Machine Learning

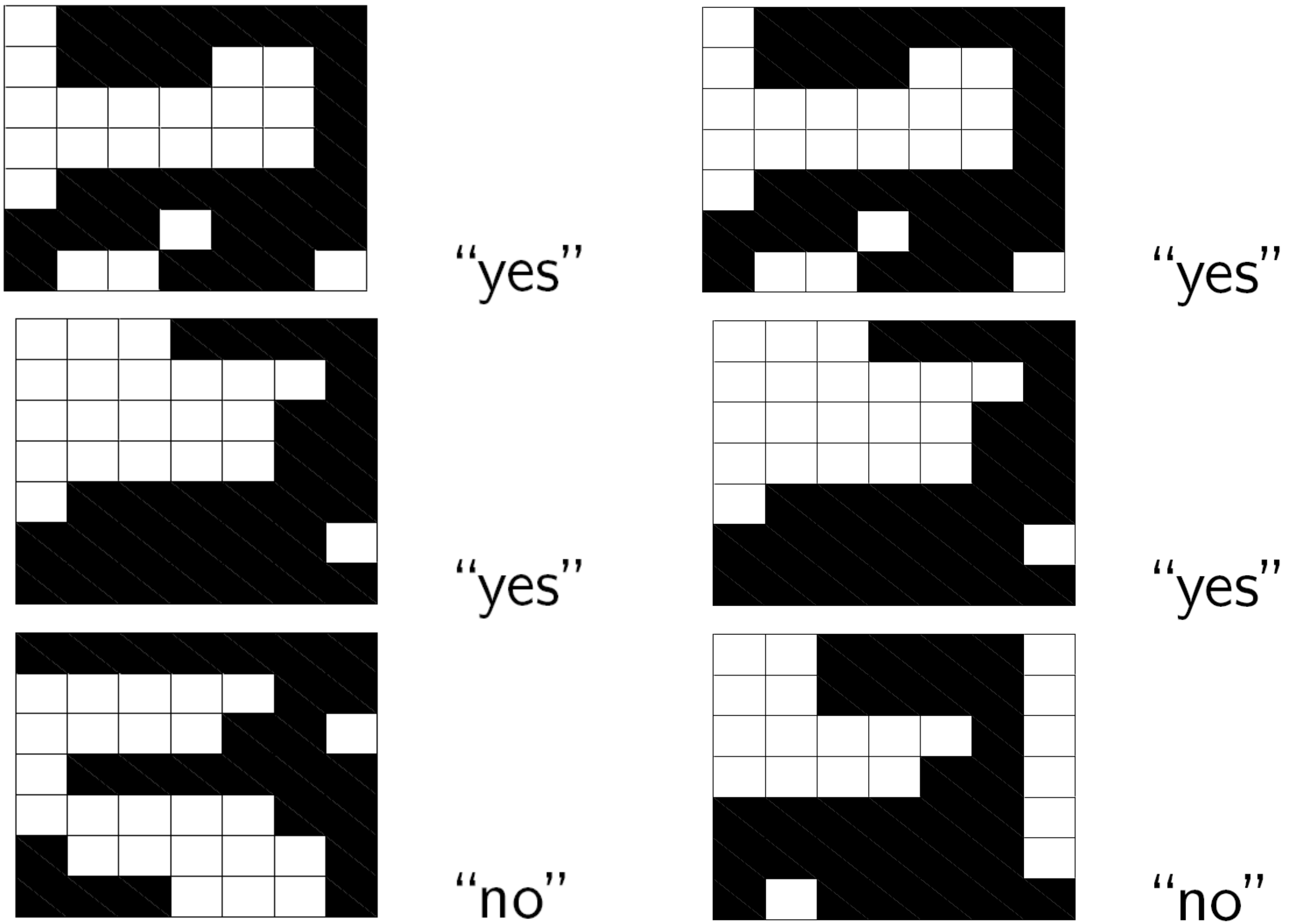
CSE 6363 (Fall 2016)

Lecture 2 Basic Machine Learning

Heng Huang, Ph.D.

