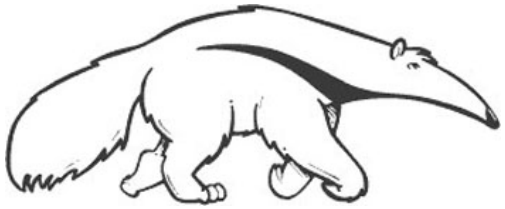
+

Machine Learning and Data Mining

Introduction

Prof. Alexander Ihler



Artificial Intelligence (AI)

- Building “intelligent systems”
- Lots of parts to intelligent behavior

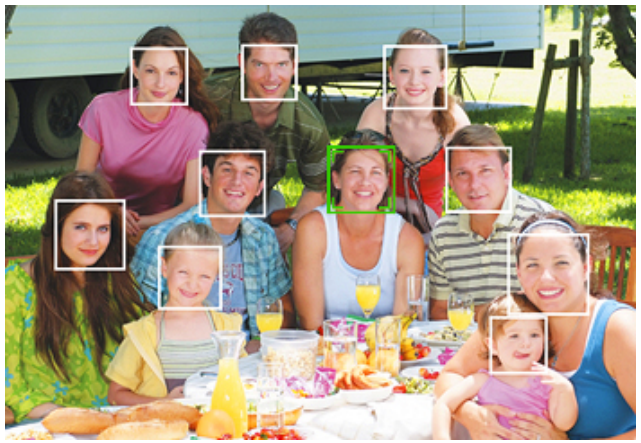
Darpa GC (Stanley)

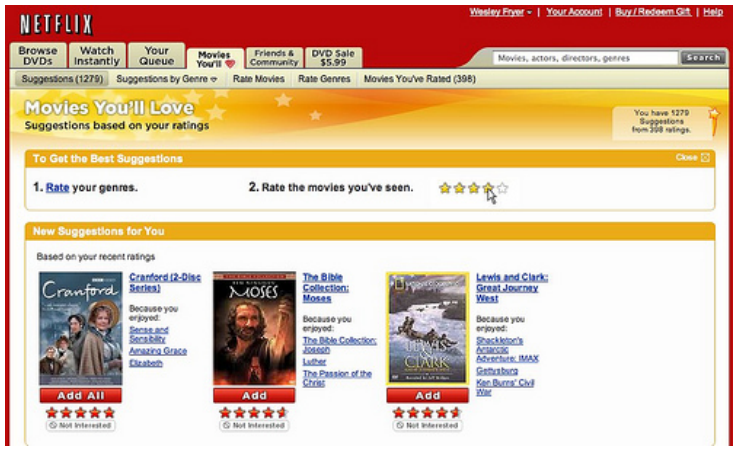
RoboCup

Chess (Deep Blue v. Kasparov)

Machine learning (ML)

- One (important) part of AI
- Making predictions (or decisions)
- Getting better with experience (data)
- Problems whose solutions are “hard to describe”

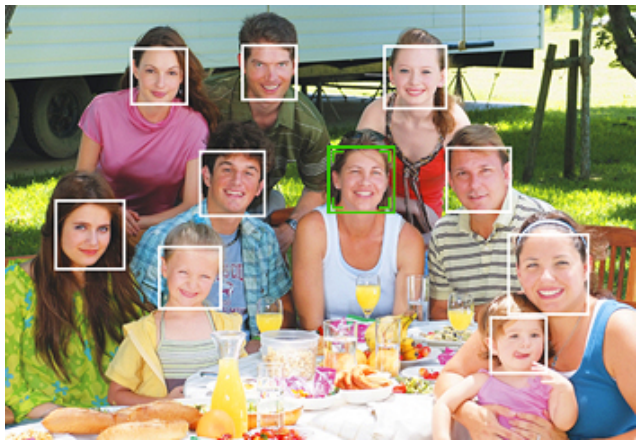


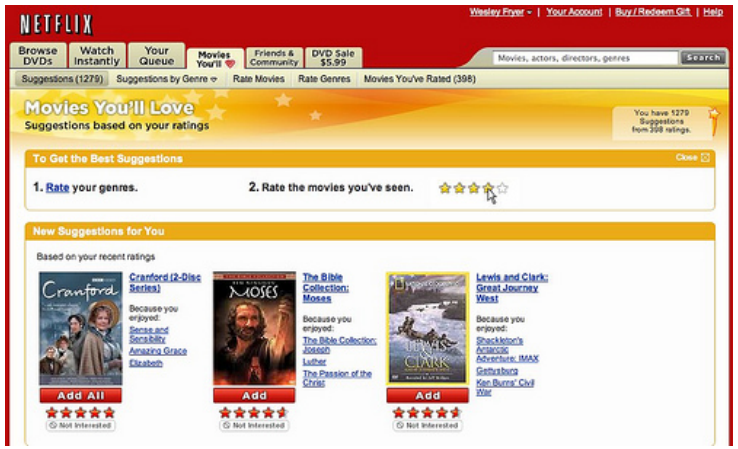


Types of prediction problems

- Supervised learning
  - "Labeled" training data
  - Every example has a desired target value (a "best answer")
  - Reward prediction being close to target

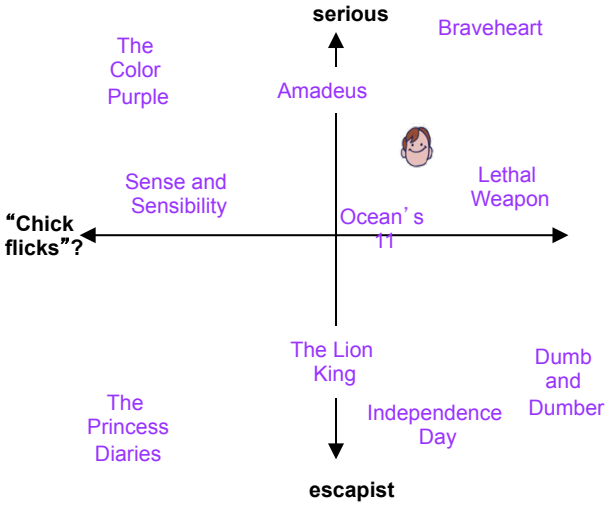
  - Classification: a discrete-valued prediction
  - Regression: a continuous-valued prediction

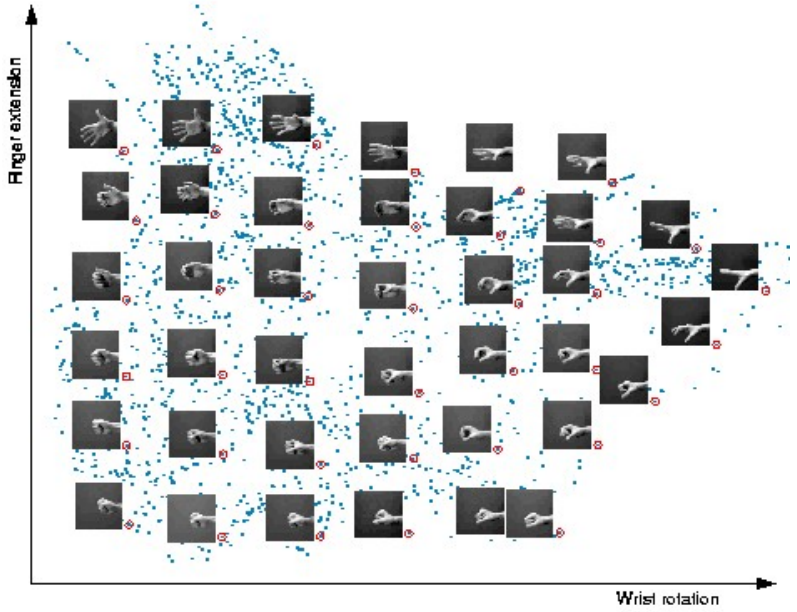




Types of prediction problems

- Supervised learning
- Unsupervised learning
  - No known target values
  - No targets = nothing to predict?
  - Reward “patterns” or “explaining features”
  - Often, data mining





Types of prediction problems

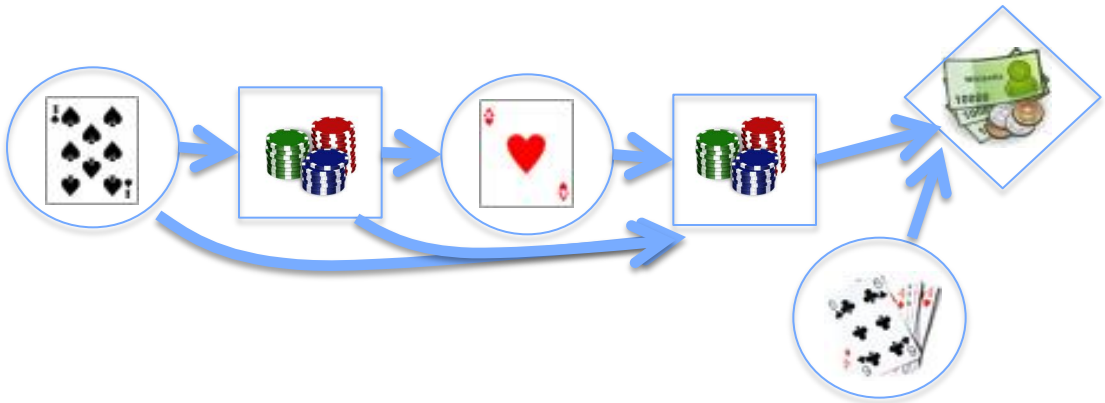
- Supervised learning
- Unsupervised learning
- Semi-supervised learning
  - Similar to supervised
  - some data have unknown target values

- Ex: medical data
  - Lots of patient data, few known outcomes
- Ex: image tagging
  - Lots of images on Flikr, but only some of them tagged

Types of prediction problems

- Supervised learning
- Unsupervised learning
- Semi-supervised learning
- Reinforcement learning

- "Indirect" feedback on quality
  - No answers, just "better" or "worse"
  - Feedback may be delayed



Logistics

- Course webpage for assignments & other info
- EEE for homework submission & return
  - Emails: will send a test email tomorrow – make sure you get it
- Piazza for questions & discussions

- No required textbook
  - Recommended: Murphy, “Machine Learning...”, 2012.
  - Also
    - Duda, Hart & Stork, “Pattern classification”
    - Hastie, Tibshirani & Friedman, “Elements of Statistical Learning”
- But
  - I’ll try to cover everything needed in lectures and notes
  - All textbooks mainly for reference purposes

Logistics

- Grading (approximate)
  - 25% homework (~6, drop lowest)
  - 10% project (Kaggle)
  - 5% reading quizzes
  - 25% midterm, 35% final
  - Due 5pm listed day, EEE or my office
  - No late homework (solutions posted)
    - Turn in what you have

- Collaboration
  - Study groups, discussion, assistance encouraged
    - Whiteboards, etc.
  - Do your homework yourself
    - Don't exchange solutions or HW code

Data exploration

- Machine learning is a data science
  - Look at the data; get a “feel” for what might work
- What types of data do we have?
  - Binary values? (spam; gender; …)
  - Categories? (home state; labels; …)
  - Integer values? (1..5 stars; age brackets; …)
  - (nearly) real values? (pixel intensity; prices; …)
- Are there missing data?
- “Shape” of the data? Outliers?

