Course: Analytics, Machine Learning, and the Digital Economy

Tree-based methods and ensemble learning

Lecturer Radjabova Dilnora

What is Learning?

- Herbert Simon: “Learning is any process by which a system improves performance from experience.”

What is Learning?

- “A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.”

  – Tom Mitchell

Learning

- Learning is essential for unknown environments,
  - i.e., when designer lacks omniscience
- Learning is useful as a system construction method,
  - i.e., expose the agent to reality rather than trying to write it down
- Learning modifies the agent's decision mechanisms to improve performance

Machine Learning

- Machine learning: how to acquire a model on the basis of data / experience
  - Learning parameters (e.g. probabilities)
  - Learning structure (e.g. BN graphs)
  - Learning hidden concepts (e.g. clustering)

Machine Learning Areas

- **Supervised Learning:** Data and corresponding labels are given

Machine Learning Areas

- **Unsupervised Learning:** Only data is given, no labels provided

Machine Learning Areas

- **Semi-supervised Learning:** Some (if not all) labels are present

Machine Learning Areas

- **Reinforcement Learning**: An agent interacting with the world makes observations, takes actions, and is rewarded or punished; it should learn to choose actions in such a way as to obtain a lot of reward

Supervised Learning : Important Concepts

- Data: labeled instances <x_i, y>, e.g. emails marked spam/not spam
  - Training Set
  - Held-out Set
  - Test Set

Supervised Learning : Important Concepts

- Features: attribute-value pairs which characterize each *x*

Supervised Learning : Important Concepts

- Experimentation cycle
  - Learn parameters (e.g. model probabilities) on training set
  - (Tune hyper-parameters on held-out set)
  - Compute accuracy of test set
  - Very important: never “peek” at the test set!

Supervised Learning : Important Concepts

- Evaluation
  - Accuracy: fraction of instances predicted correctly

Supervised Learning : Important Concepts

- Overfitting and generalization
  - Want a classifier which does well on test data
  - Overfitting: fitting the training data very closely, but not generalizing well

Example: Spam Filter

Input: email

Output: spam/ham

Setup:

- Get a large collection of example emails, each labeled "spam" or "ham"
- Note: someone has to hand label all this data!
- Want to learn to predict labels of new, future emails

Features: The attributes used to make the ham / spam decision

- Words: FREE!
- Text Patterns: $dd, CAPS
- Non-text: SenderInContacts
- …

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virture of its nature as being utterly confidencial and top secret. …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99

Ok, Iknow this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.

Slide from Mackassy

Example: Digit Recognition

Input: images / pixel grids

Output: a digit 0-9

Setup:

- Get a large collection of example images, each labeled with a digit
- Note: someone has to hand label all this data!
- Want to learn to predict labels of new, future digit images

0

1

2

1

??

Features: The attributes used to make the digit decision

- Pixels: (6,8)=ON
- Shape Patterns: NumComponents, AspectRatio, NumLoops
- ...

Slide from Mackassy

Classification Examples

- In classification, we predict labels *y* (classes) for inputs *x*

- Examples:
  - OCR (input: images, classes: characters)
  - Medical diagnosis (input: symptoms, classes: diseases)
  - Automatic essay grader (input: document, classes: grades)
  - Fraud detection (input: account activity, classes: fraud / no fraud)
  - Customer service email routing
  - Recommended articles in a newspaper, recommended books
  - DNA and protein sequence identification
  - Categorization and identification of astronomical images
  - Financial investments
  - … many more

Inductive learning

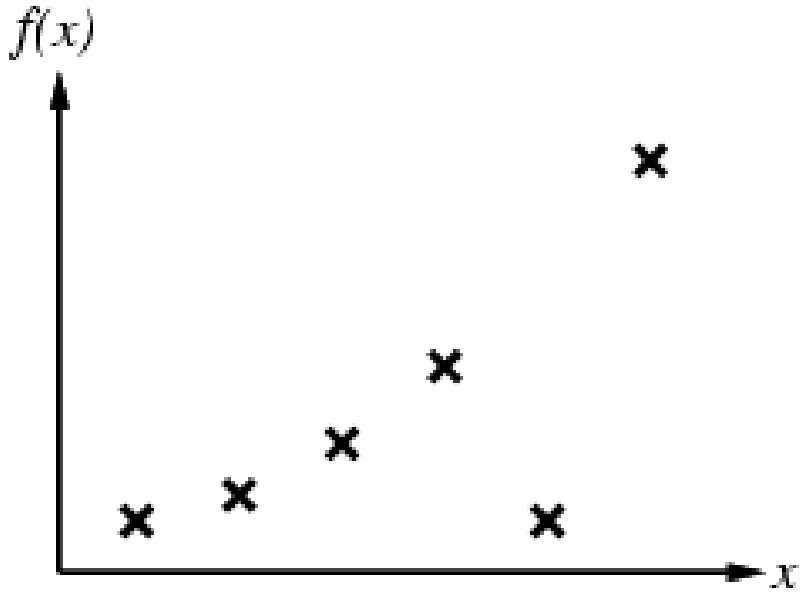
- Simplest form: learn a function from examples
-
- *f* is the target function
- An example is a pair (*x*, *f(x)*)
-