Department of Computer Science and Engineering

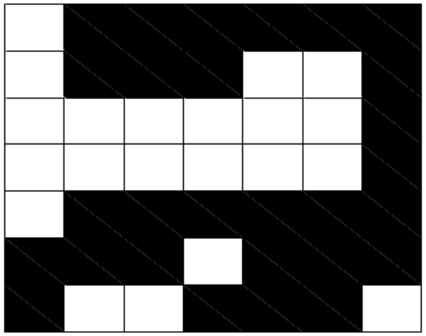
Learning, Biases, Representation



Ref: Tommi Jaakkola

Representation

- There are many ways of presenting the same information



  0111111001110010000000100000001001111110111011111001110111110001

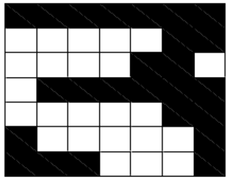
- The choice of representation may determine whether the learning task is very easy or very difficult

Ref: Tommi Jaakkola

Hypothesis Class

- Representation: examples are binary vectors of length $d = 64$

$$\mathbf{x} = [111\ldots0001]^T \;=$$



and labels $y \in \{-1, 1\}$ ("no","yes")

- The mapping from examples to labels is a "linear classifier"

$$\hat{y} = \text{sign}\,(\,\theta \cdot \mathbf{x}\,) = \text{sign}\,(\,\theta_1 x_1 + \ldots + \theta_d x_d\,)$$

where θ is a vector of *parameters* we have to learn from examples.

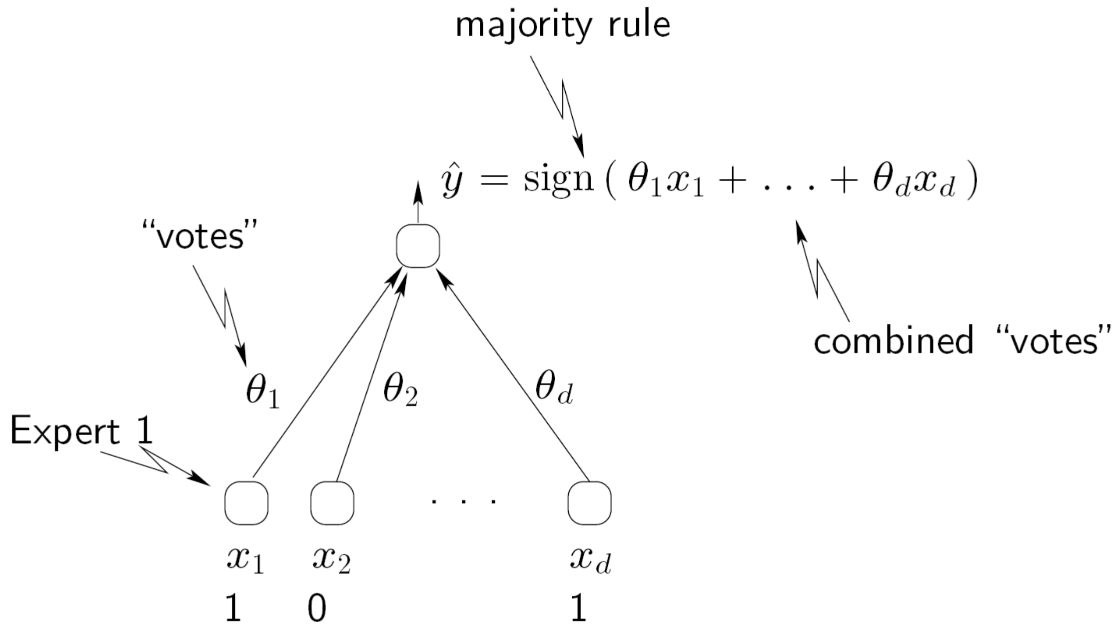
Ref: Tommi Jaakkola

Linear Classifier/Experts

- We can understand the simple linear classifier

$$\hat{y} = \text{sign}\,(\,\theta \cdot \mathbf{x}\,) = \text{sign}\,(\,\theta_1 x_1 + \ldots + \theta_d x_d\,)$$

as a way of combining expert opinion (in this case simple binary features)



Ref: Tommi Jaakkola

Estimation

$\mathbf{x}$	y
0111111001110010000000010000000100111111011101111110011101111110001	+1
0001111100000011000000111000000110011111101111110011111111100000011	+1
1111111000000110000011000111111000000111100000011111000110111111	−1
...	...