Matlab and alternatives

- Matlab: interpreted language for scientific computing
  - Originally designed for linear algebra (matrices, vectors)
  - Heavily adopted in research; lots of available code
  - Can be inefficient and slow

Alternatives

- Octave: free near-equivalent to Matlab
  - Often less optimized & lags Matlab features, but almost code-equivalent
- FreeMat: another free Matlab alternative

- R: Free, heavily adopted in statistics, less so in computer science
- Python: (SciPy, MatPlotLib): some adoption, esp. in comp. bio.
- C++: Fast & efficient but slow to prototype

Representing the data (Matlab)

- Have m observations (data points)

$$\left\{x^{(1)} \ldots, x^{(m)}\right\}$$

- Each observation is a vector consisting of n features

$$x^{(j)} = [x_1^{(j)} x_2^{(j)} \ldots x_n^{(j)}]$$

- Often, represent this as a "data matrix"

$$\underline{X} = \begin{bmatrix} x_0^{(1)} & \ldots & x_n^{(1)} \\ \vdots & \ddots & \vdots \\ x_0^{(m)} & \ldots & x_n^{(m)} \end{bmatrix}$$

```
>> load('fisheriris');  % Load Fisher's "Iris" dataset
>> X = meas;            % Rename measurements as "X"
>> size(X),
ans =
   150     4             % 150 data points; 4 features each
```

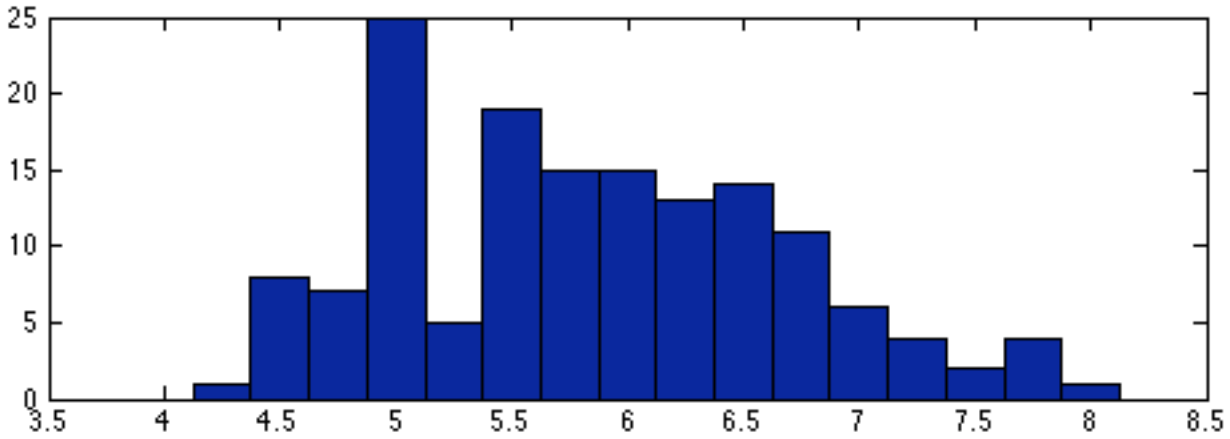
Basic statistics

- Look at basic information about features
  - Average value? (mean, median, etc.)
  - "Spread"? (standard deviation, etc.)
  - Maximum / Minimum values?

```
>>  mean(X)        % compute mean of each feature
ans =
   5.8433    3.0573    3.7580    1.1993
>>  std(X)         % compute standard deviation of each feature
ans =
   0.8281    0.4359    1.7653    0.7622
>> max(X)         % largest value per feature
ans =
   7.9411    4.3632    6.8606    2.5236
>> min(X)         % smallest value per feature
ans =
   4.2985    1.9708    1.0331    0.0536
```

Histograms

- Count the data falling in each of K bins
  - "Summarize" data as a length-K vector of counts (& plot)
  - Value of K determines "summarization"; depends on # of data
    - K too big: every data point falls in its own bin; just "memorizes"
    - K too small: all data in one or two bins; oversimplifies



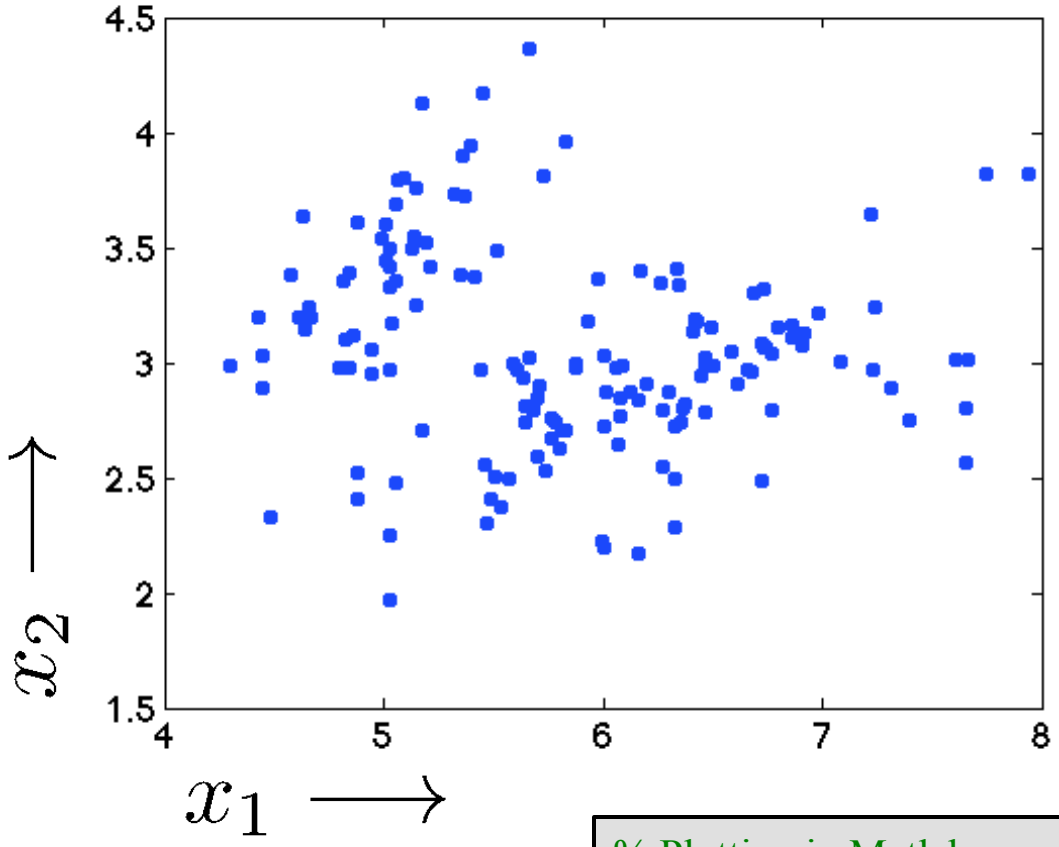

```
% Histograms in Matlab
>>  x1 = X(:,1);          % Histogram for feature #1
>>  Bins = 4:0.25:8;      % Use explicit bin locations
>>  hist(x1,Bins);        % Compute & plot histogram
```

Scatterplots

- Illustrate the relationship between two features

ɔts”

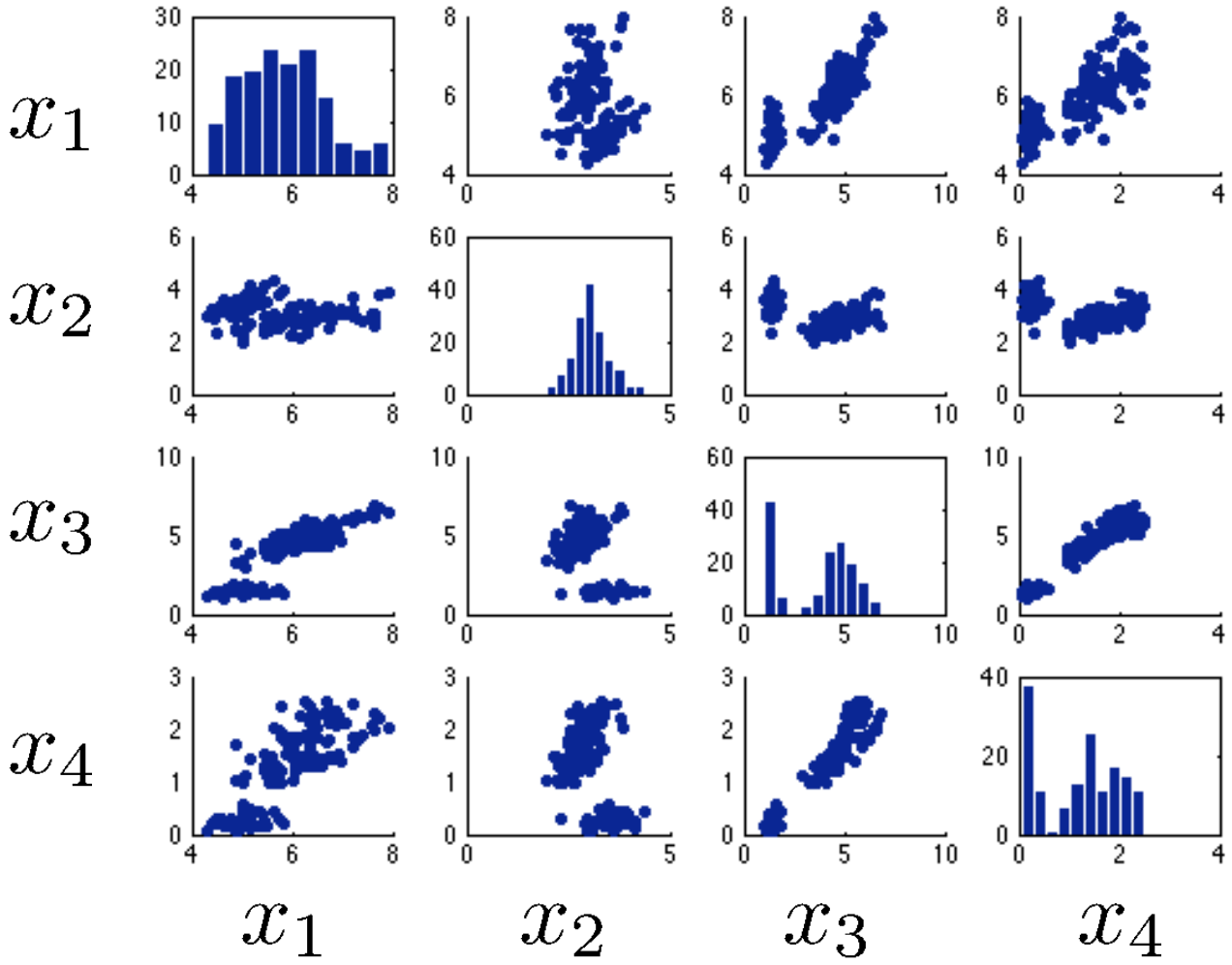
le that too




```
% Plotting in Matlab
>>  plot(X(:,1), X(:,2), 'b.');   % plot data points as blue dots
```

