Introduction to Machine Learning

Brown University CSCI 1950-F, Spring 2012
Prof. Erik Sudderth

Lecture 2:
Probability: Discrete Random Variables
Classification: Validation & Model Selection

Many figures courtesy Kevin Murphy's textbook,
Machine Learning: A Probabilistic Perspective

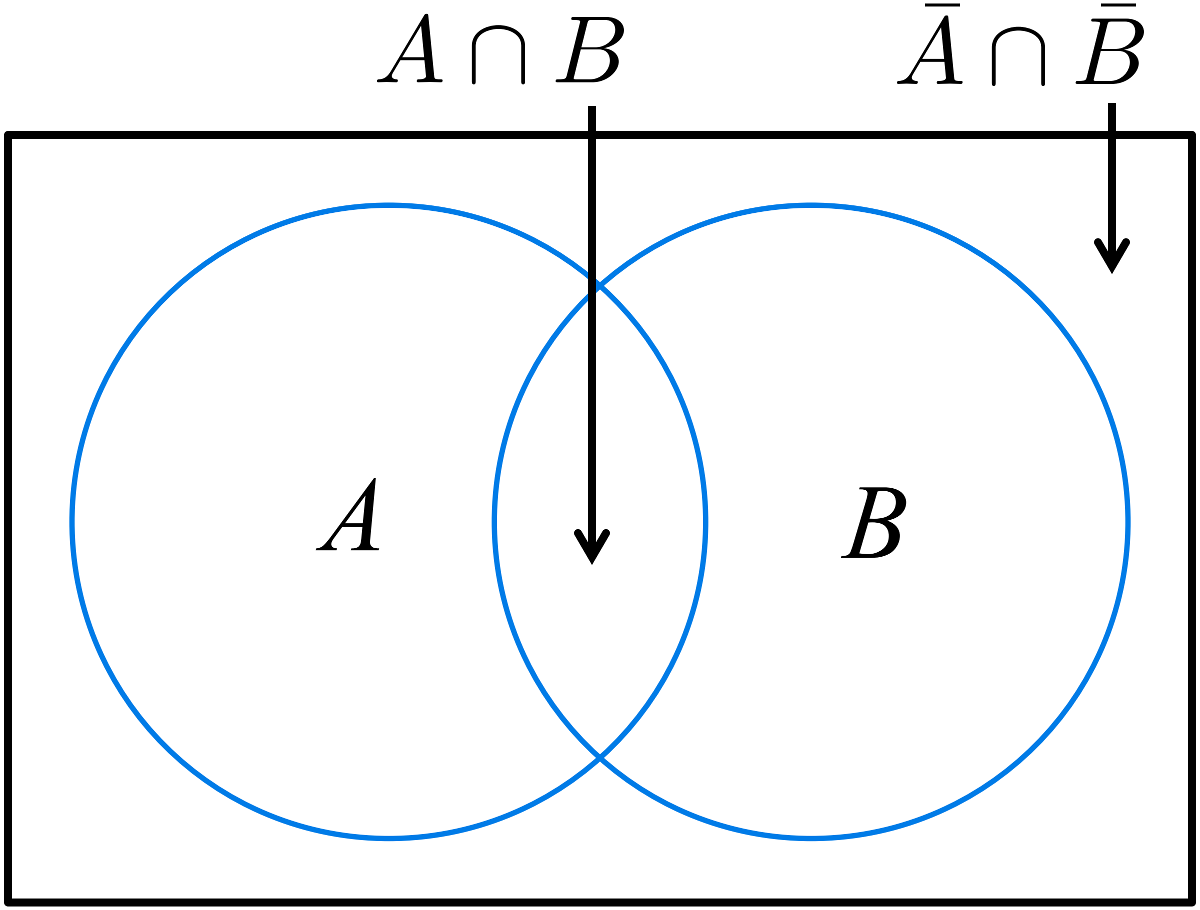
What is Probability?

If I flip this coin, the probability that it will come up heads is 0.5

- *Frequentist Interpretation:* If we flip this coin many times, it will come up heads about half the time. *Probabilities are the expected frequencies of events over repeated trials.*
- *Bayesian Interpretation:* I believe that my next toss of this coin is equally likely to come up heads or tails. *Probabilities quantify subjective beliefs about single events.*
- Viewpoints play complementary roles in *machine learning:*
  - Bayesian view used to build models based on domain knowledge, and automatically derive learning algorithms
  - Frequentist view used to analyze worst case behavior of learning algorithms, in limit of large datasets
- From either view, basic mathematics is the same!

Probability of Two Events

$$p(A \cup B) = p(A) + p(B) - p(A \cap B)$$



$$0 \leq p(A) \leq 1$$

$$p(\bar{A}) = 1 - p(A)$$

$$p(A \cap B) = p(A \mid B)p(B)$$

$$p(A \mid B) = \frac{p(A \cap B)}{p(B)}$$

Discrete Random Variables

$X \longrightarrow$ discrete random variable

$\mathcal{X} \longrightarrow$ sample space of possible outcomes, which may be finite or countably infinite

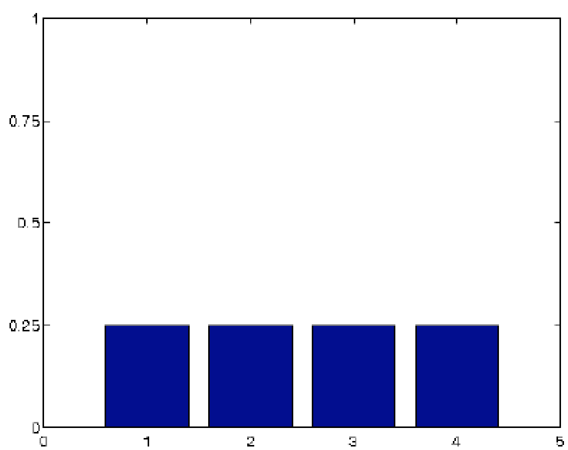
$x \in \mathcal{X} \longrightarrow$ outcome of sample of discrete random variable

$p(X = x) \longrightarrow$ probability distribution (probability mass function)