- How do we adjust the parameters θ based on the labeled examples?

$$\hat{y} = \text{sign}\left(\theta \cdot \mathbf{x} \right)$$

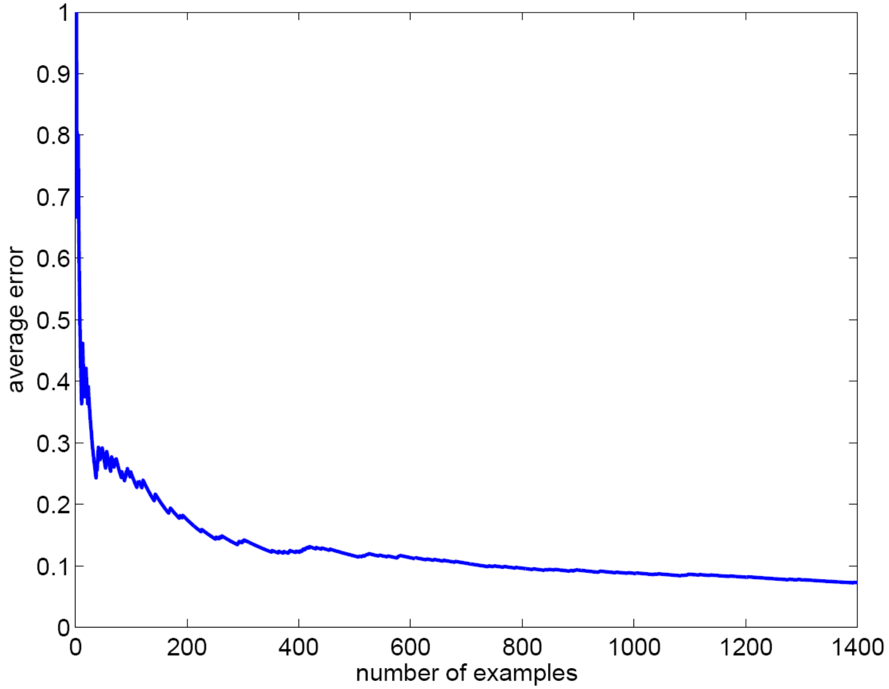
For example, we can simply refine/update the parameters whenever we make a mistake:

$$\theta_i \leftarrow \theta_i + y\, x_i,\ i = 1, \ldots, d \quad \text{if prediction was wrong}$$

Ref: Tommi Jaakkola

Evaluation

- Does the simple mistake driven algorithm work?



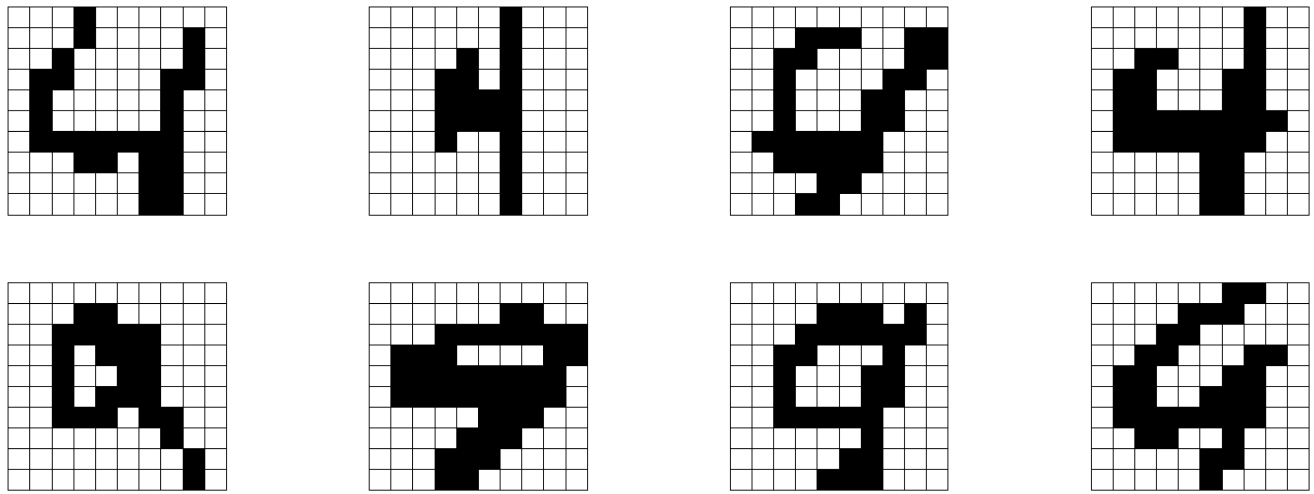
(average classification error as a function of the number of examples and labels seen so far)

