross validation
It's generalization that counts

- compare multiple learning approaches
- there is no single best approach

Data alone is not enough

- learning algorithms require inductive biases
  - smoothness
  - similar instances having similar classes
  - limited dependencies
  - limited complexity
Data alone is not enough

• when choosing a representation, consider what kinds of background knowledge are easily expressed in it
  • what makes instances similar → kernels
  • dependencies → graphical models
  • logical rules → inductive logic programming
  • etc.

The importance of representation

• each domino covers two squares
• can you cover the board with dominoes?

• the solution is more apparent when we change the representation
Feature engineering is key

• typically the most important factor in a learning task is the feature representation
• many independent features that correlate with class → learning is easy
• class is a complex function of features → learning is hard
• try to craft features that make apparent what might be most important for the task

Learn many models, not just one

• winning team and runner-up were both formed by merging multiple teams
• winning systems were ensembles with > 100 models
• combination of the the two winning systems was even more accurate
Learn many models, not just one

- the lesson is more general than the Netflix prize
- ensembles very often improve the accuracy of individual models

We may care more about the model than actually making predictions

- two principal reasons for using machine learning
  1. to make predictions about test instances
  2. to gain insight into the problem domain
- for the former, a complicated black box may be okay
- for the latter, we want our models to be comprehensible to some degree
We may care more about the model than actually making predictions

- example: inferring Bayesian networks to represent intracellular networks [Sachs et al., *Science* 2005]

In many cases, we care about both

- example: predicting post-hospitalization VTE risk given patient histories [Kawaler et al., *AMIA* 2012]
  - want to identify patients at risk with high accuracy
  - want to identify previously unrecognized risk factors

<table>
<thead>
<tr>
<th>Category</th>
<th>Risk Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low Blood Volume</td>
<td>Furosemide</td>
</tr>
<tr>
<td></td>
<td>Hypovolemia</td>
</tr>
<tr>
<td></td>
<td>Hypo-osmolarity</td>
</tr>
<tr>
<td></td>
<td>Postschemorrhagic Anemia</td>
</tr>
<tr>
<td></td>
<td>Acute Renal Failure</td>
</tr>
<tr>
<td>Infection</td>
<td>E.Coli Infection</td>
</tr>
<tr>
<td></td>
<td>Levofoxacin</td>
</tr>
<tr>
<td></td>
<td>Cephalexin</td>
</tr>
<tr>
<td>Inflammation</td>
<td>High Alpha-1 Globulin Count</td>
</tr>
<tr>
<td></td>
<td>Angina Pectoris</td>
</tr>
<tr>
<td>Immobilization</td>
<td>Pathologic Fracture of Vertebræ</td>
</tr>
<tr>
<td>Malnutrition</td>
<td>Protein Celenis Malnutrition</td>
</tr>
</tbody>
</table>
Theoretical guarantees are not what they seem

- PAC bounds are extremely loose
- Asymptotic results tell us what happens when given infinite amounts of data – we don’t usually have this
- Learning theory results are generally
  - Useful for understanding learning, driving algorithm design
  - Not a criterion for practical decisions

Do assumptions of algorithm hold?

- Be sure to check the assumptions made by an approach/methodology against your problem domain
  - Are the instances \textit{i.i.d.} or should we take into account dependencies among them?
  - When we divide a data set into training/test sets, is the division representative of how the learner will be used in practice?
  - Etc.

- Questioning the assumptions of standard approaches sometimes results in new paradigms
  - Active learning
  - Multiple-instance learning
  - Etc.
Compare against reasonable baselines

- Empirically determine whether fancy ML methods have value by comparing against
  - simple predictors (e.g. tomorrow’s weather will be the same as today’s)
  - standard predictors in use
  - individual features