$p(x) \longrightarrow$ shorthand used when no ambiguity

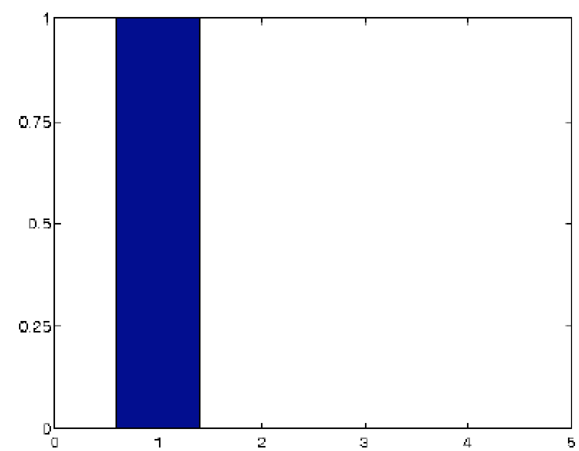
$$0 \leq p(x) \leq 1 \text{ for all } x \in \mathcal{X}$$

$$\sum_{x \in \mathcal{X}} p(x) = 1$$



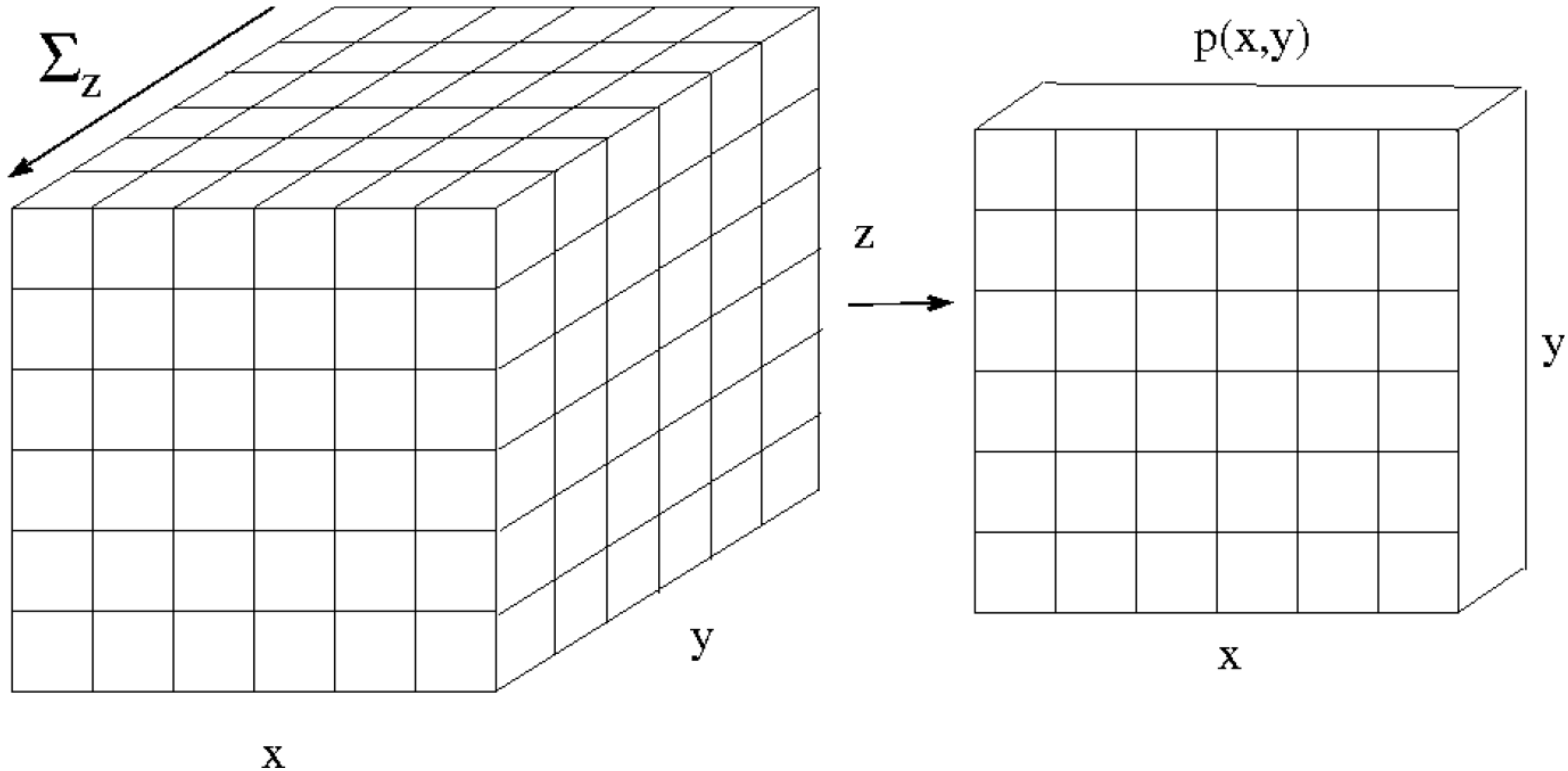
uniform distribution

$$\mathcal{X} = \{1, 2, 3, 4\}$$



degenerate distribution

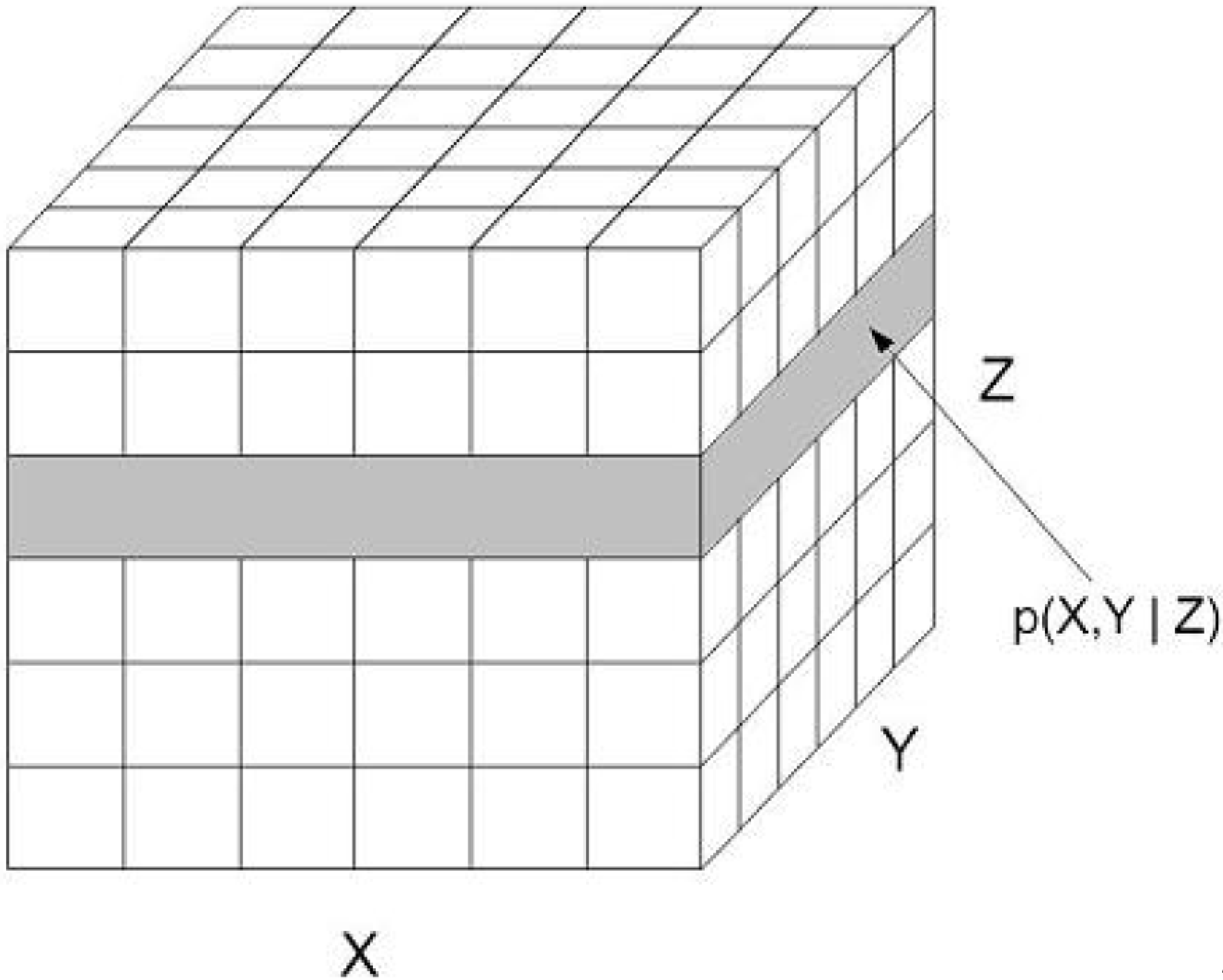
Marginal Distributions



$$p(x,y) = \sum_{z \in \mathcal{Z}} p(x,y,z)$$

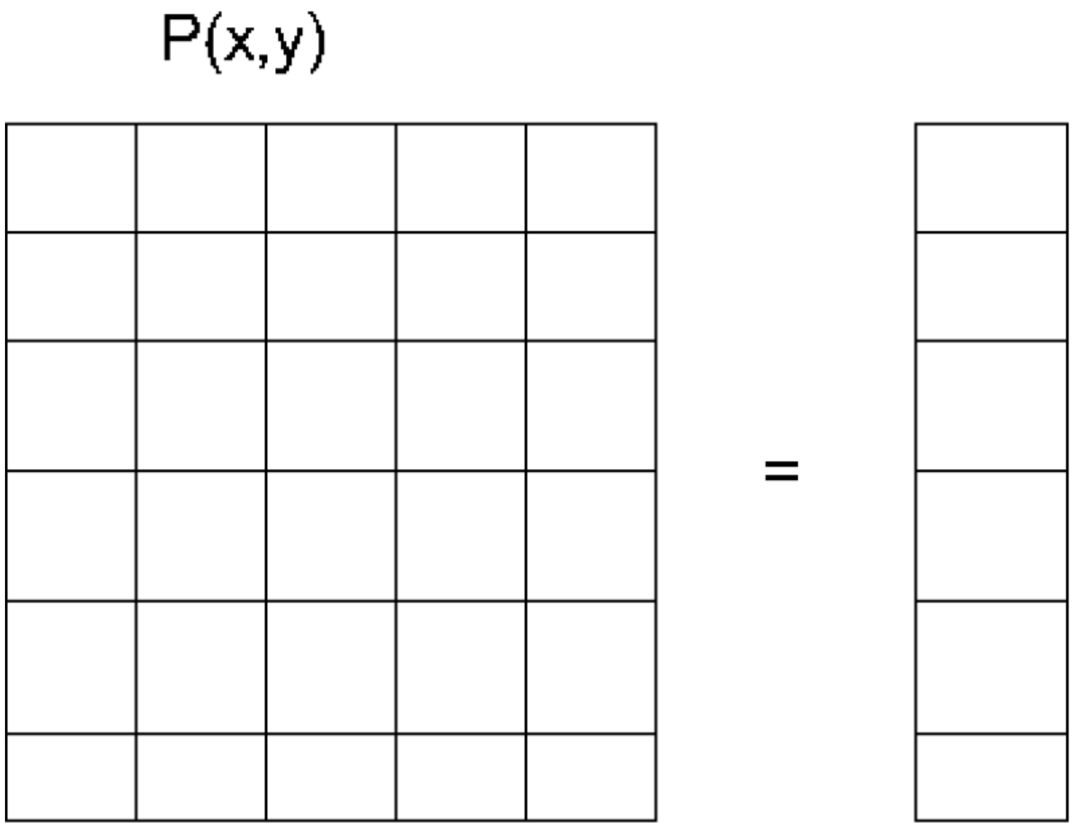
$$p(x) = \sum_{y \in \mathcal{Y}} p(x,y)$$

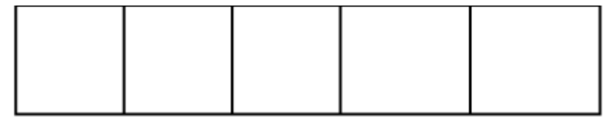
Conditional Distributions



$$p(x, y \mid Z = z) = \frac{p(x, y, z)}{p(z)}$$

Independent Random Variables





$$X \perp Y$$

$\Updownarrow$

$$p(x, y) = p(x)p(y)$$

$$\text{for all } x \in \mathcal{X}, y \in \mathcal{Y}$$

Equivalent conditions on conditional probabilities:

$$p(x \mid Y = y) = p(x) \text{ and } p(y) > 0 \text{ for all } y \in \mathcal{Y}$$

$$p(y \mid X = x) = p(y) \text{ and } p(x) > 0 \text{ for all } x \in \mathcal{X}$$

Bayes Rule (Bayes Theorem)

$$p(x, y) = p(x)p(y \mid x) = p(y)p(x \mid y)$$

$$p(x \mid y) = \frac{p(x, y)}{p(y)} = \frac{p(y \mid x)p(x)}{\sum_{x' \in \mathcal{X}} p(x')p(y \mid x')}$$

$$\propto p(y \mid x)p(x)$$

- A basic identity from the definition of conditional probability
- Used in ways that have nothing to do with Bayesian statistics!
- Typical application to learning and data analysis:

$X \longrightarrow$ unknown parameters we would like to infer

$Y = y \longrightarrow$ observed data available for learning

$p(x) \longrightarrow$ prior distribution (domain knowledge)

$p(y \mid x) \longrightarrow$ likelihood function (measurement model)

$p(x \mid y) \longrightarrow$ posterior distribution (learned information)