Ref: Tommi Jaakkola

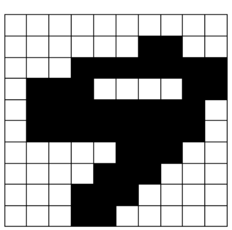
Similar Problem

$y = +1$

$y = -1$



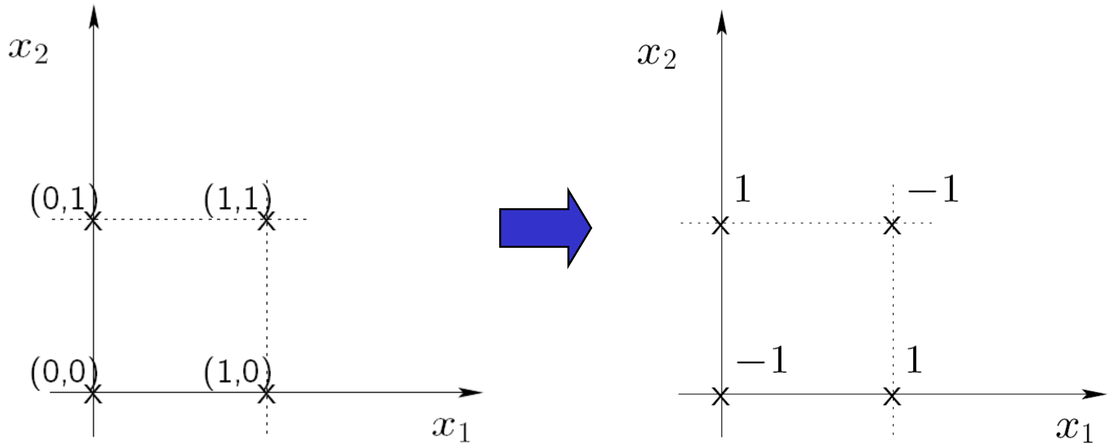
- Representation as a vector:



$$\Rightarrow [0000000000\ 0000001100\ 0001111111\ \ldots\ 0001100000]^T$$

Model Selection

- The simple linear classifier cannot solve all the problems (e.g., XOR)



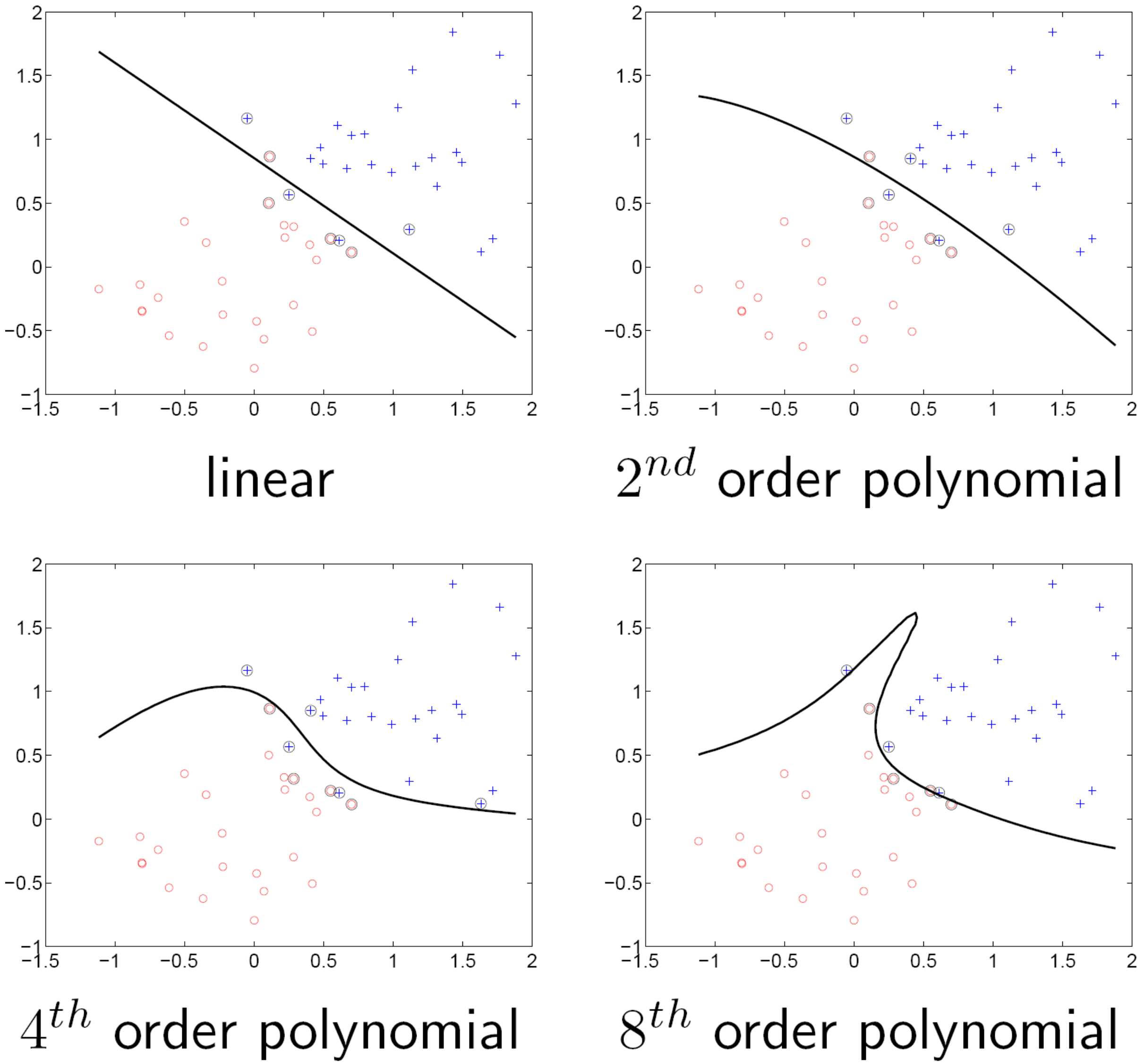
- Can we rethink the approach to do even better?