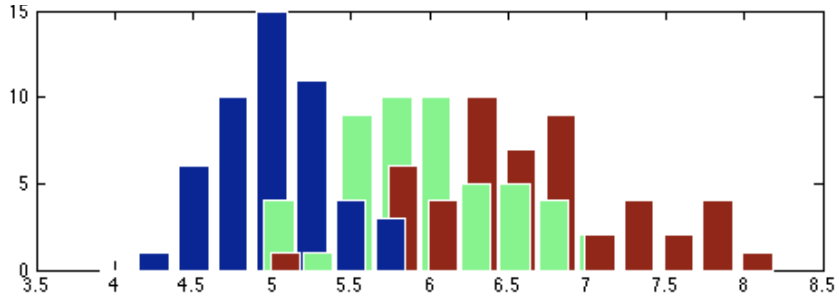
Scatterplots

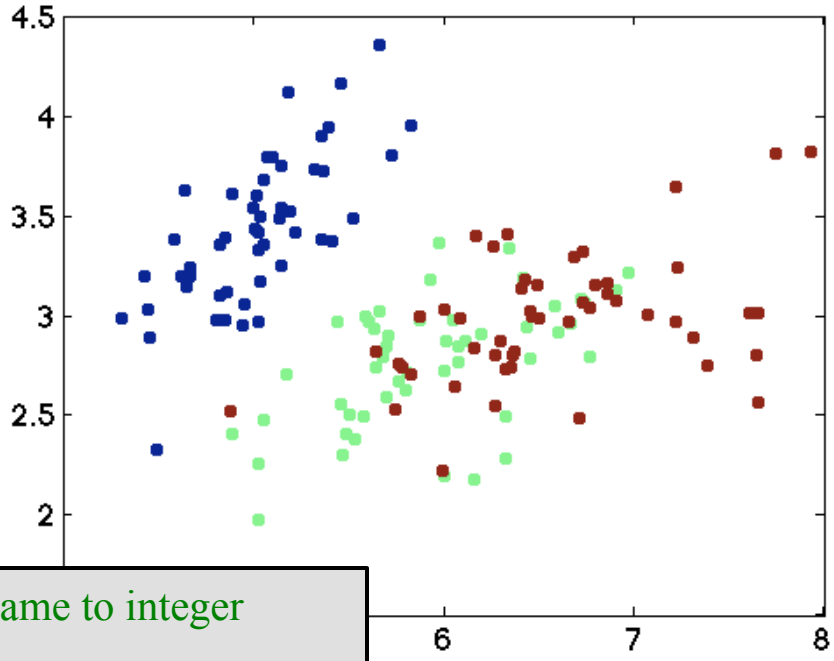
- For more than two features can use a pair plot



Supervised learning and targets

- Supervised learning: predict target values
- For discrete targets, often visualize with color



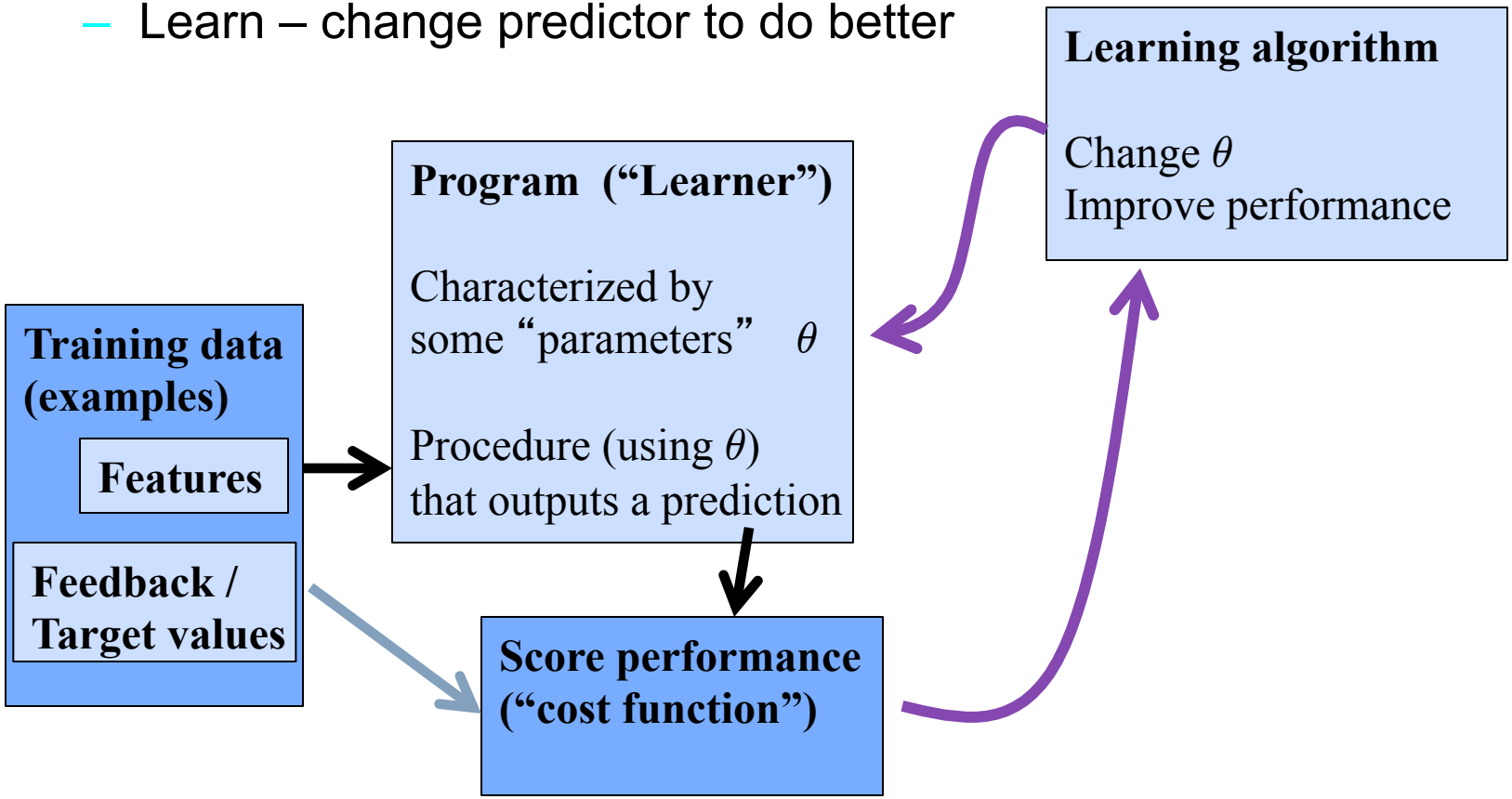



```
>> Y = zeros(size(species));        % Convert species name to integer
>> names = unique(species);
>> for i=1:length(names), Y(strcmp(species,names{i})) = i; end;

>> histy(X(:,1),Y);                 % Colored histogram (not built-in)
>> scatter(X(:,1),X(:,2),[],Y,'filled'); % Colored scatterplot
```

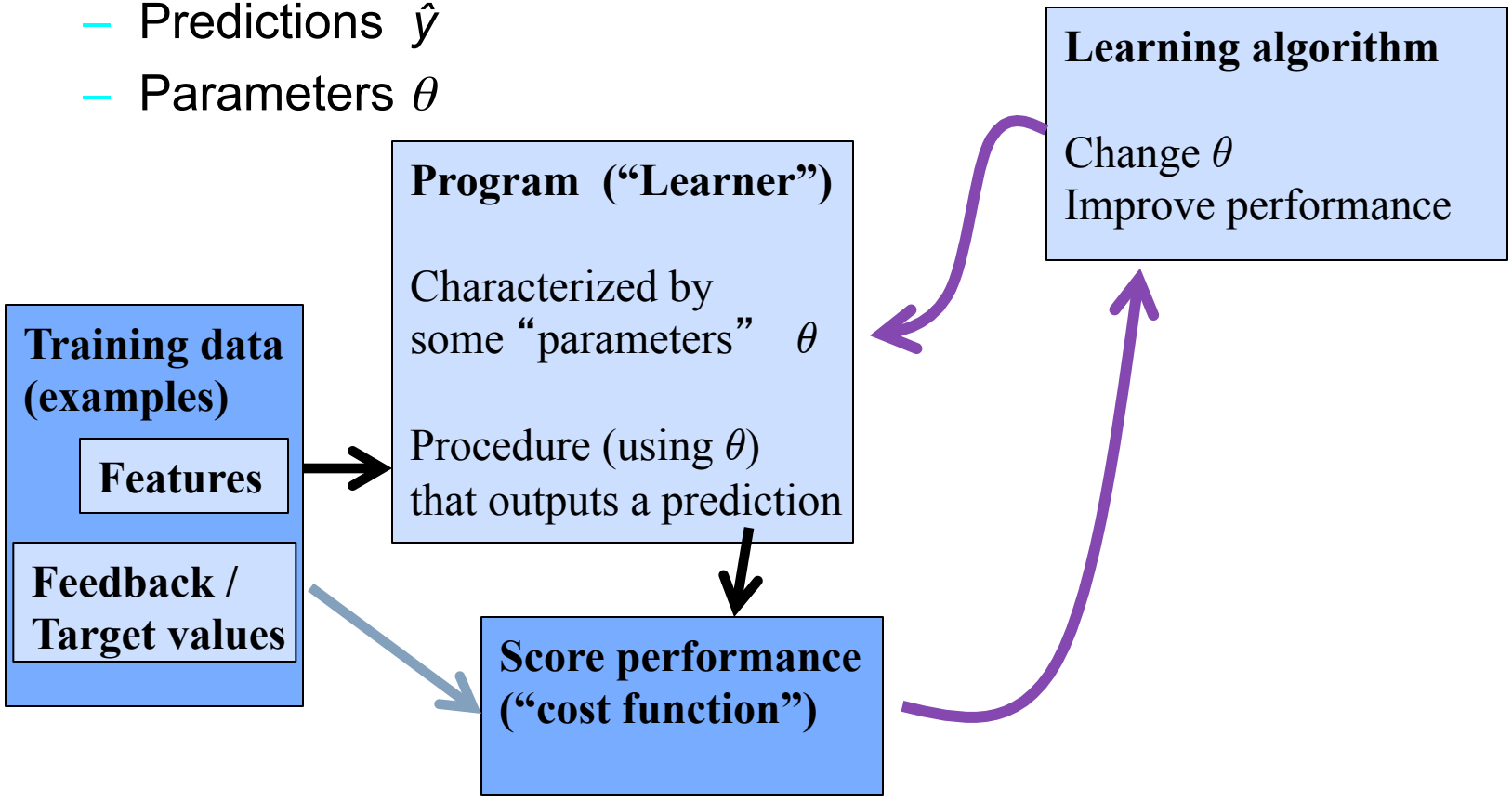
How does machine learning work?