Binary Random Variables

Bernoulli Distribution: Single toss of a (possibly biased) coin

$$\mathcal{X} = \{0, 1\}$$

$$0 \leq \theta \leq 1$$

$$\mathrm{Ber}(x \mid \theta) = \theta^{\delta(x,1)}(1-\theta)^{\delta(x,0)}$$

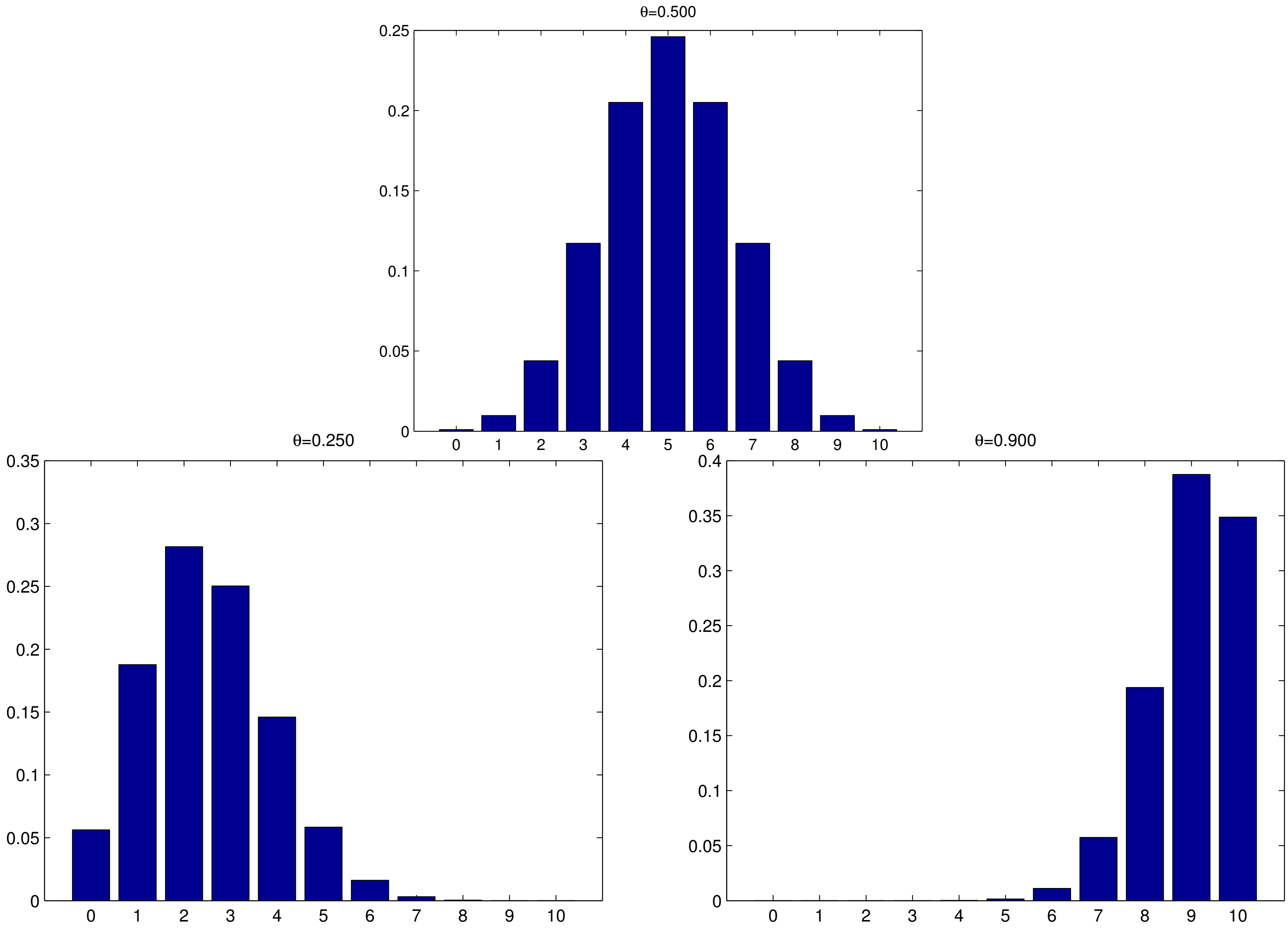
Binomial Distribution: Toss a single (possibly biased) coin *n* times, and record the number *k* of times it comes up heads

$$\mathcal{K} = \{0, 1, 2, \ldots, n\}$$

$$0 \leq \theta \leq 1$$

$$\mathrm{Bin}(k \mid n, \theta) = \binom{n}{k} \theta^k (1-\theta)^{n-k} \qquad \binom{n}{k} = \frac{n!}{(n-k)!k!}$$

Binomial Distributions



Categorical Random Variables

Multinoulli Distribution: Single roll of a (possibly biased) die

$$\mathcal{X} = \{0,1\}^K, \sum_{k=1}^{K} x_k = 1$$

binary vector encoding

$$\theta = (\theta_1, \theta_2, \ldots, \theta_K), \theta_k \geq 0, \sum_{k=1}^{K} \theta_k = 1$$

$$\mathrm{Cat}(x \mid \theta) = \prod_{k=1}^{K} \theta_k^{x_k}$$

Multinomial Distribution: Roll a single (possibly biased) die *n* times, and record the number n_k of each possible outcome

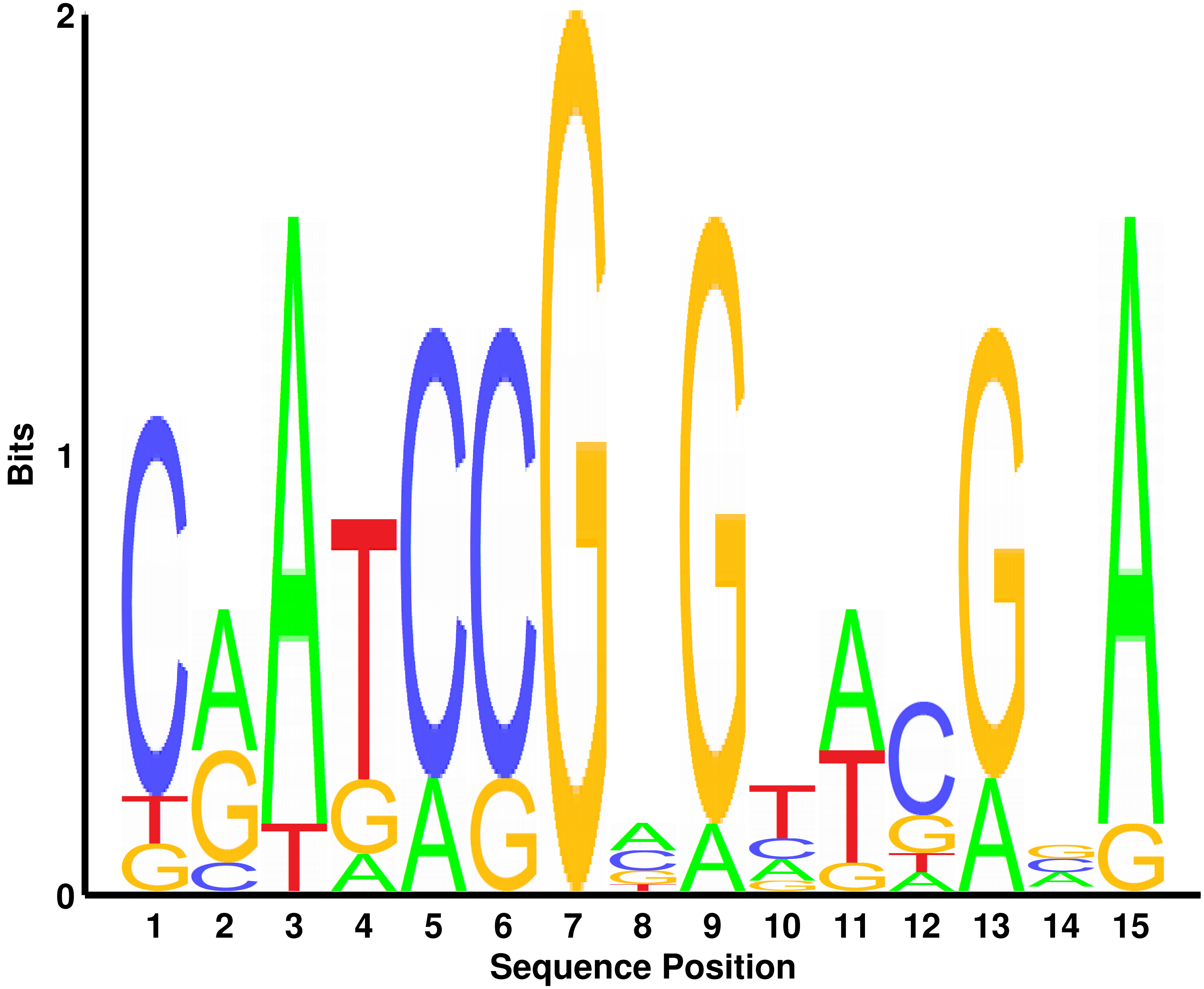
$$\mathrm{Mu}(x \mid n, \theta) = \binom{n}{n_1 \ldots n_K} \prod_{k=1}^{K} \theta_k^{n_k}$$

