Course: Analytics, Machine Learning, and the Digital Economy

Machine Learning

Lecturer Dilnora Radjabova

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

3

Learning

- Learning is essential for unknown environments,
  - i.e., when designer lacks omniscience

Learning

- Learning is useful as a system construction method,
  - i.e., expose the agent to reality rather than trying to write it down

Learning

- 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
- **Unsupervised Learning:** Only data is given, no labels provided
- **Semi-supervised Learning:** Some (if not all) labels are present
- **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
- Features: attribute-value pairs which characterize each x
- 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!
- Evaluation
  - Accuracy: fraction of instances predicted correctly
- 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.

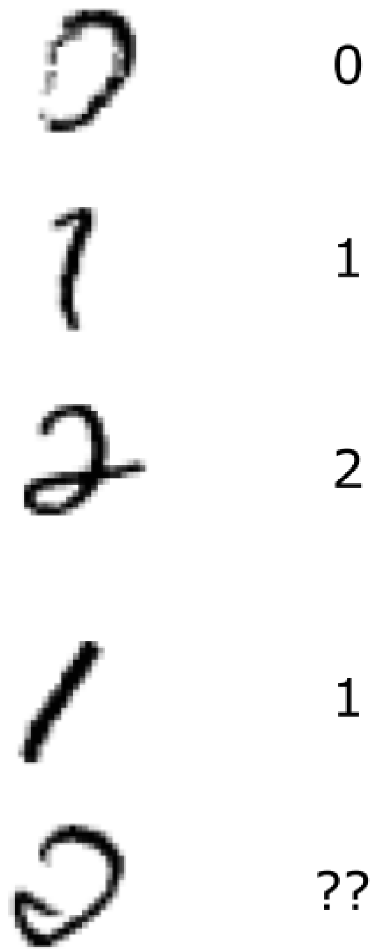
Example: Digit Recognition

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



Features: The attributes used to make the digit decision

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

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

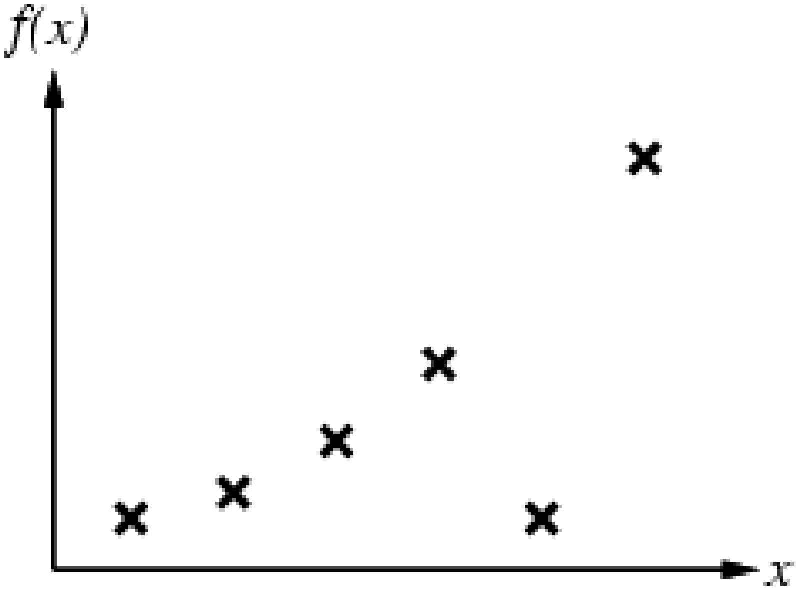
- Simplest form: learn a function from examples
-
- *f* is the target function
- An example is a pair (*x*, *f(x)*)
-