We can, for example, add "polynomial experts"

$$\hat{y} = \text{sign}\left(\theta_1 x_1 + \ldots + \theta_d x_d + \theta_{12} x_1 x_2 + \ldots \right)$$

Ref: Tommi Jaakkola

Model Selection (cont.)

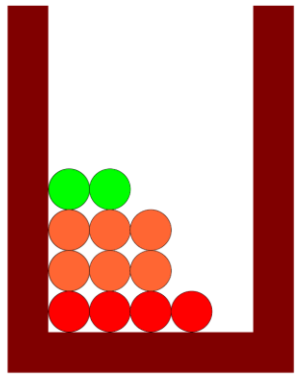


Ref: Tommi Jaakkola

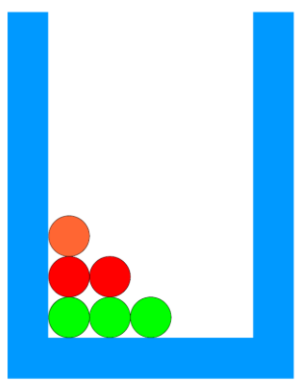
Probability Theory

- Boxes of fruit

apple
orange
strawberry



	apples	oranges	strawberries
red jar	2	6	4

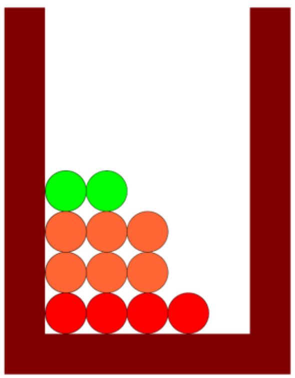


	apples	oranges	strawberries
blue jar	3	1	2

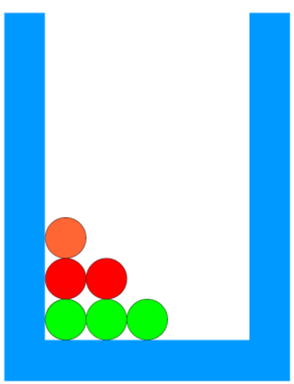
Ref: Chuck Anderson

Probabilities of Fruit from a Given Jar

- apple
- orange
- strawberry



	apples	oranges	strawberries	
red jar	2/12 = 0.167	6/12 = 0.5	4/12 = 0.33	sum = 1.0