$$n_k = \sum_{i=1}^{n} x_{ik}$$

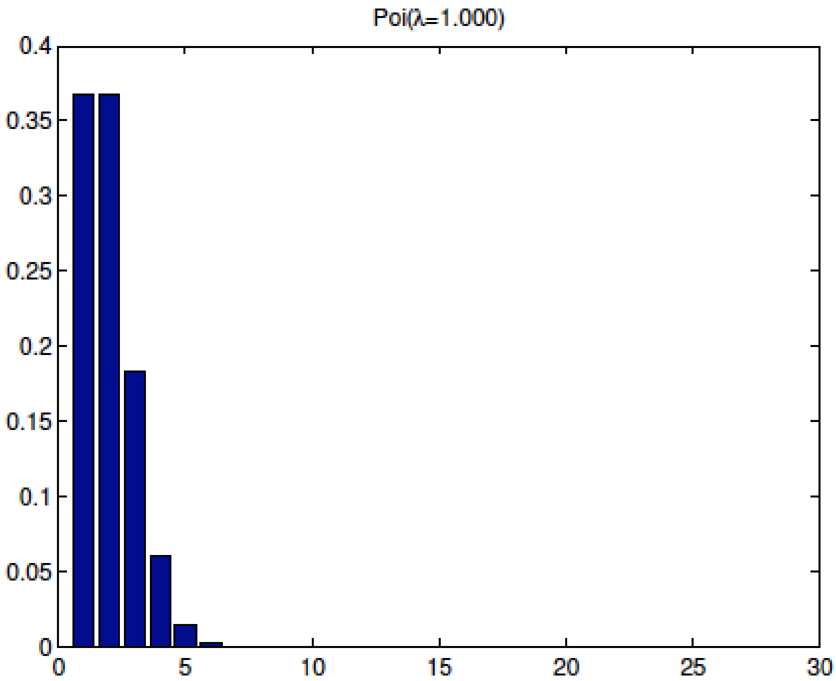
Aligned DNA Sequences

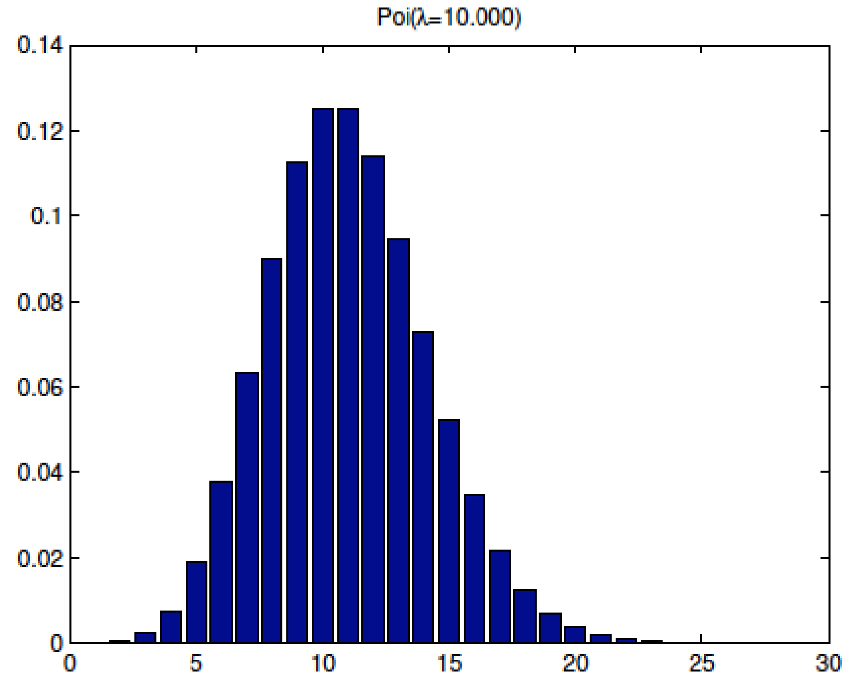
c	g	a	t	a	c	g	g	g	g	t	c	g	a	a
c	a	a	t	c	c	g	a	g	a	t	c	g	c	a
c	a	a	t	c	c	g	t	g	t	t	g	g	g	a
c	a	a	t	c	g	g	c	a	t	g	c	g	g	g
c	g	a	g	c	c	g	c	g	t	a	c	g	a	a
c	a	t	a	c	g	g	a	g	c	a	c	g	a	a
t	a	a	t	c	c	g	g	g	c	a	t	g	t	a
c	g	a	g	c	c	g	a	g	t	a	c	a	g	a
c	c	a	t	c	c	g	c	g	t	a	a	g	c	a
g	g	a	t	a	c	g	a	g	a	t	g	a	c	a

Multinomial Model of DNA



Poisson Distribution for Counts





$$\mathcal{X} = \{0, 1, 2, 3, \ldots\}$$

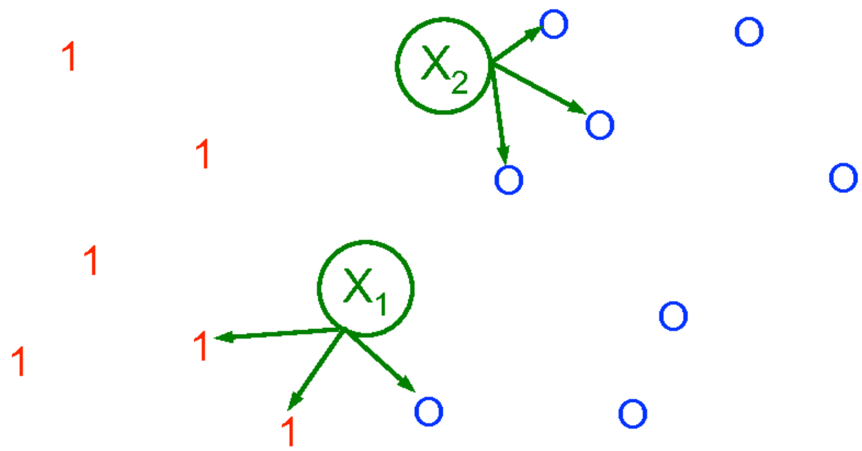
$$\mathrm{Poi}(x \mid \theta) = e^{-\theta} \frac{\theta^x}{x!} \qquad \theta > 0$$