- **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

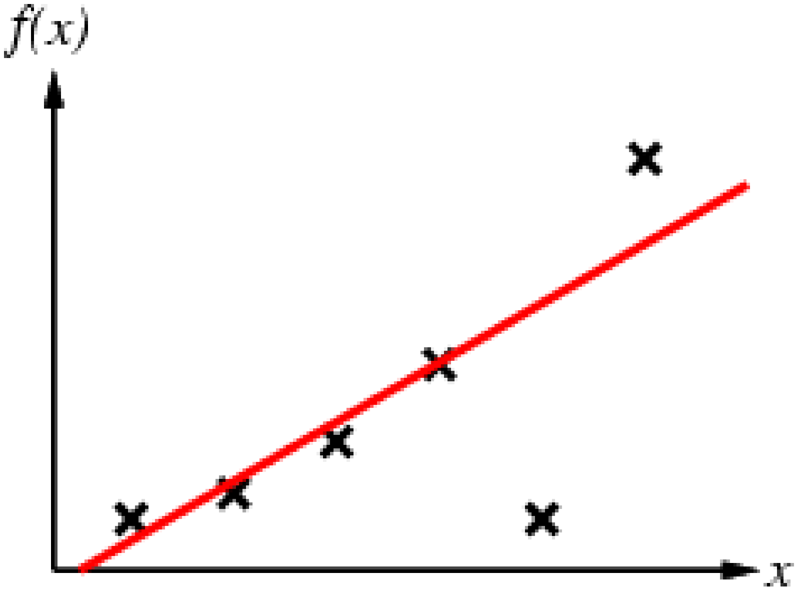
Inductive learning method

- 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:
-



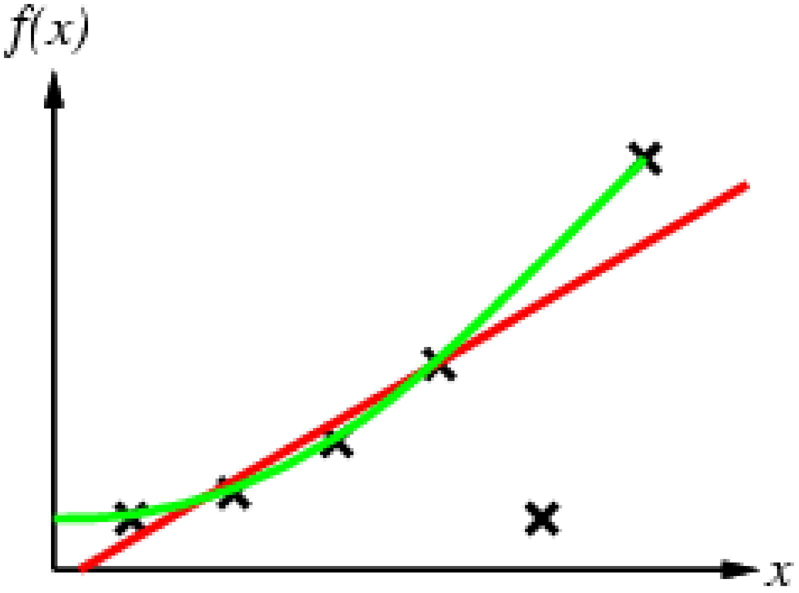
Inductive learning method

- 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:
-



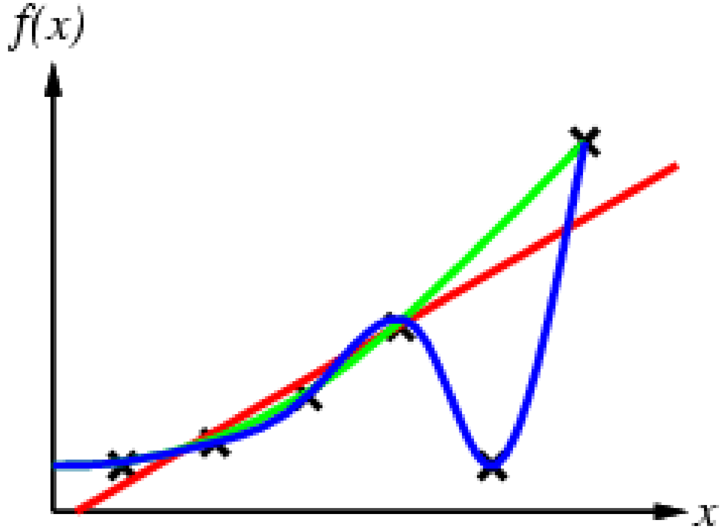
Inductive learning method

- 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:
-



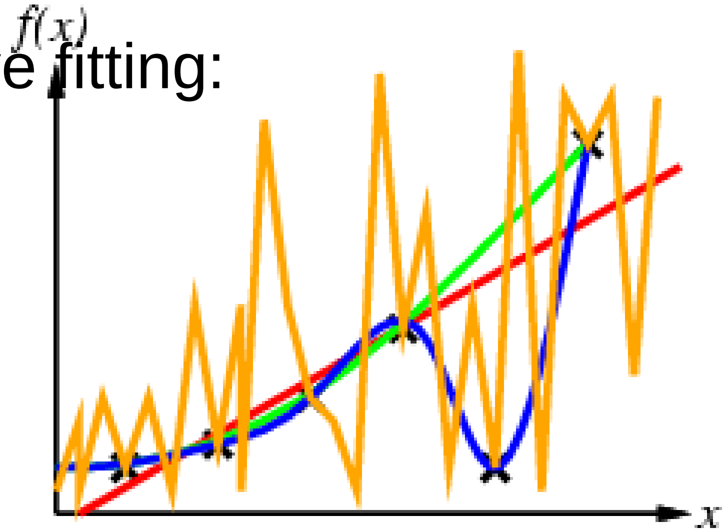
Inductive learning method

- 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:
