On the virtues of the cumulative distribution function
matlab code

function [f,x] = cdist(data)

x=sort(data);
f=[0, 0:(1/length(x)):1];
x=[x(1); x(:); x(end)];
CDFs and sorting

- To Create a CDF you need to sort
- Can knowing the CDF help you sort?
No. of comparisons needed for sorting

• There are $n!$ permutations

• Identifying the correct permutation requires $\log(n!) \text{ comparisons.}$

• $\log(n!) \sim n \log n$

• Merge-sort achieves time $O(n \log n)$
Efficient sorting for uniform distribution

- $n$ Elements drawn IID from uniform distribution over $[0,1]$
- $A = \text{array}[1:n] \text{ of lists}$
- Given $x$ insert it into list at $A[\text{floor}(x \cdot n)]$
- Lists would rarely have $> 1$ element
- Time: $O(n)$
Efficient sorting using the CDF
Efficient sorting for an IID source

- Source is IID but distribution is unknown
- Sort first $m$ instances to estimate CDF
- Use CDF to sort rest of data.
- How large should $m$ be?
Stability of empirical CDF

250 instances

1000 instances
The difference between two empirical CDFs

% input: samples d1,d2
n = length(d1) % assuming length(d1)==length(d2)
vals = [d1; d2];
labels = [repmat(1,n,1); repmat(-1,n,1)];

[s,i] = sort(vals); % sort labels by
s_labels = labels(i); % increasing vals

d = cumsum(s_labels);
plot(0:length(d),[0;d]);
Typical cumulative difference
Typical cumulative difference
Maximal divergence of a returning random walk
The Glivenko-Cantelli Theorem

\[ F(x) = \text{CDF} \]

\[ F_n(x) = \text{empirical CDF using } n \text{ random instances} \]

\[ \epsilon = \text{error tolerance} \]

\[ x^n = \langle x_1, x_2, \ldots, x_n \rangle = \text{sample} \]

\[ P_{x_1^n} \left( \max_x |F_n(x) - F(x)| \geq \epsilon \right) \leq a e^{-b n \epsilon^2} \]

Elegant proof.
Devroye, Gyorfi, Lugosi / A probabilistic Theory of Pattern Recognition, page 192

Best Constants: \( a=b=2 \), Complex proof.

Empirical test of the Glivenko-Cantelli Theorem

250 instances

1000 instances
The Kolmogorov-Smirnoff Test

the maximal difference between the two empirical CDFs
KS for hue distributions
Conclusions

• Computing CDFs ~ Sorting.

• CDFs are very stable for all distributions

• Proof: Glivenko Cantelli - related to properties of random walks.

• KS test is a powerful test for n>1000

• When there is systematic variation - KS might be too sensitive.

• For colors of fruit - we need to estimate parameters of distribution - Use distribution model.