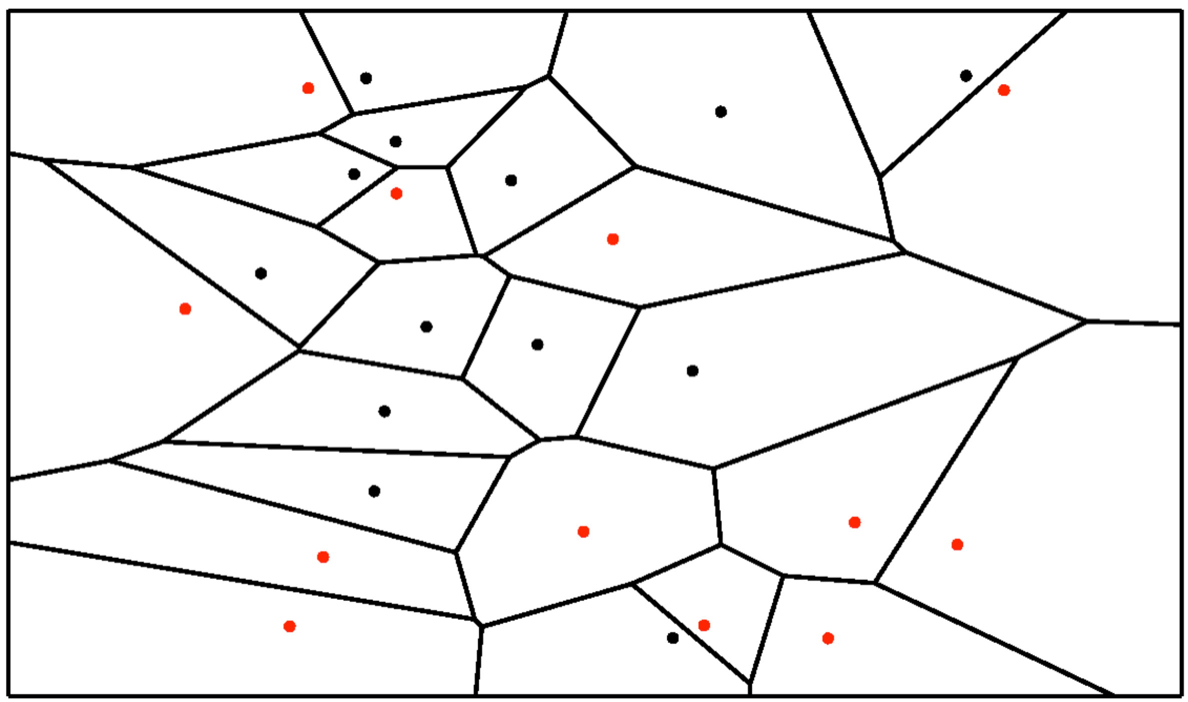
Modeling assumptions reduce number of parameters

Machine Learning Problems

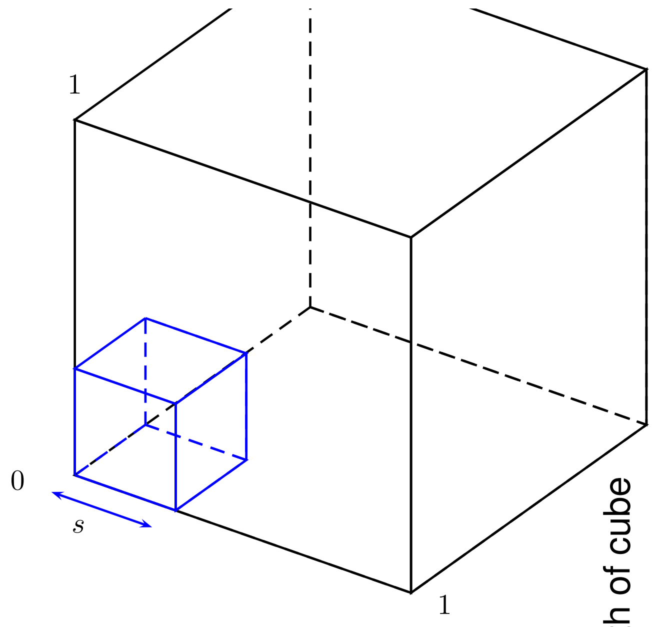
	Supervised Learning	*Unsupervised Learning*
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