- "Meta-programming"
  - Predict – apply rules to examples
  - Score – get feedback on performance
  - Learn – change predictor to do better

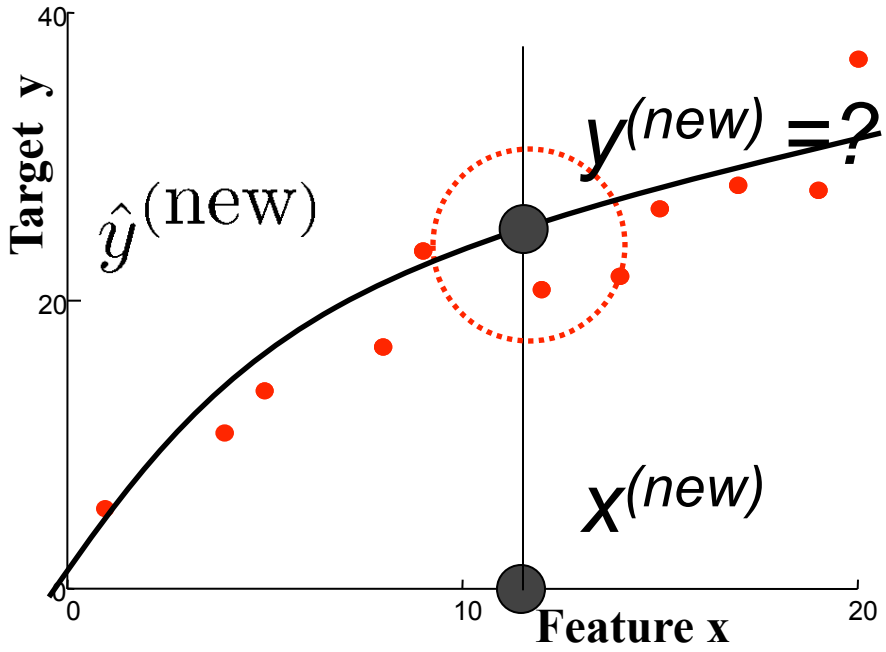


Supervised learning

- Notation
  - Features x
  - Targets y
  - Predictions $\hat{y}$
  - Parameters θ

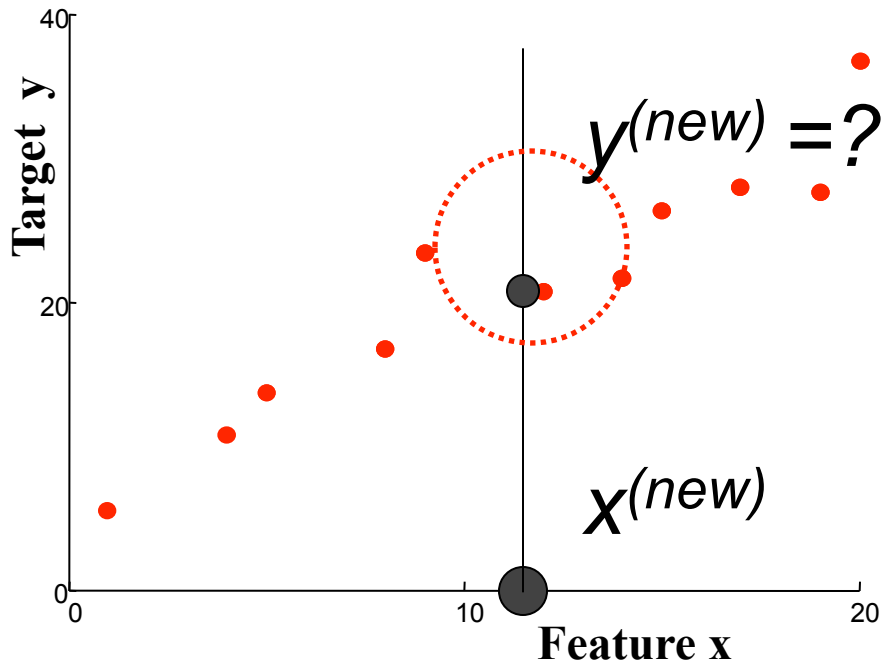


Regression; Scatter plots



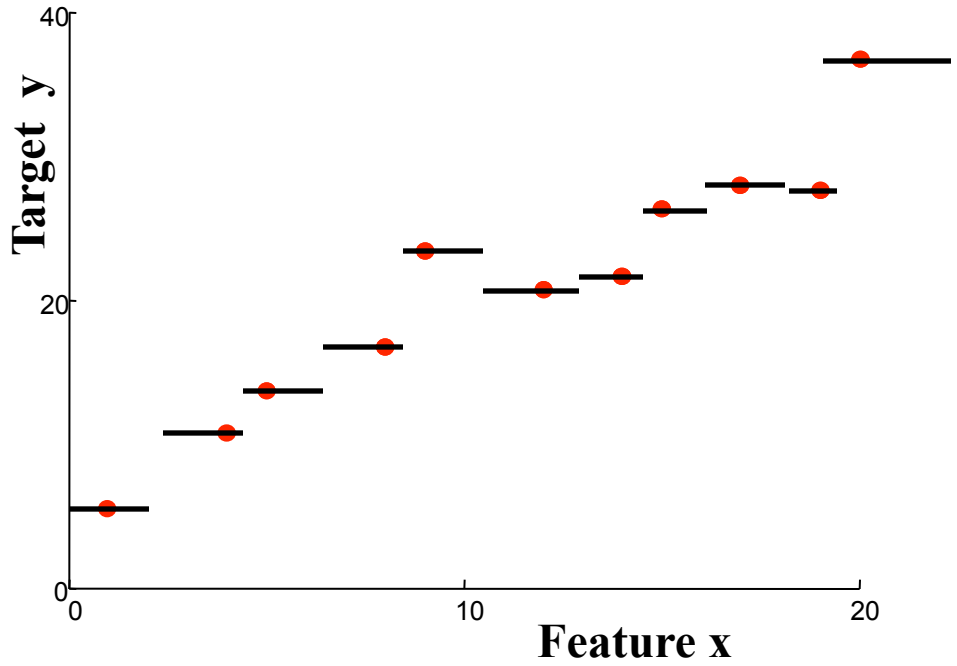
- Suggests a relationship between x and y
- *Prediction*: new x, what is y?