Inductive learning

- **Pure induction task:**
  - **Given a collection of examples of *f*, return a function *h* that approximates *f*.**
  - find a hypothesis *h*, such that $h \approx f$, given a training set of examples
  -
- (This is a highly simplified model of real learning:
  - Ignores prior knowledge
  - Assumes examples are given)
  -

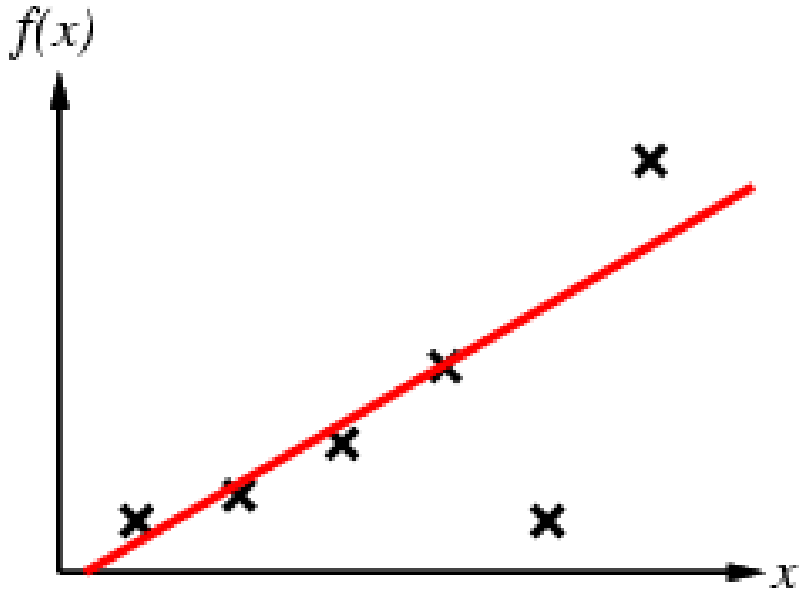
Inductive learning method

- Construct/adjust *h* to agree with *f* on training set
- (*h* is consistent if it agrees with *f* on all examples)
-
- E.g., curve fitting:
-



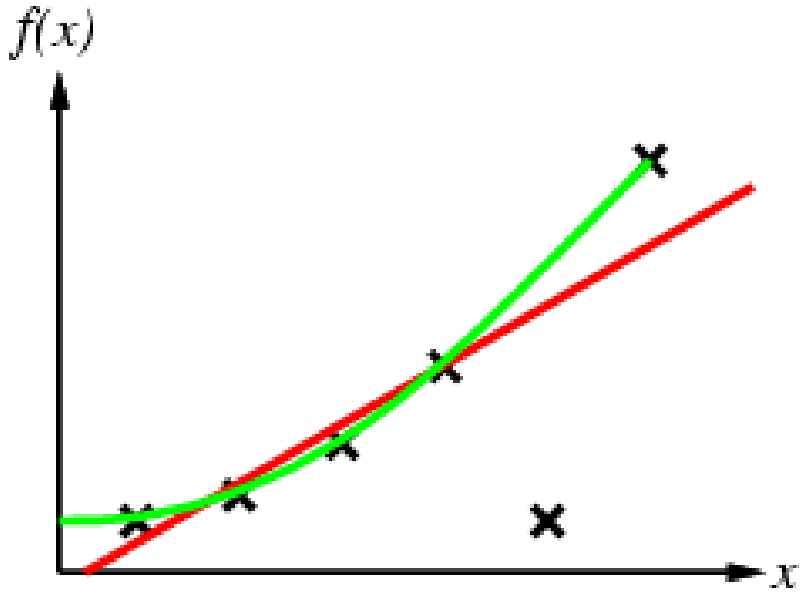
Inductive learning method

- Construct/adjust *h* to agree with *f* on training set
- (*h* is consistent if it agrees with *f* on all examples)
-
- E.g., curve fitting:
-