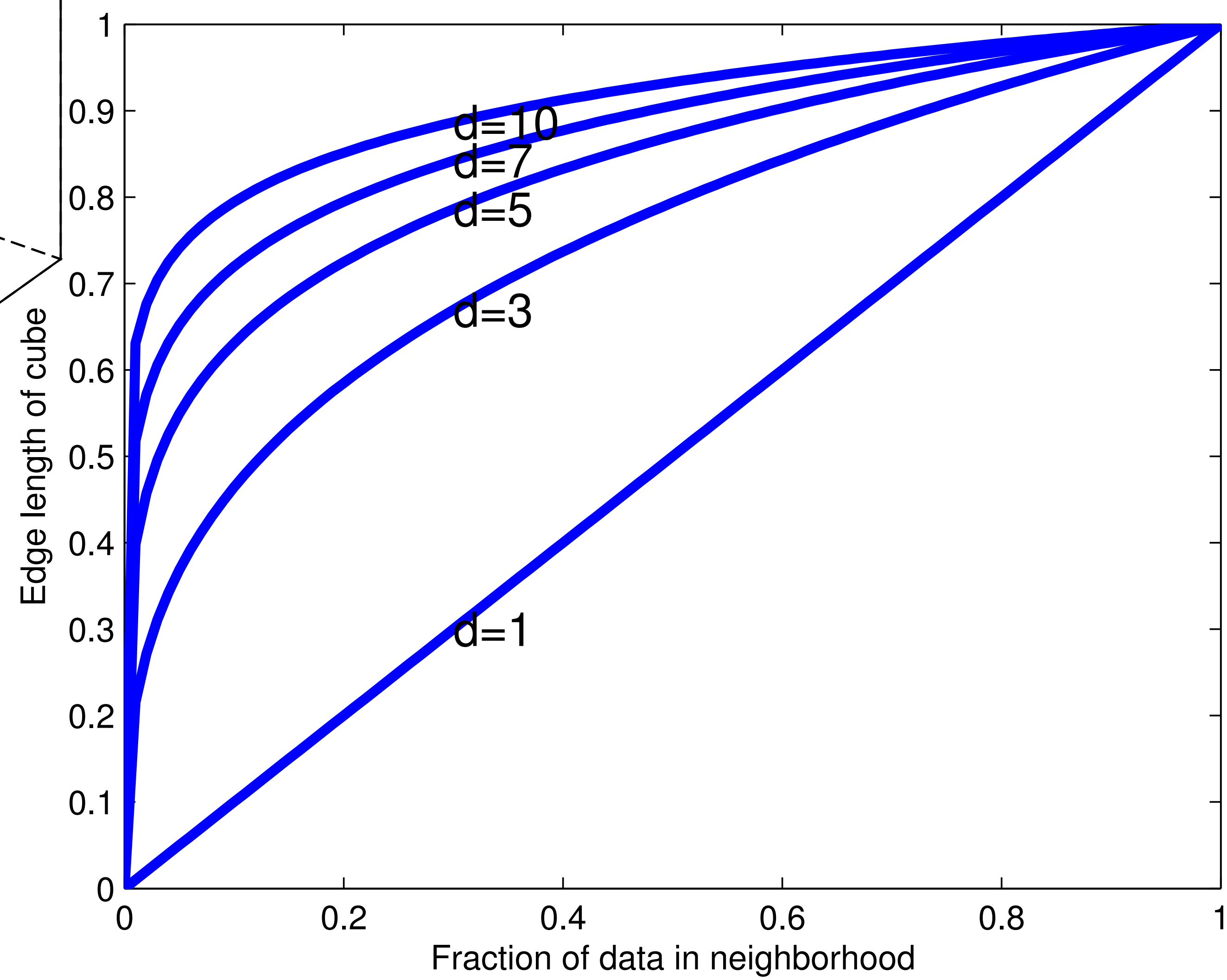
1-Nearest Neighbor Classification



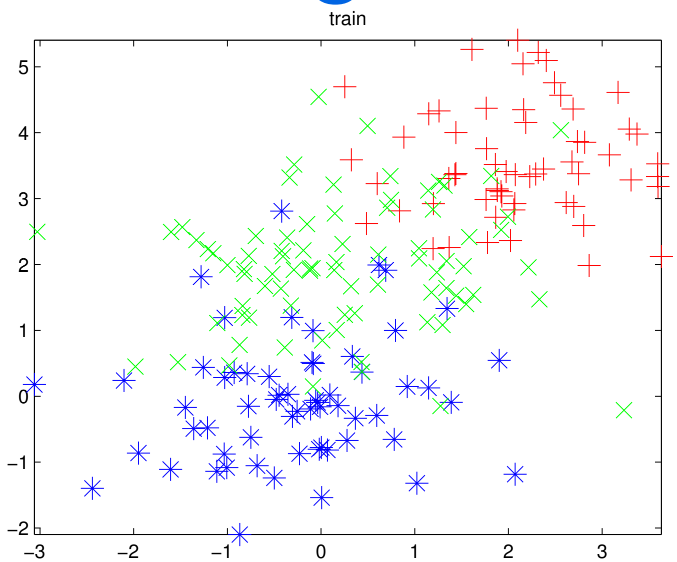


Curse of Dimensionality





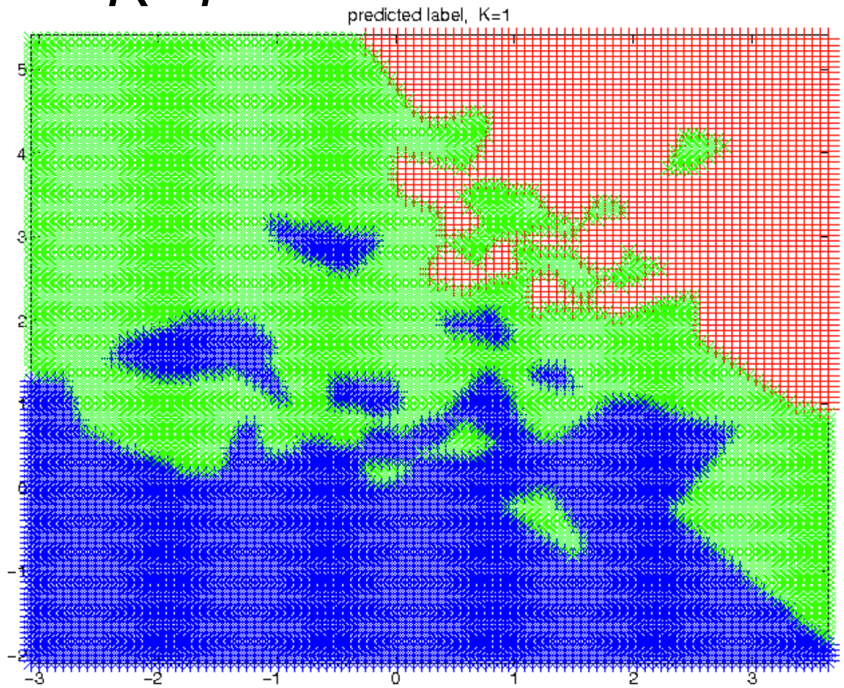
Overfitting & K-Nearest Neighbors



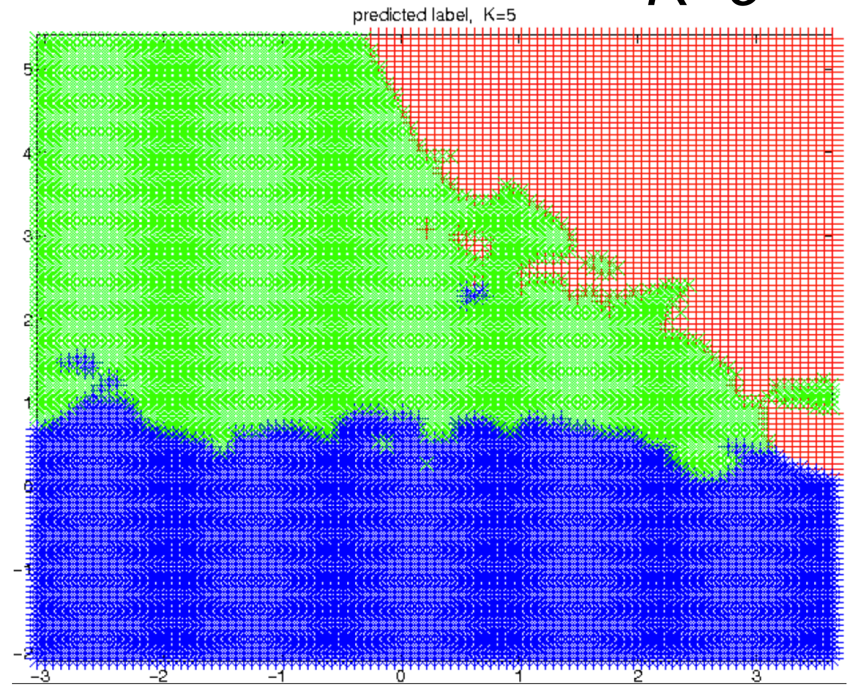
$$p(y = c|\mathbf{x}, \mathcal{D}, K) = \frac{1}{K} \sum_{i \in N_K(\mathbf{x}, \mathcal{D})} \mathbb{I}(y_i = c)$$

How should we choose K?