Nearest neighbor regression



- Find training datum $x^{(i)}$ closest to $x^{(new)}$
  Predict $y^{(i)}$

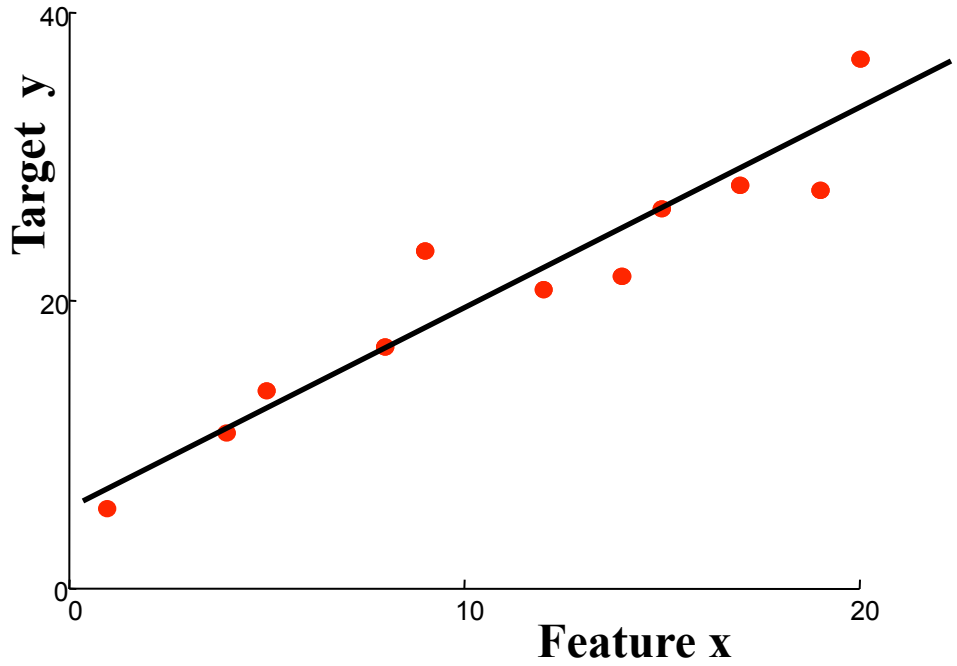
Nearest neighbor regression



"Predictor":
Given new features:
Find nearest example
Return its value

- Defines a function f(x) implicitly
- "Form" is piecewise constant

Linear regression



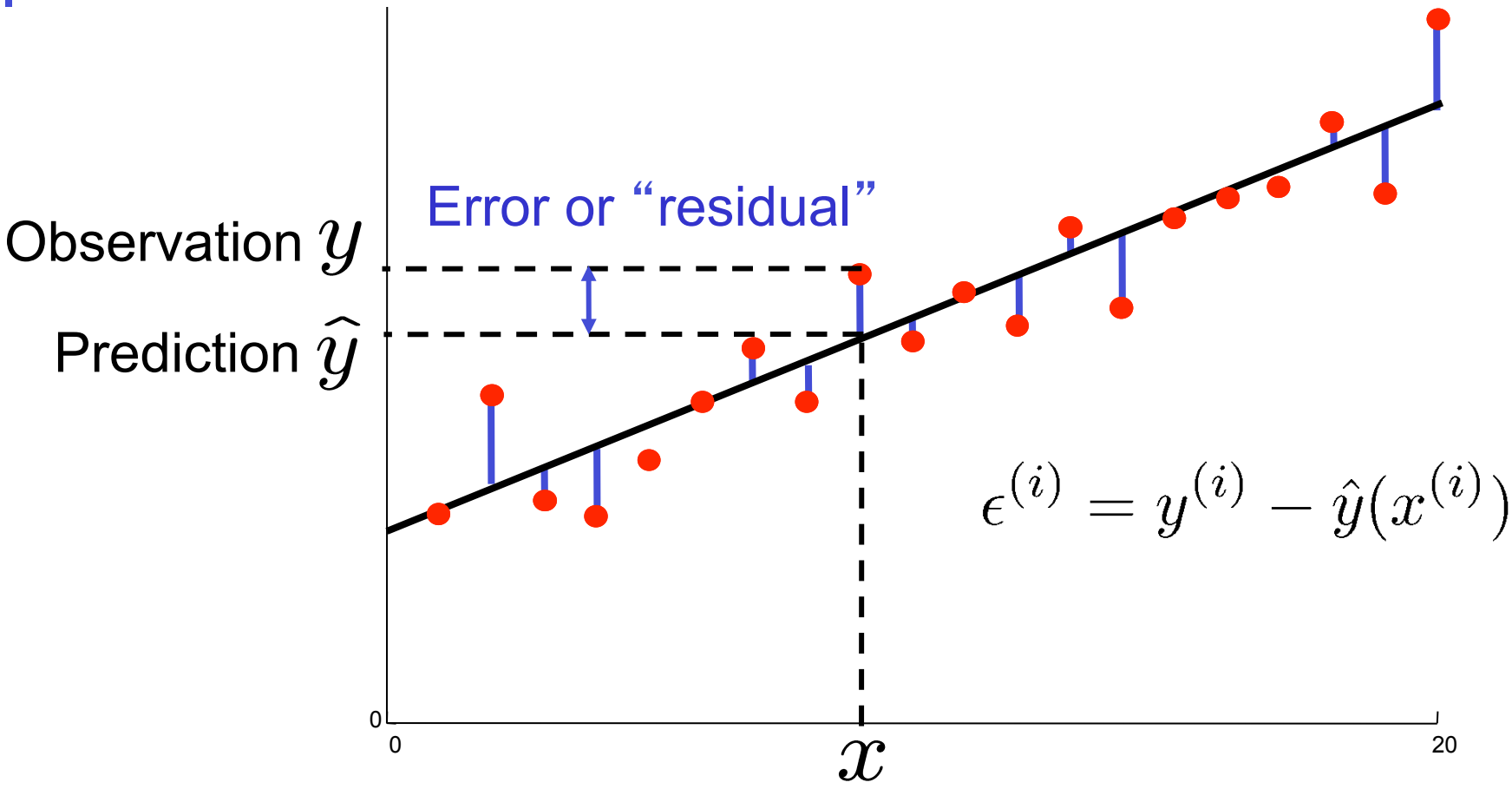
"Predictor":
Evaluate line:

$$r = \theta_0 + \theta_1 x_1$$

return r

- Define form of function f(x) explicitly
- Find a good f(x) within that family

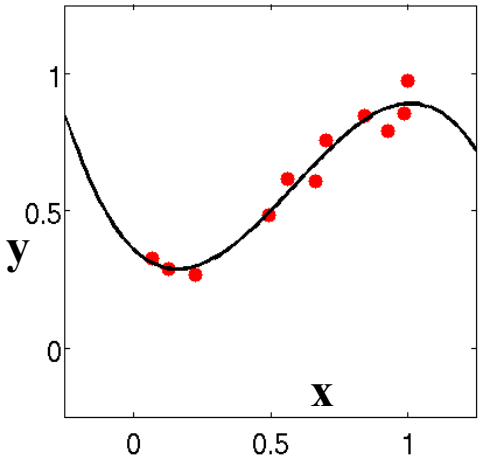
Measuring error



$$\text{MSE} = \frac{1}{m} \sum_i \left(y^{(i)} - \hat{y}(x^{(i)}) \right)^2$$

Regression vs. Classification

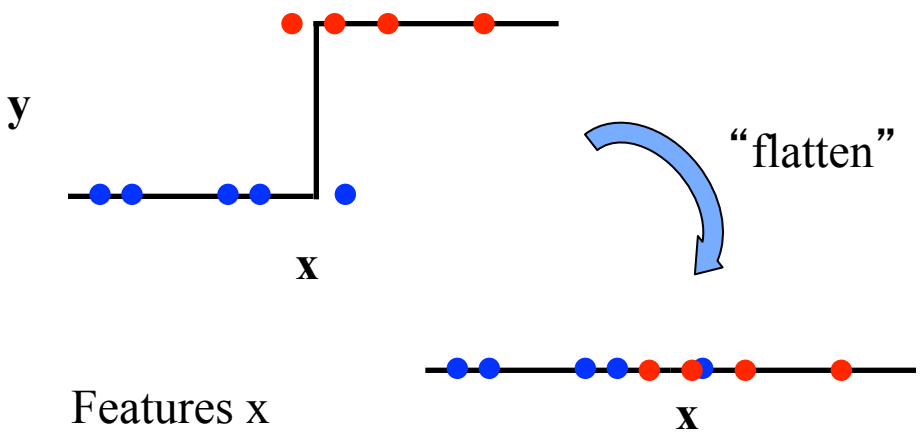
Regression



Features x
Real-valued target y

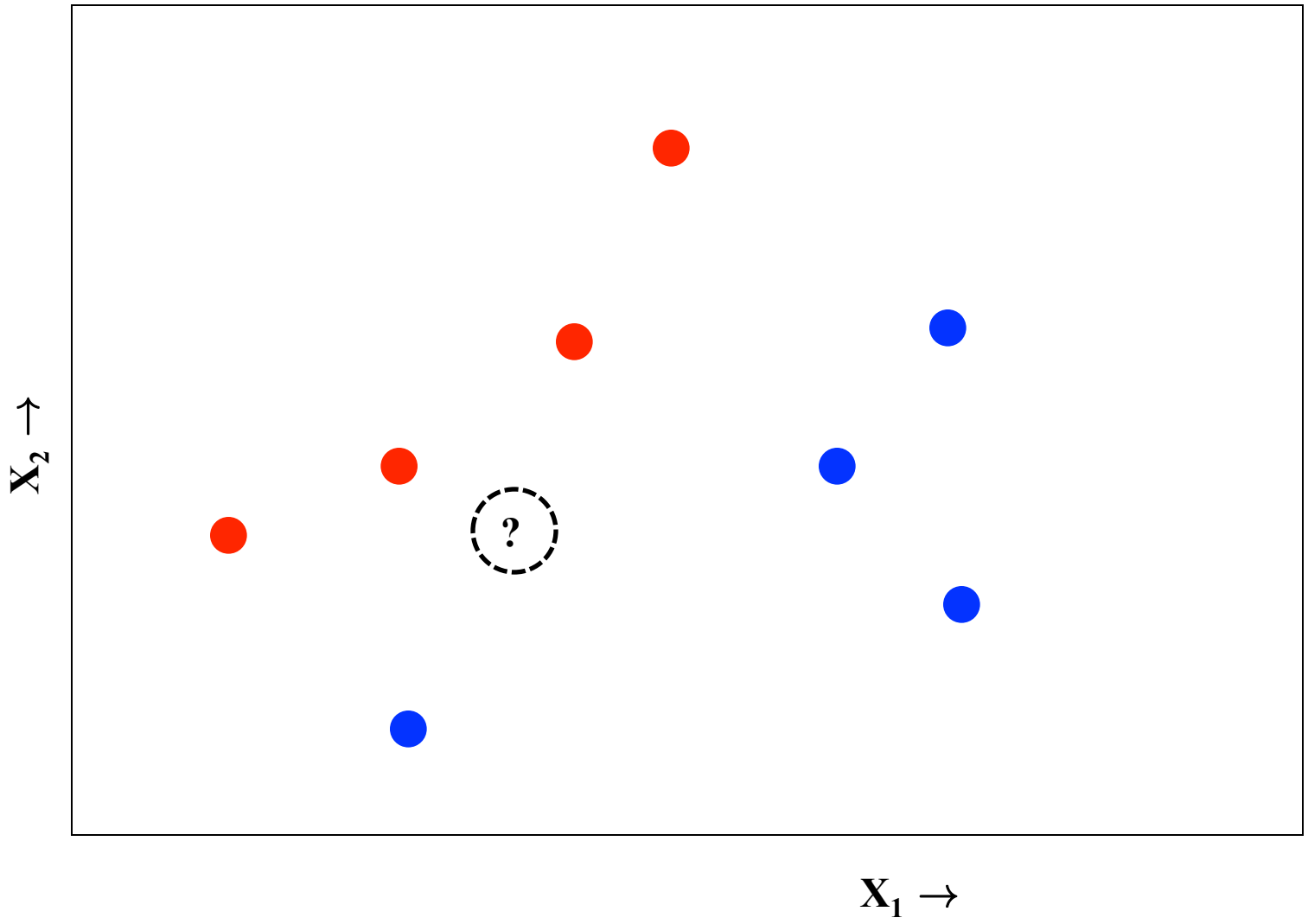
Predict continuous function ŷ(x)