-



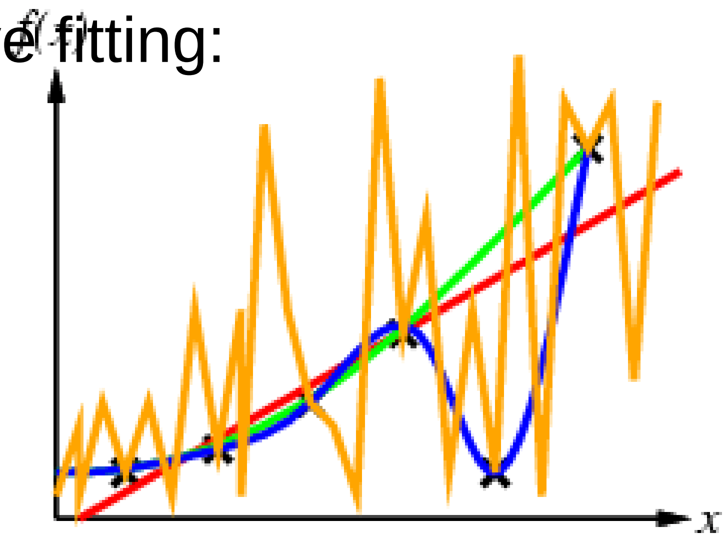
Inductive learning method

- 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:



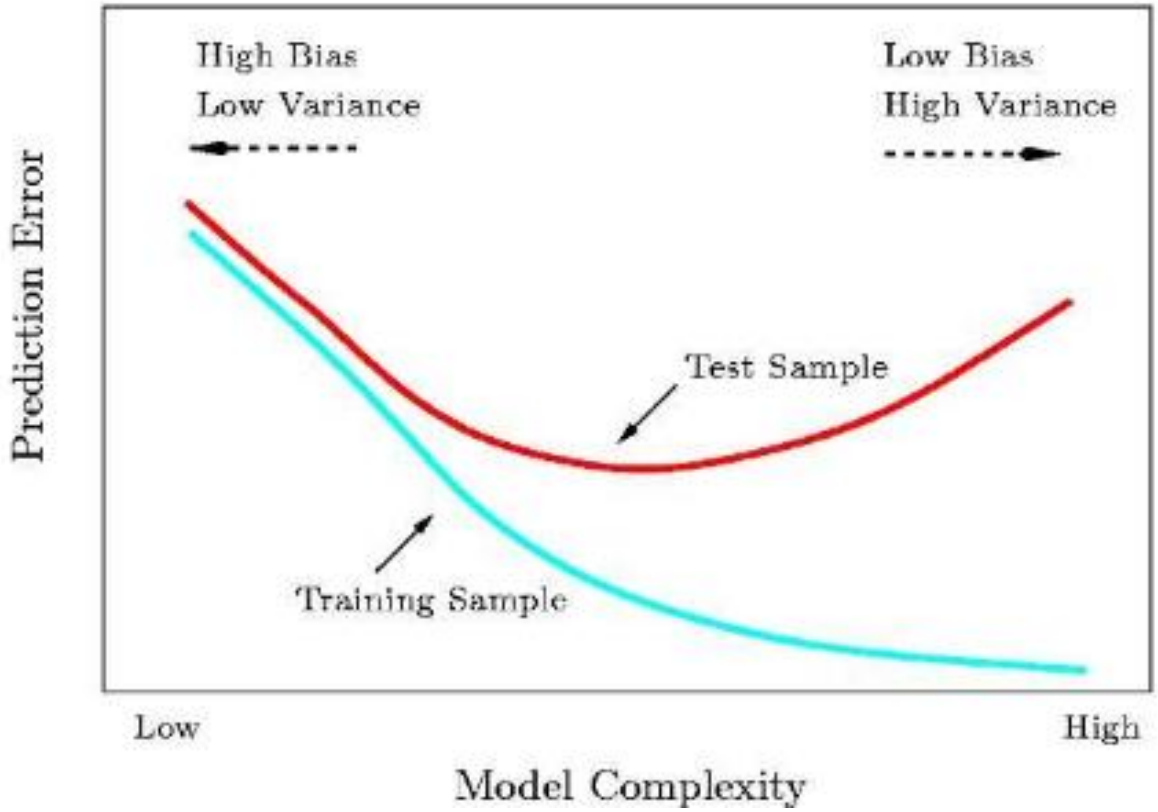
- Ockham's razor: prefer the simplest hypothesis consistent with data

Generalization

- Hypotheses must generalize to correctly classify instances not in the training data.
- Simply memorizing training examples is a consistent hypothesis that does not generalize.
- *Occam's razor*:
  - Finding a *simple* hypothesis helps ensure generalization.