K=1



K=5



Training and Test Data

Data

- Several candidate learning algorithms or models, each of which can be fit to data and used for prediction
- How can we decide which is best?

Approach 1: Split into train and test data

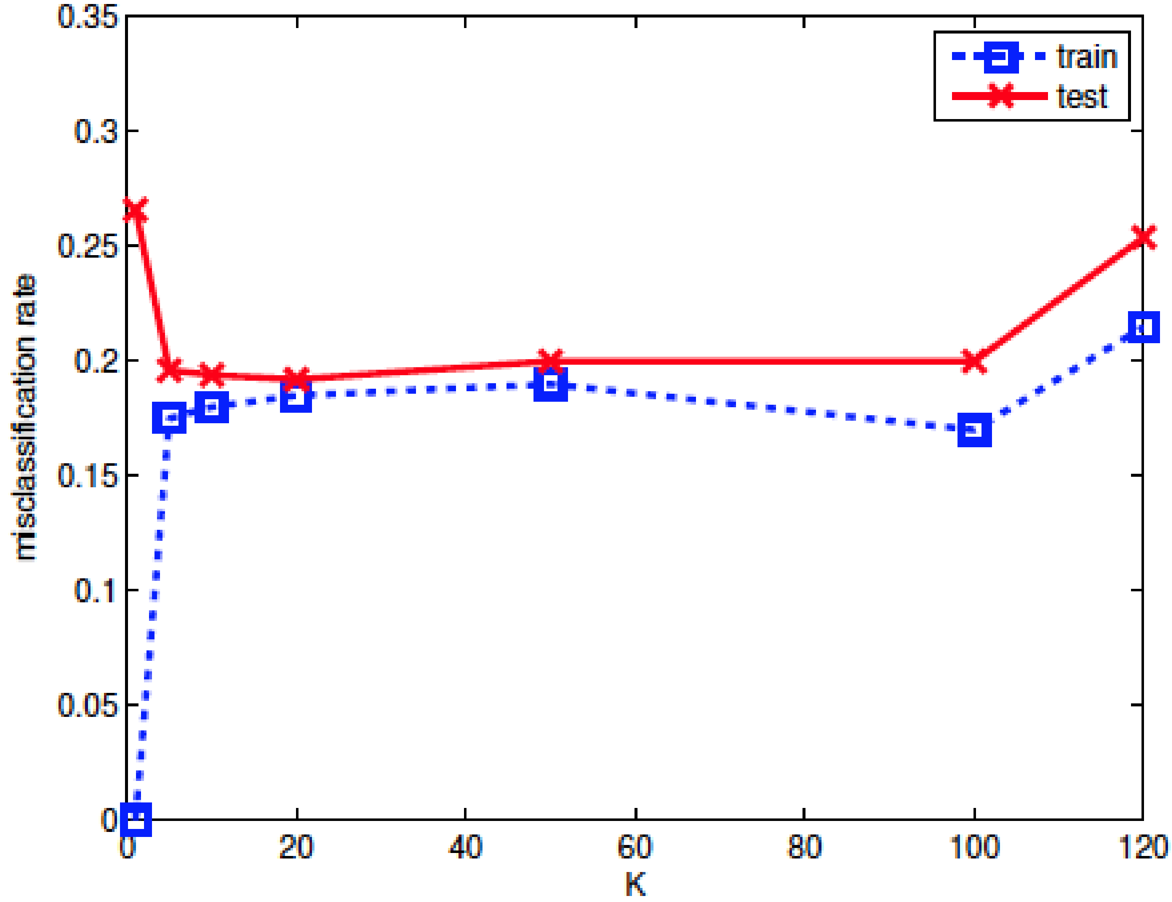
Training Data	Test Data

- Learn parameters of each model from training data
- Evaluate all models on test data, and pick best performer

Problem:

- Over-estimates test performance ("lucky" model)
- Learning algorithms can *never* have access to test data

Example: K Nearest Neighbors



Training, Test, and Validation Data

Data

- Several candidate learning algorithms or models, each of which can be fit to data and used for prediction
- How can we decide which is best?

Approach 2: Reserve some data for validation

Training Data	Validation	Test Data

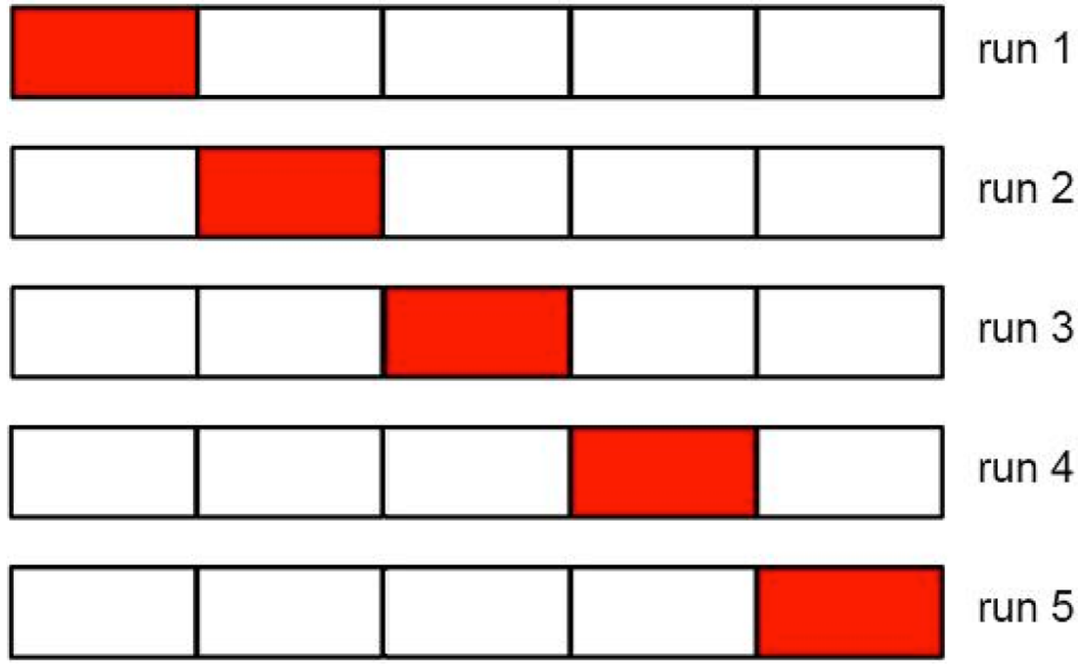
- Learn parameters of each model from training data
- Evaluate models on validation data, pick best performer

Problem:

- Wasteful of training data (learning can't use validation)
- May bias selection towards overly simple models

Cross-Validation

- Divide training data into K equal-sized *folds*
- Train on K-1 folds, evaluate on remainder
- Pick model with best average performance across K trials



How many folds?

- *Bias:* Too few, and effective training dataset much smaller
- *Variance:* Too many, and test performance estimates noisy
- *Cost:* Must run training algorithm once per fold (parallelizable)
- *Practical rule of thumb:* 5-fold or 10-fold cross-validation
- *Theoretically troubled:* Leave-one-out cross-validation, K=N