Classification

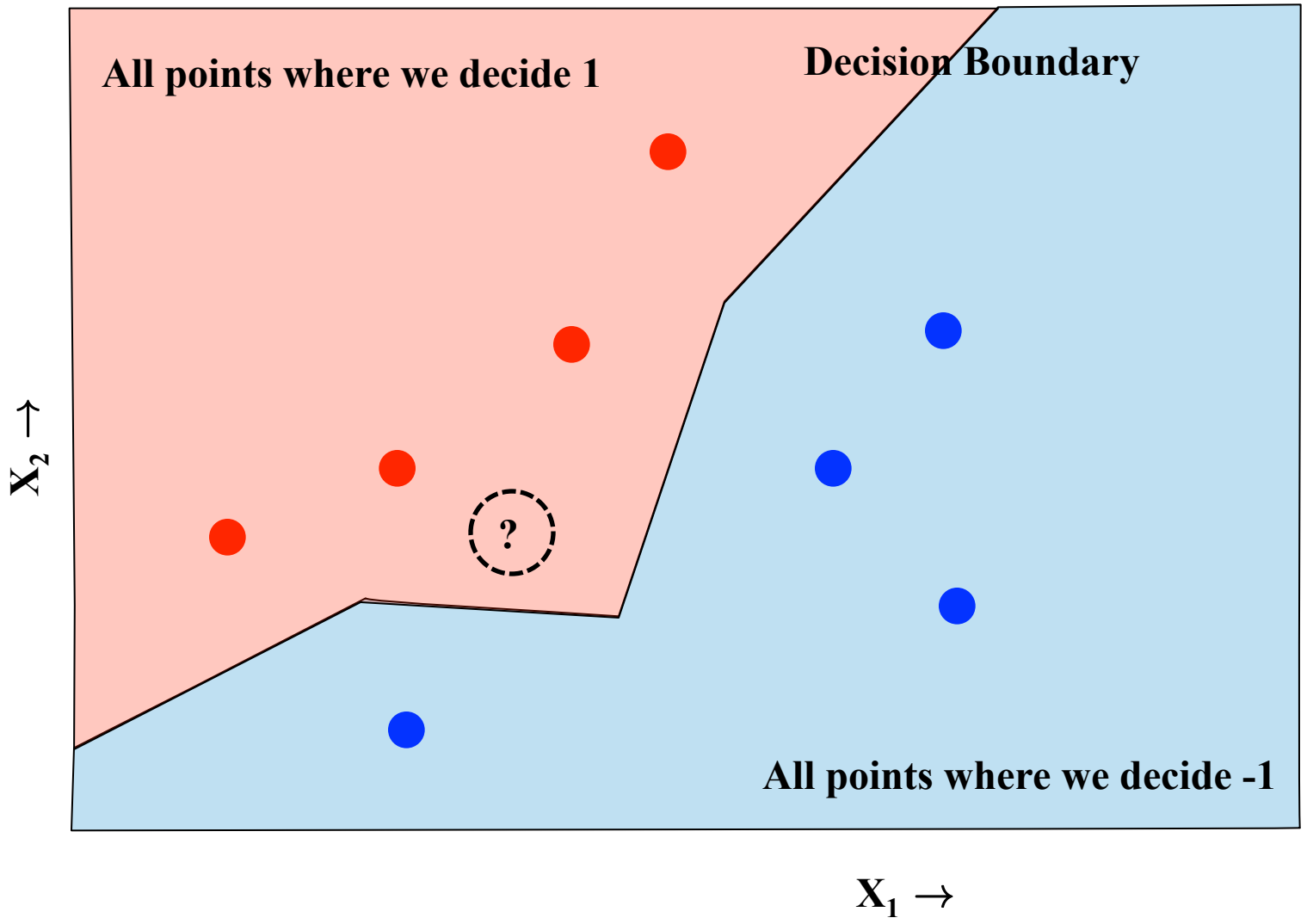


Features x
Discrete class c
(usually 0/1 or +1/-1)
Predict discrete function ŷ(x)