	apples	oranges	strawberries	
blue jar	3/6 = 0.5	1/6 = 0.167	2/6 = 0.33	sum = 1.0

Ref: Chuck Anderson

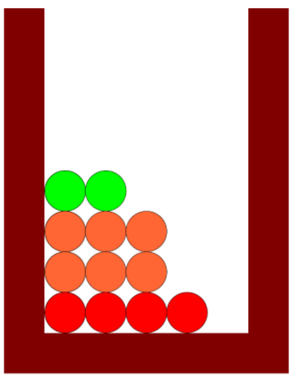
Choose Jar then Draw a Fruit

- apple
- orange
- strawberry

Say the probability of choosing a jar is

P(Jar = red) = 0.6
P(Jar = blue) = 0.4

Prob is 0.6



The probability of choosing the red jar **and** drawing an apple out of it is

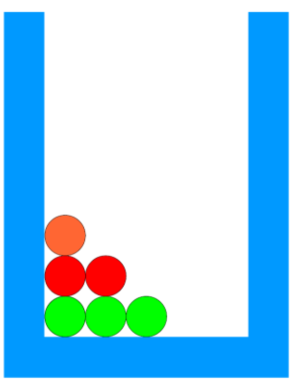
P(Jar=red, Fruit=apple)
= P(Jar=red) P(Fruit=apple|Jar=red)
= 0.6 (0.167) = 0.1

conditional probabiliity

Doing all multiplications results in:

	apples	oranges	strawberries	
red jar (P=0.6)	0.6(0.167) = 0.1	0.6(0.5) = 0.3	0.6(0.33) = 0.2	sum = 0.6

Prob is 0.4



	apples	oranges	strawberries	
blue jar (P=0.4)	0.4(0.5) = 0.2	0.4(0.167) = 0.067	0.4(0.33) = 0.133	sum = 0.4

Joint Probability Table

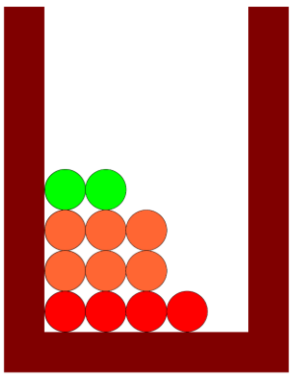
- apple
- orange
- strawberry

Combine in a two-dimensional table to show joint probabilities of two events.

r = red, b = blue

a = apple, o = orange, s = strawberry

Prob is 0.6

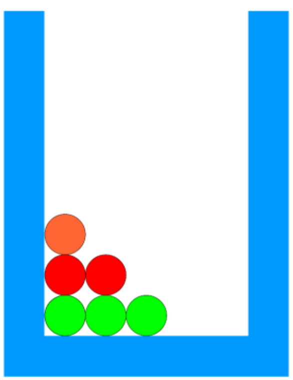


Fruit

Jar	a	o	s	
r	0.1	0.3	0.2	$\Sigma = 0.6$
b	0.2	0.067	0.133	$\Sigma = 0.4$
	$\Sigma = 0.3$	$\Sigma = 0.367$	$\Sigma = 0.333$	$\Sigma = 1.0$

Let J be random variable for Jar, and F be random variable for fruit.

Prob is 0.4



Fruit

Jar	a	o	s	
r	$P(J=r, F=a)$	$P(J=r, F=o)$	$P(J=r, F=s)$	$P(J=r)$
b	$P(J=b, F=a)$	$P(J=b, F=o)$	$P(J=b, F=s)$	$P(J=b)$
	$P(F=a)$	$P(F=o)$	$P(F=s)$	1.0

Ref: Chuck Anderson

Joint Probabilities and Bayes Rule

Just saw example of the *product rule*:

P(Fruit=orange, Jar = blue)
= P(Fruit=orange | Jar = blue) P(Jar = blue)

Since P(Fruit=orange, Jar = blue) = P(Jar = blue, Fruit = orange),

P(Jar = blue, Fruit=orange)
= P(Jar = blue | Fruit = orange) P(Fruit = orange).

Setting these equal leads to *Bayes Rule*:

P(Jar = blue | Fruit = orange) P(Fruit = orange)
= P(Fruit=orange | Jar = blue) P(Jar = blue)

so

P(Jar = blue | Fruit = orange)
= P(Fruit=orange | Jar = blue) P(Jar = blue) / P(Fruit = orange)

Ref: Chuck Anderson

Joint Probabilities and Bayes Rule

On the right hand side of Bayes Rule, all terms are given to us except P(Fruit = orange):

$$P(Jar = blue \mid Fruit = orange) = \frac{P(Fruit=orange \mid Jar = blue)\, P(Jar = blue)}{P(Fruit = orange)}$$

We can use the *sum rule* to get this.