20

Training Error vs Test Error



Supervised Learning

- Learning a discrete function: **Classification**
  - Boolean classification:
    - Each example is classified as true(positive) or false(negative).
- Learning a continuous function: **Regression**

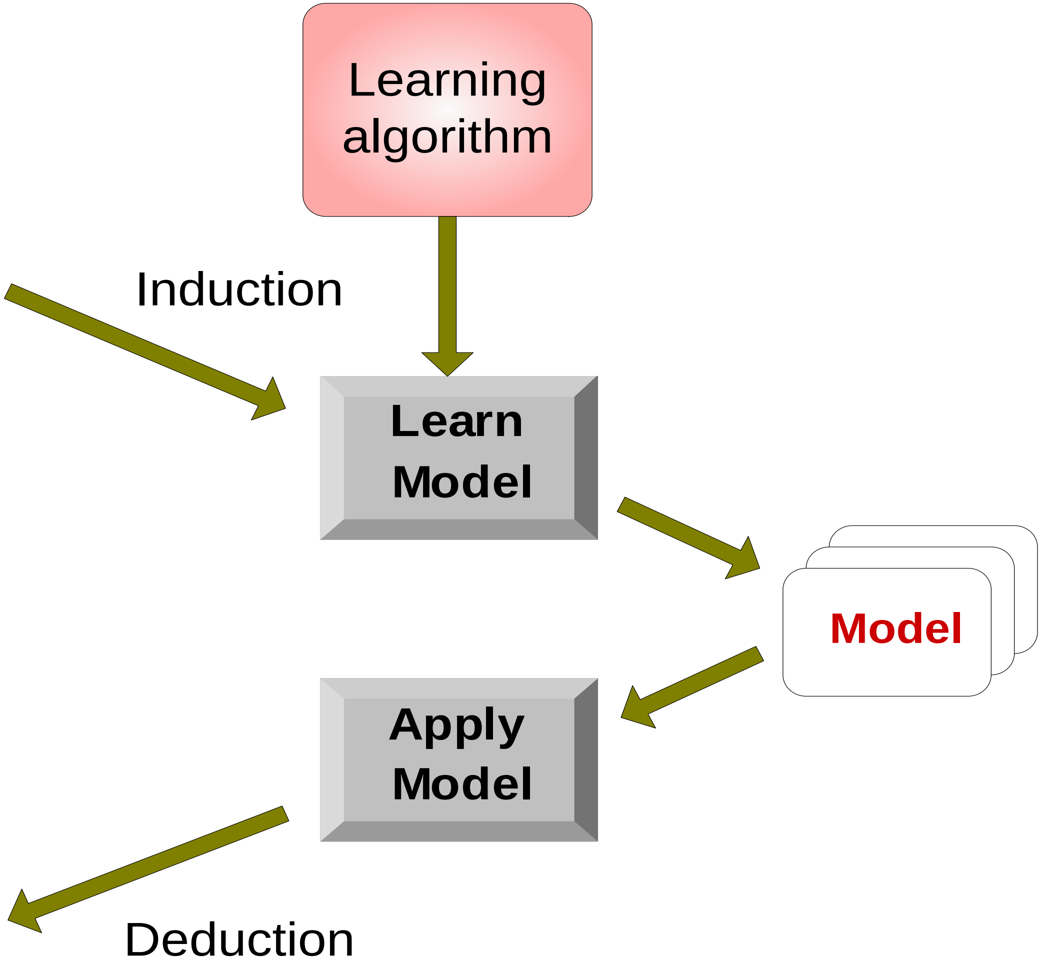
Classification—A Two-Step Process

- Model construction: describing a set of predetermined classes
  - Each tuple/sample is assumed to belong to a predefined class, as determined by the class label
  - The set of tuples used for model construction is training set
  - The model is represented as **classification rules**, **decision trees**, or **mathematical formulae**
- Model usage: for classifying future or unknown objects
  - Estimate accuracy of the model
    - The known label of test sample is compared with the classified result from the model
    - **Test set is independent of training set**, otherwise over-fitting will occur
  - If the accuracy is acceptable, use the model to classify data tuples whose class labels are not known

Illustrating Classification Task

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

Issues: Data Preparation

- Data cleaning
  - Preprocess data in order to reduce noise and handle missing values
- Relevance analysis (feature selection)
  - Remove the irrelevant or redundant attributes
- Data transformation
  - Generalize data to (higher concepts, discretization)
  - Normalize attribute values

Classification Techniques

- Decision Tree based Methods
- Rule-based Methods
- Naïve Bayes and Bayesian Belief Networks
- Neural Networks
- Support Vector Machines
- and more...