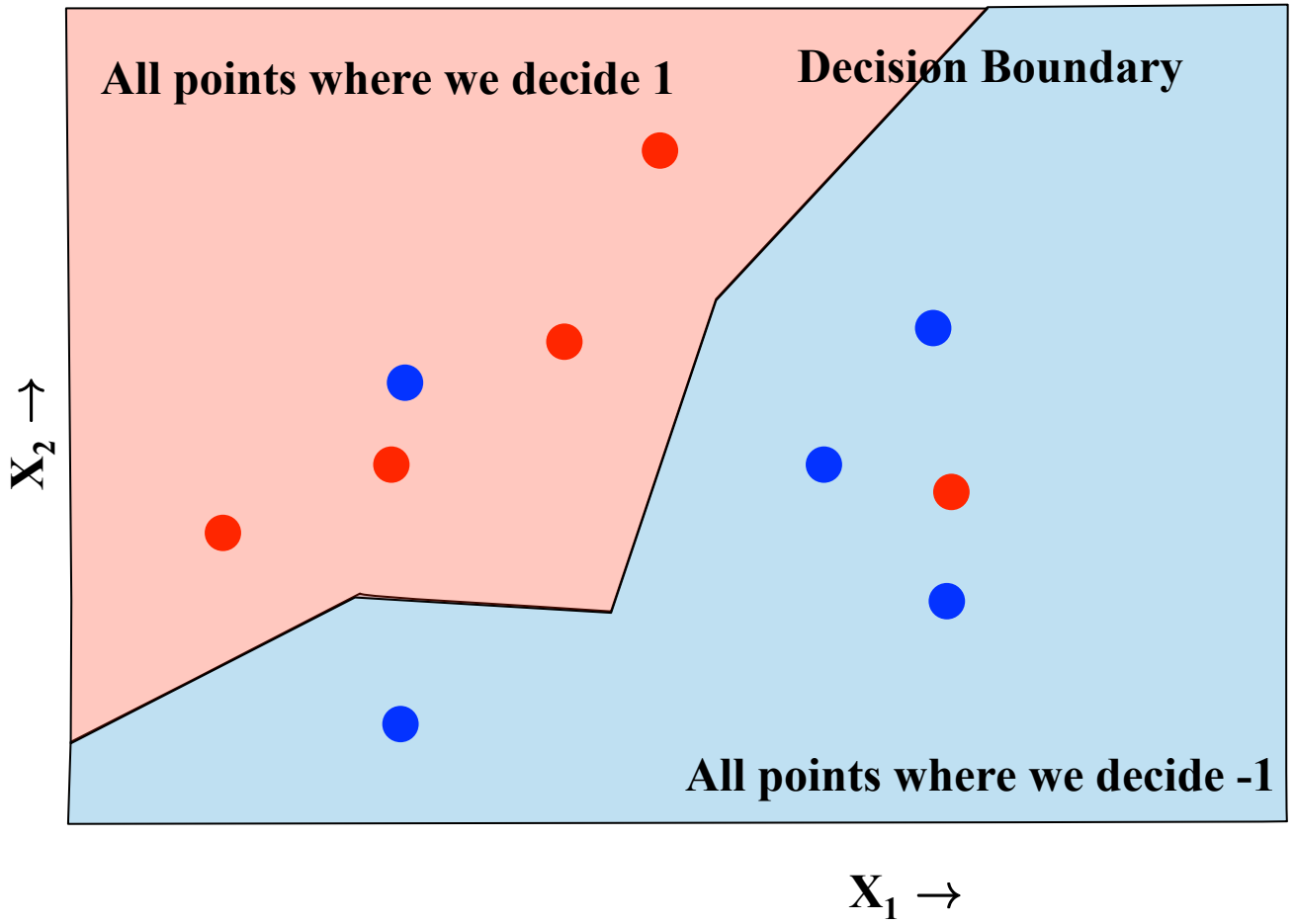
Classification



Classification

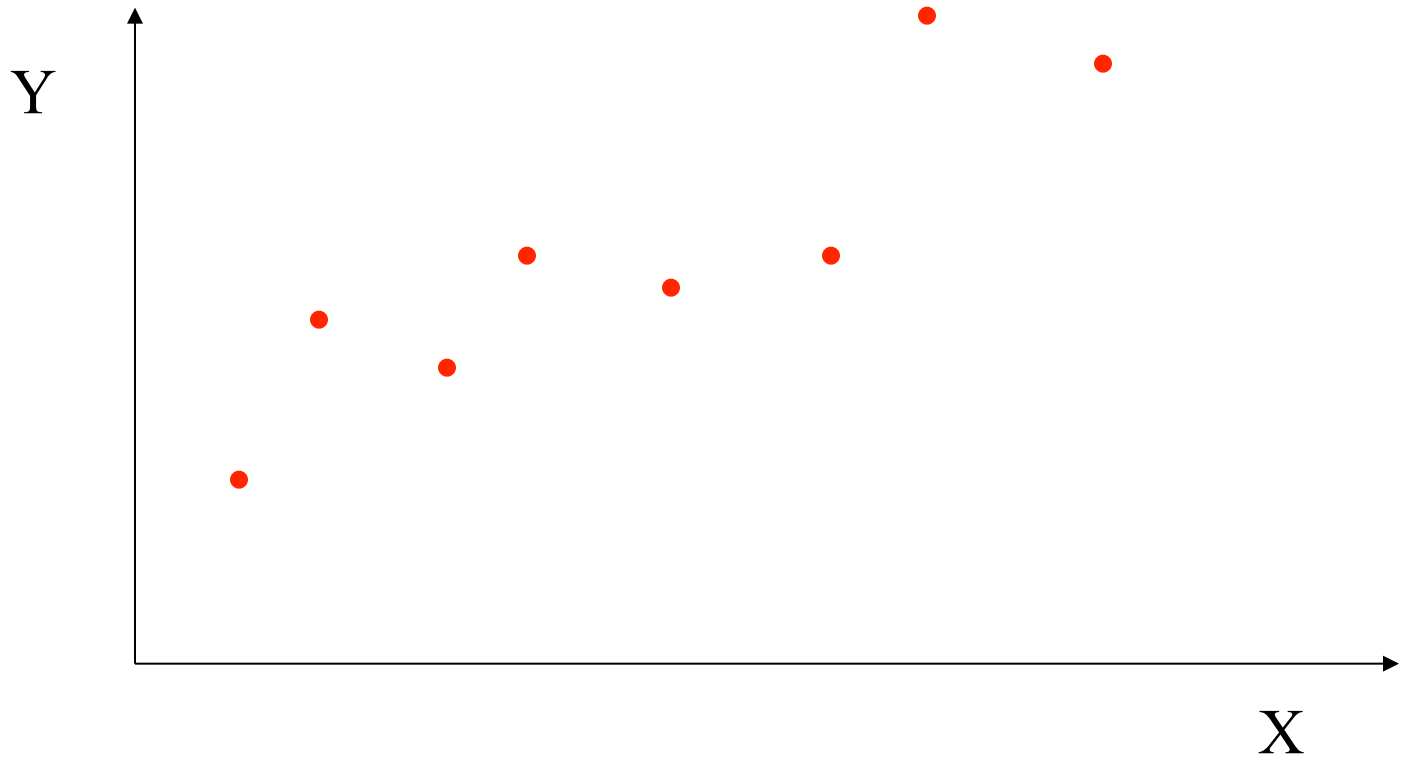


Measuring error



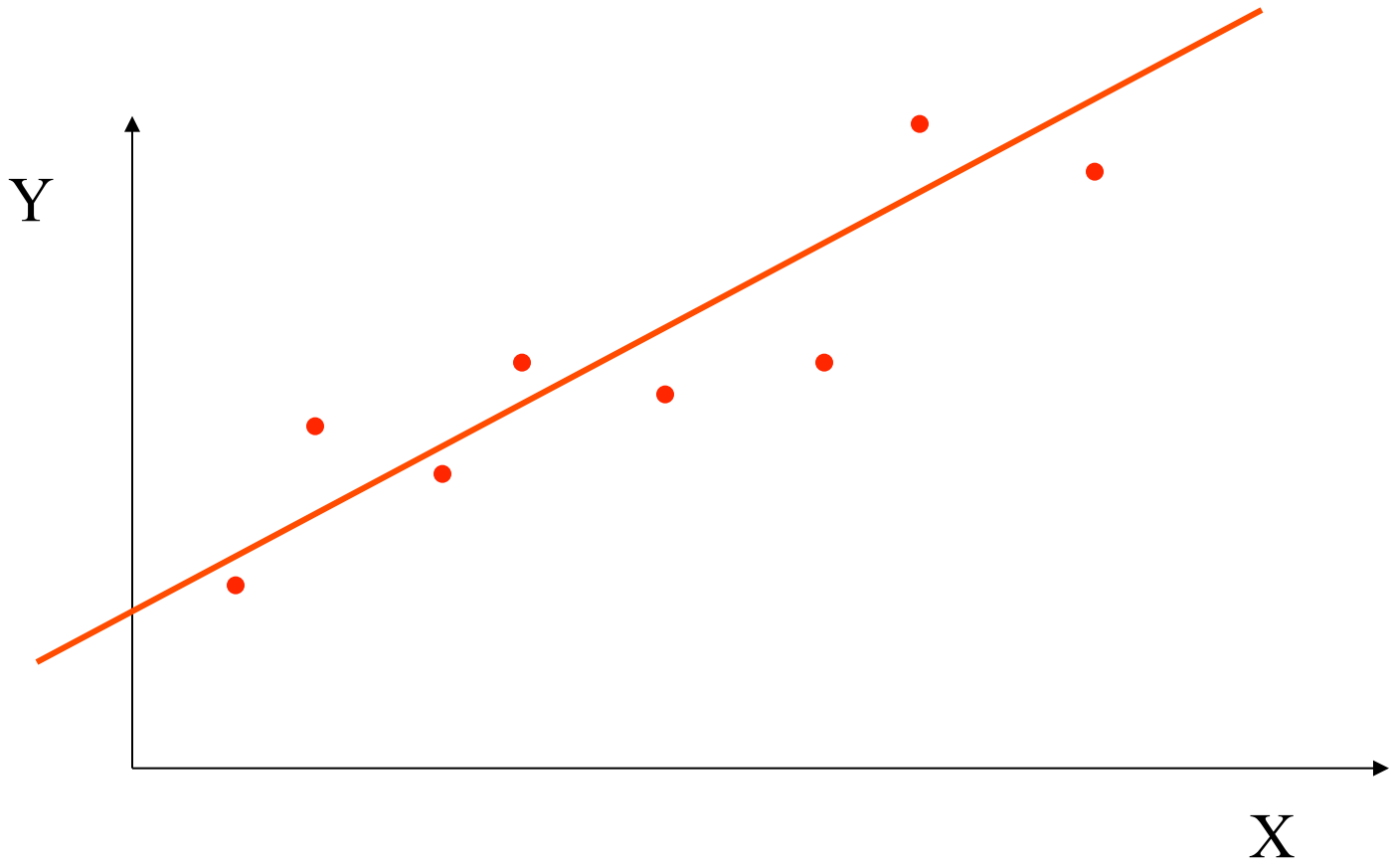
$$\mathrm{ERR} = \frac{1}{m} \sum_{i} \left[y^{(i)} \neq \hat{y}(x^{(i)}) \right]$$