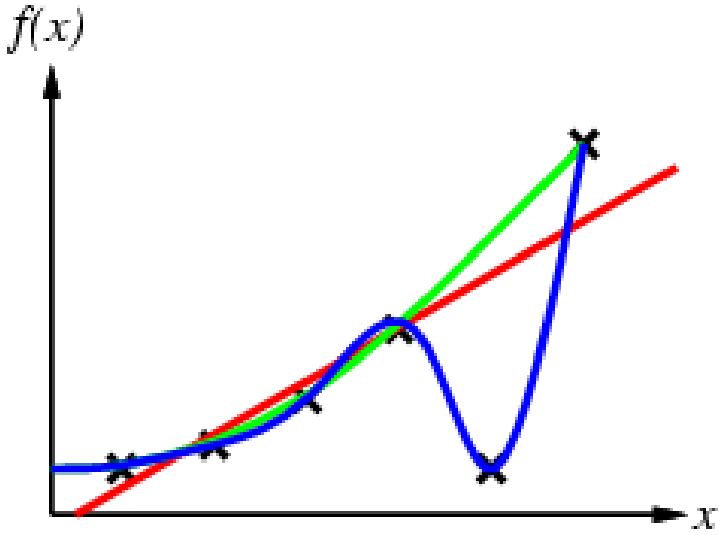
Inductive learning method

- Construct/adjust *h* to agree with *f* on training set
- (*h* is consistent if it agrees with *f* on all examples)
-
- E.g., curve fitting:
-



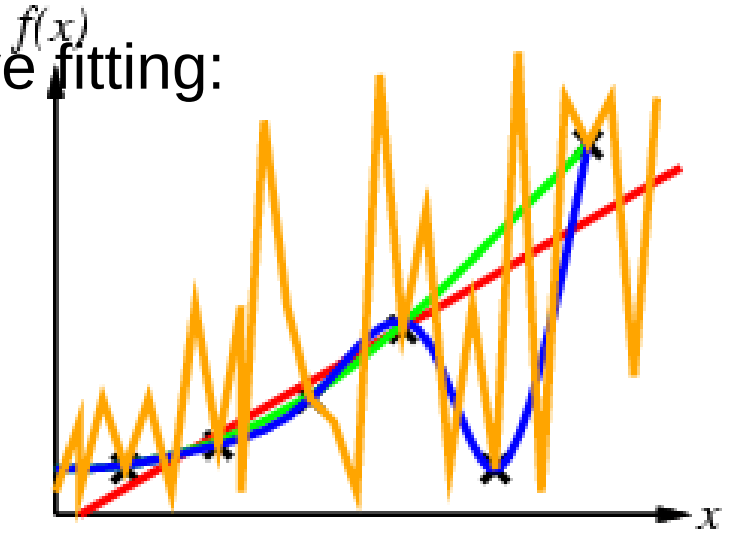
Inductive learning method

- Construct/adjust *h* to agree with *f* on training set
- (*h* is consistent if it agrees with *f* on all examples)
-
- E.g., curve fitting:
-



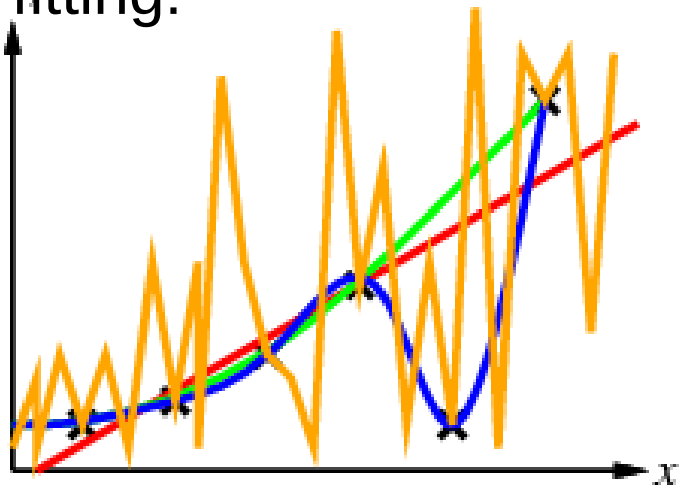
Inductive learning method

- Construct/adjust *h* to agree with *f* on training set
- (*h* is consistent if it agrees with *f* on all examples)
-
- E.g., curve fitting:



Inductive learning method

- Construct/adjust *h* to agree with *f* on training set
- (*h* is consistent if it agrees with *f* on all examples)
-
- E.g., curve fitting:



- Ockham's razor: prefer the simplest hypothesis consistent with data

Generalization

- Hypotheses must generalize to correctly classify instances not in the training data.