Learning decision trees

Example Problem: decide whether to wait for a table at a restaurant, based on the following attributes:

1. **Alternate**: is there an alternative restaurant nearby?
2. **Bar**: is there a comfortable bar area to wait in?
3. **Fri/Sat**: is today Friday or Saturday?
4. **Hungry**: are we hungry?
5. **Patrons**: number of people in the restaurant (None, Some, Full)
6. **Price**: price range ($, $$, $$$)
7. **Raining**: is it raining outside?
8. **Reservation**: have we made a reservation?
9. **Type**: kind of restaurant (French, Italian, Thai, Burger)
10. **WaitEstimate**: estimated waiting time (0-10, 10-30, 30-60, >60)

Feature(Attribute)-based representations

- Examples described by feature(attribute) values
  - (Boolean, discrete, continuous)
- E.g., situations where I will/won't wait for a table:

Example	Attributes										Target
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*Wait*
X_1	T	F	F	T	Some	$$$	F	T	French	0–10	T
X_2	T	F	F	T	Full	$	F	F	Thai	30–60	F
X_3	F	T	F	F	Some	$	F	F	Burger	0–10	T
X_4	T	F	T	T	Full	$	F	F	Thai	10–30	T
X_5	T	F	T	F	Full	$$$	F	T	French	>60	F
X_6	F	T	F	T	Some	$$	T	T	Italian	0–10	T
X_7	F	T	F	F	None	$	T	F	Burger	0–10	F
X_8	F	F	F	T	Some	$$	T	T	Thai	0–10	T
X_9	F	T	T	F	Full	$	T	F	Burger	>60	F
X_{10}	T	T	T	T	Full	$$$	F	T	Italian	10–30	F
X_{11}	F	F	F	F	None	$	F	F	Thai	0–10	F
X_{12}	T	T	T	T	Full	$	F	F	Burger	30–60	T

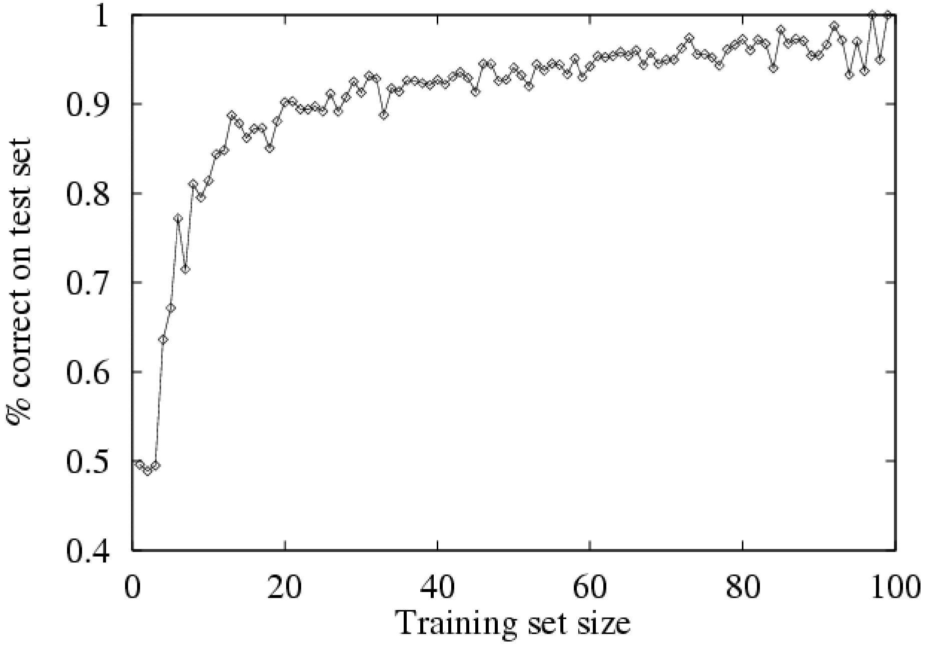
- Classification of examples is positive (T) or negative (F)
-

Performance measurement

- How do we know that $h \approx f$?
  1. Use theorems of computational/statistical learning theory
  2. Try h on a new test set of examples

     (use same distribution over example space as training set)

Learning curve = % correct on test set as a function of training set size



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)