Overfitting and complexity

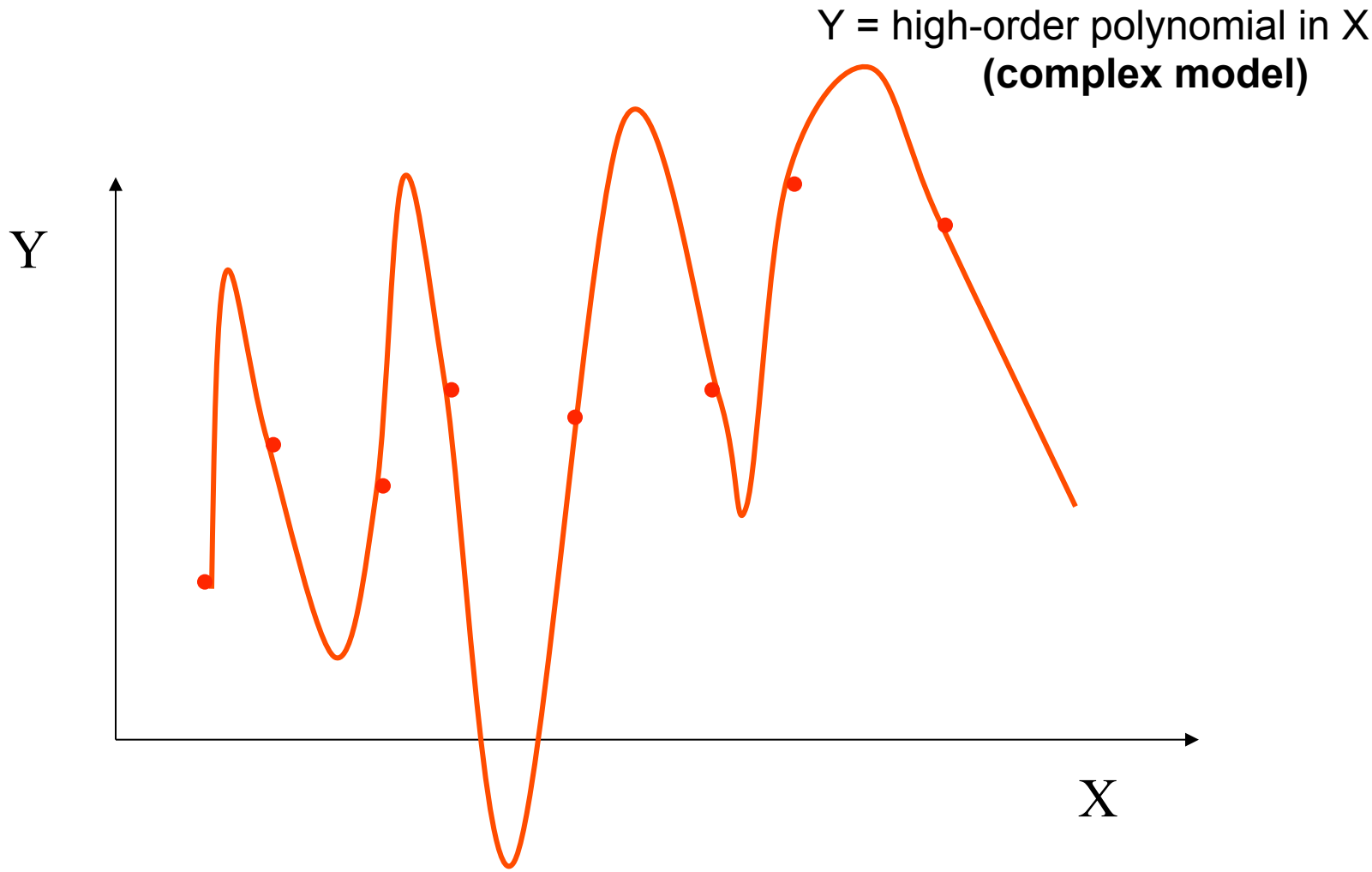


Overfitting and complexity

Simple model: Y= aX + b + e

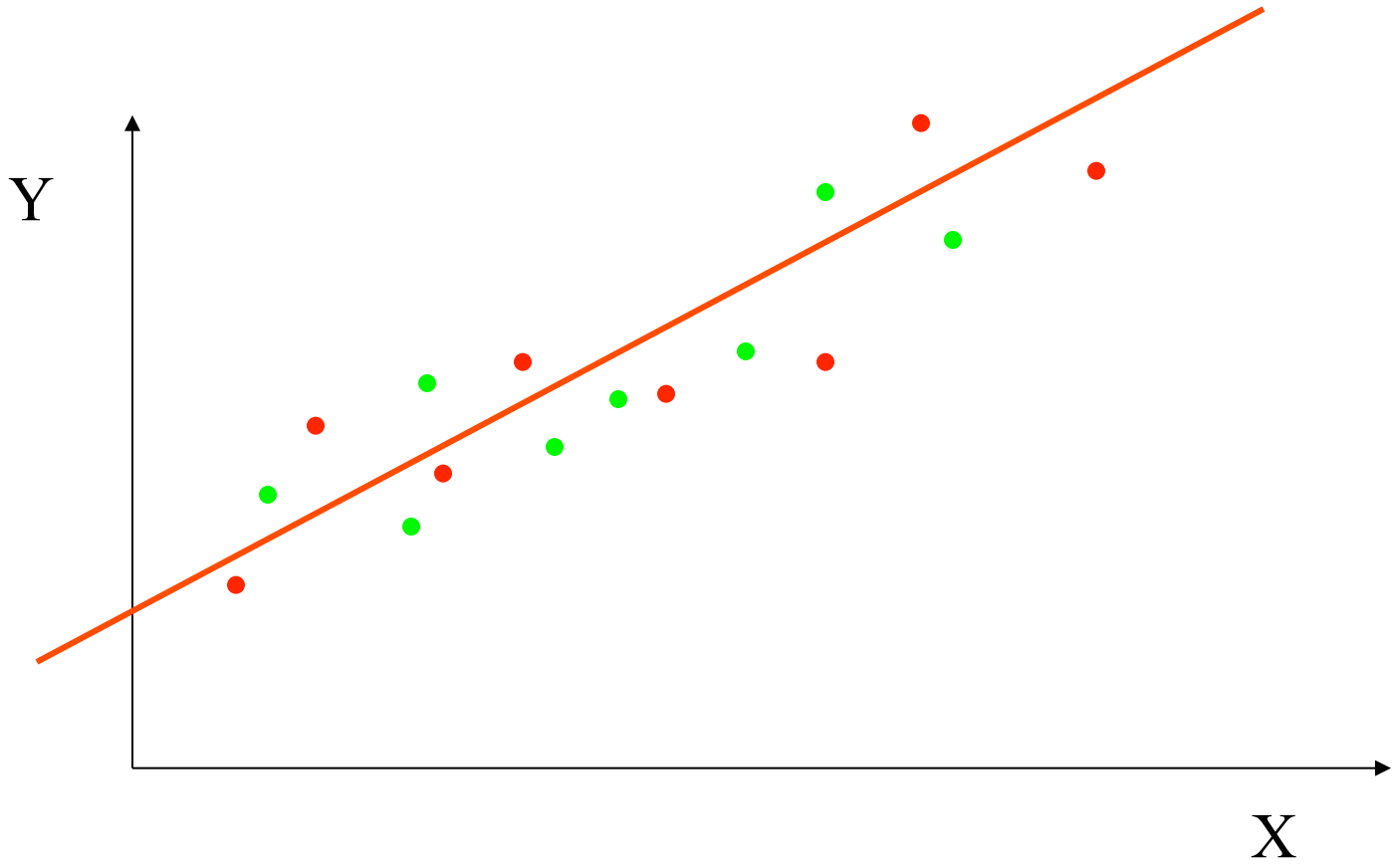


Overfitting and complexity

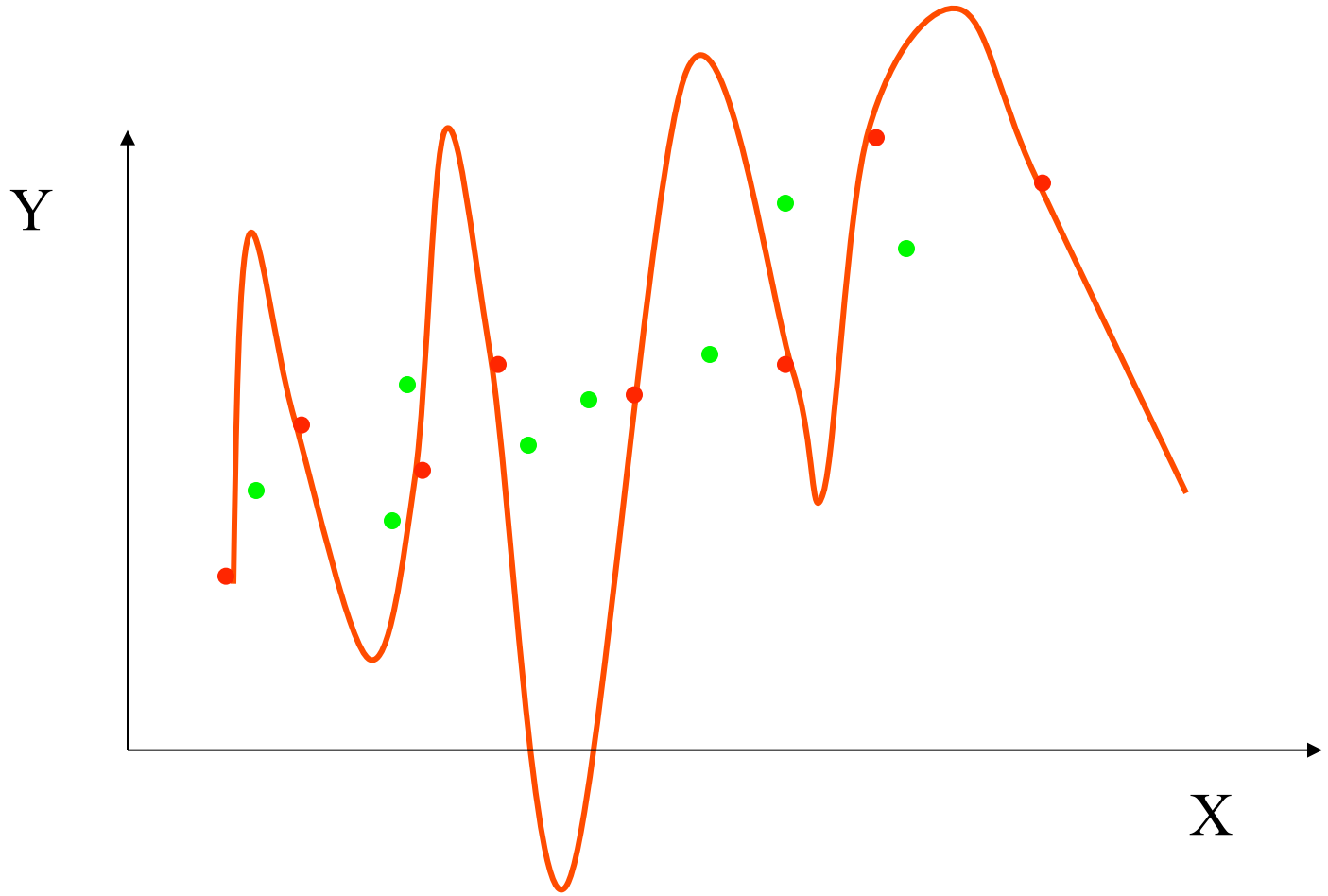


Overfitting and complexity

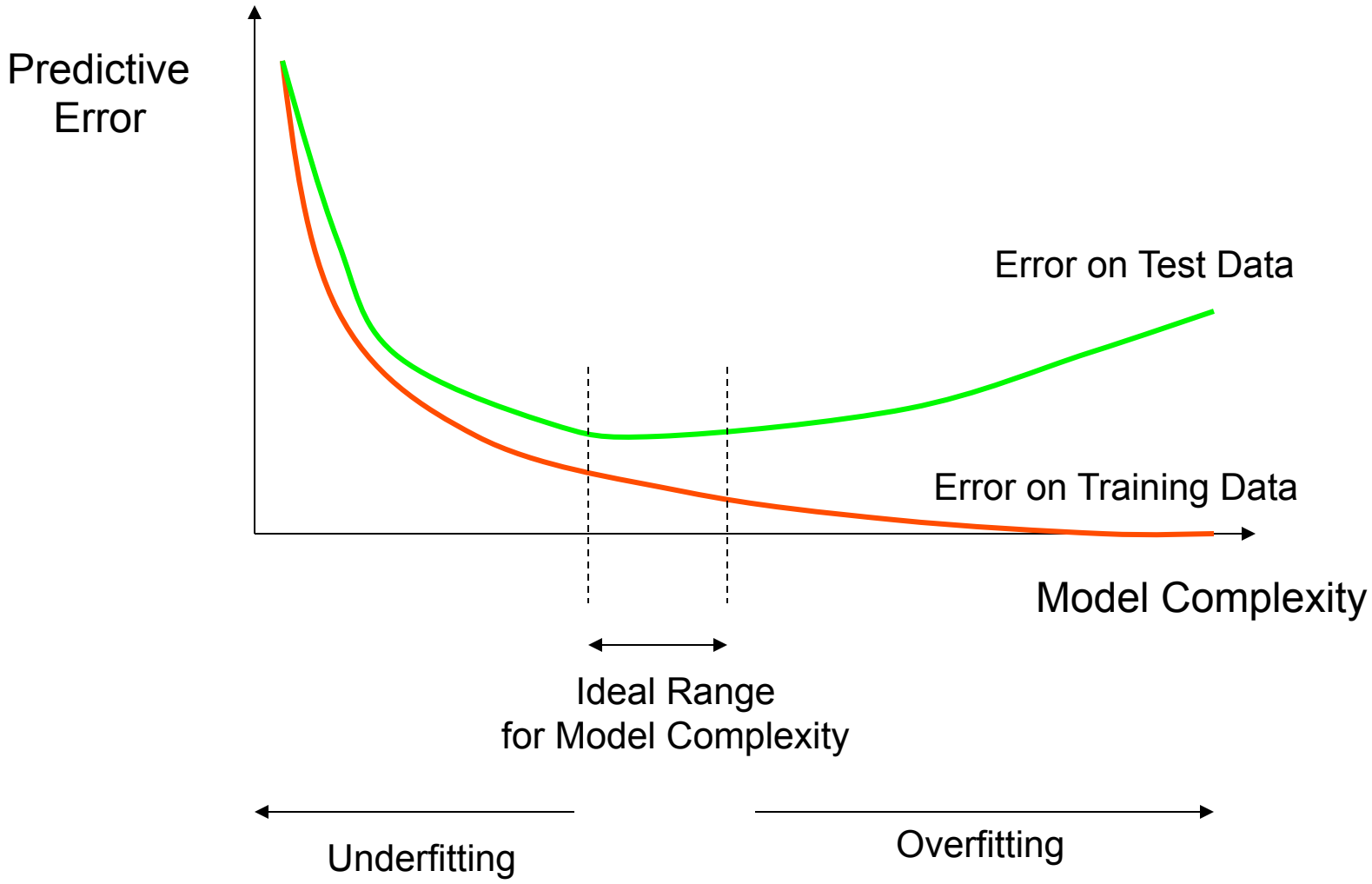
Simple model: Y= aX + b + e



Overfitting and complexity



How Overfitting affects Prediction



Competitions

- Training data
  - Used to build your model(s)
- Validation data
  - Used to assess, select among, or combine models
  - Personal validation; leaderboard; …
- Test data
  - Used to estimate “real world” performance

#	Δ1w	Team Name * in the money	Score	Entries	Last Submission UT
1	-	BrickMover *	1.21251	40	Sat, 31 Aug 2013 23:
2	new	vsu *	1.21552	13	Sat, 31 Aug 2013 20:
3	↑2	Merlion	1.22724	29	Sat, 31 Aug 2013 23:
4	↓2	Sergey	1.22856	15	Sat, 31 Aug 2013 23:
5	new	liuyongqi	1.22980	13	Sat, 31 Aug 2013 13:

Summary

- What is machine learning?
  - Types of machine learning
  - How machine learning works
- Supervised learning
  - Training data: features x, targets y
- Regression
  - (x,y) scatterplots; predictor outputs f(x)
- Classification
  - (x,x) scatterplots
  - Decision boundaries, colors & symbols
- Complexity
  - Training vs test error
  - Under- & over-fitting