$$P(Fruit = orange) = \Sigma_j P(Fruit=orange, Jar=j) = 0.367$$

So, Bayes Rule can be rewritten as

$$P(Jar = blue \mid Fruit = orange)$$

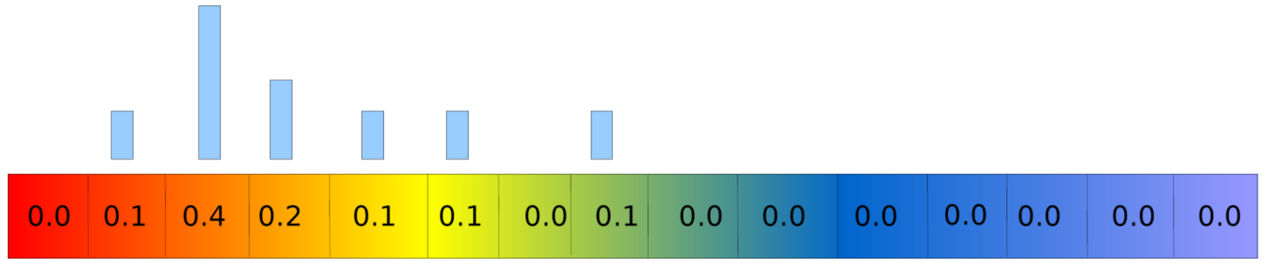
$$= \frac{P(Fruit=orange \mid Jar = blue)\, P(Jar = blue)}{\Sigma_j P(Fruit=orange, Jar=j)} = \frac{P(Fruit=orange \mid Jar = blue)\, P(Jar = blue)}{\Sigma_j P(Fruit=orange \mid Jar=j)\, P(Jar = j)}$$

Ref: Chuck Anderson

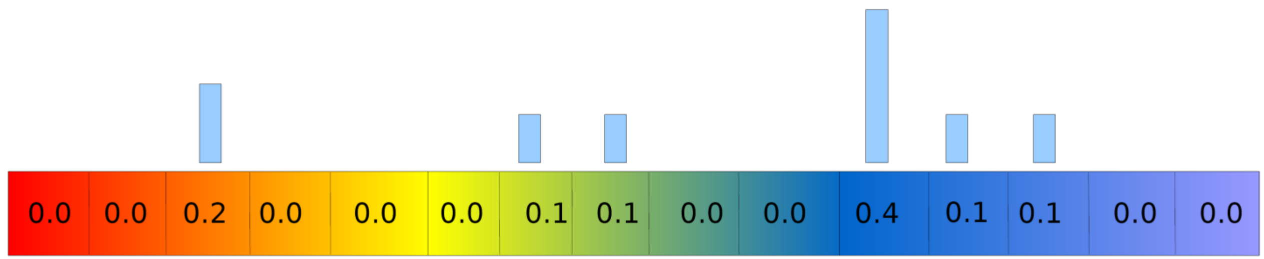
Probability Distributions

- Rather than three colors of fruit, imagine objects of 15 possible colors

Jar 1 contains objects with colors in these proportions



Jar 2 contains objects with colors in these proportions



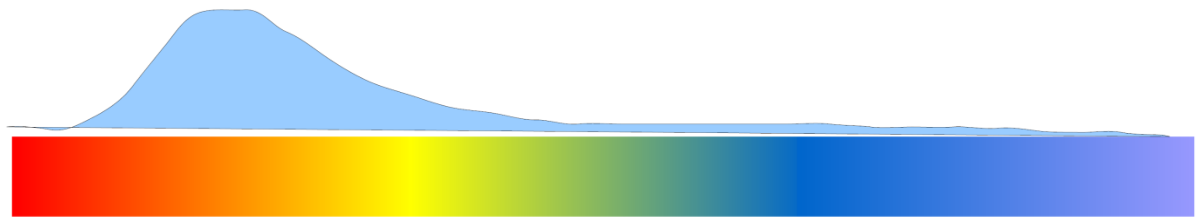
- Can calculate joint probability table as before.
- But what if we have 100 colors or 1000 colors?
- What if we have an infinite number of colors?