26

Generalization

- Simply memorizing training examples is a consistent hypothesis that does not generalize.

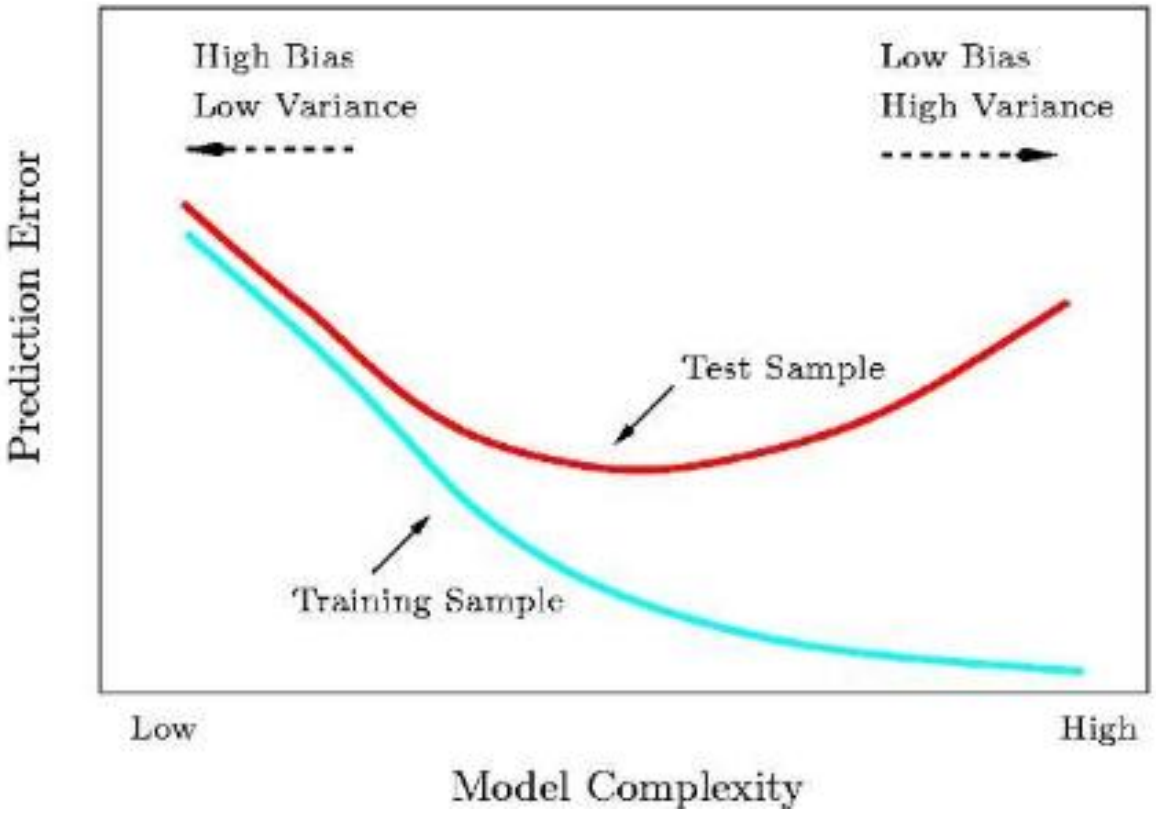
27

Generalization

- *Occam's razor*:
  - Finding a *simple* hypothesis helps ensure generalization.

28

Training Error vs Test Error



Reference

- 1. Shalev-Shwartz and Ben-David. Understanding Machine Learning: From Theory to Algorithms (Cambridge University Press, 2014)
- 2. Daum´e. A Course in Machine Learning.
- 3. The Art of Statistics: How to Learn from Data by David Shpigelter
- 4. Learning From Data – January 1, 2012 by Yaser S. Abu-Mostafa (Author), Malik Magdon-Ismail (Author), Hsuan-Tien Lin (Author)
- 5. Statistics: The Art and Science of Learning from Data by Alan Agresti
- 6. Learning From Data: An Introduction To Statistical Reasoning by M.Glenber.
- 7. Statistics: Learning from Data (with JMP Printed Access Card) by Rocky Pek
- 8. The Elements of Statistical Learning by Gerim Garold
- 9. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems 2nd Edition
- by Aurélien Géron (Author)