Ref: Chuck Anderson

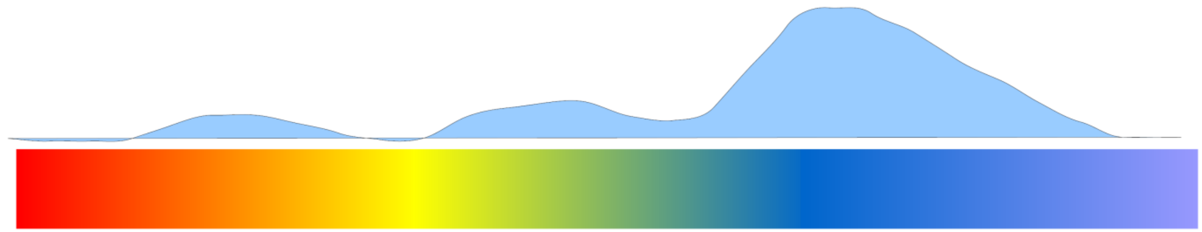
Distributions Over Continuous Values

- Probability of a color is a function over the continuous spectrum.

Jar 1 contains objects with colors in these proportions



Jar 2 contains objects with colors in these proportions



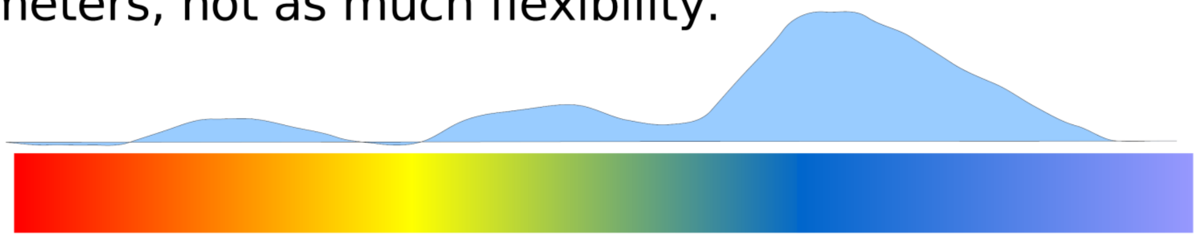
- But what function is this? Would require 1,000s of parameters to specify general function.
- Instead, let's use rather simple functions controlled by a few parameters.
- Common example: Gaussian (Normal) distribution

Ref: Chuck Anderson

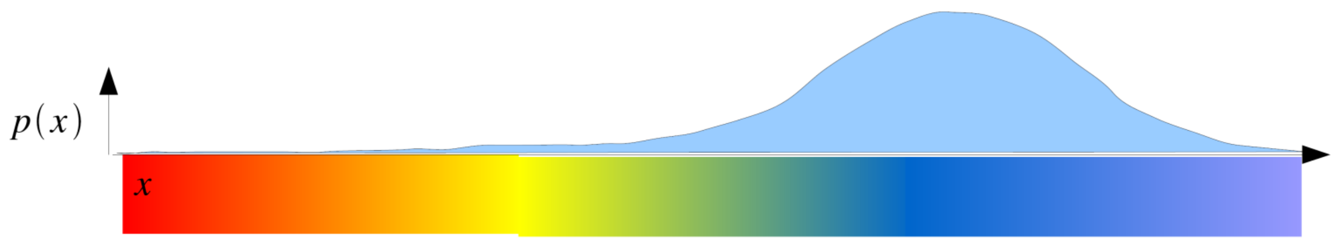
Gaussian Distribution

- With few parameters, not as much flexibility.

Jar 2 distribution



becomes



because

argument parameters

$$p(x;\mu,\sigma)=\frac{1}{(2\pi\sigma^2)^{1/2}}\exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

- Easy to estimate parameters.

$$\mu=\frac{1}{N}\sum_{i=1}^{N}x_i \qquad \sigma=\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^2$$

Ref: Chuck Anderson

Gaussian Distribution

- Where do these expressions come from?

$$\mu = \frac{1}{N} \sum_{i=1}^{N} x_i \qquad \sigma = \frac{1}{N} \sum_{i=1}^{N} (x_i - \mu)^2$$

- Maximize the likelihood of the data.
  - Likelihood of data is product of probabilities of each sample x_i

$$p(X|\mu, \sigma) = \Pi_{i=1}^{N} \left[\frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left(-\frac{1}{2\sigma^2} (x_i - \mu)^2 \right) \right]$$

  - Maximize this by maximizing its logarithm.

$$\ln p(X|\mu, \sigma) = -\frac{1}{2\sigma^2} \sum_{i=1}^{N} (x_i - \mu)^2 - \frac{N}{2} \ln \sigma^2 - \frac{N}{2} \ln(2\pi)$$

  - Set its derivative with respect to μ to zero and solve for μ.
  - Set its derivative with respect to σ to zero and solve for σ.

Ref: Chuck Anderson