The WWW: ~100 billion webpages, we want to store links and content (e.g. on our laptop).

**Problem:** We can’t store it all!

**Solution:** Convert to a mathematical model (i.e. set of integers)

~10,000 #/set

~100 billion sets

*Are two webpages (almost) identical?*

**Definition:** Resemblance = \( \frac{|A \cap B|}{|A \cup B|} \)

**Problem:** Can’t store every set A (same problem as before)

**Solution:** Store fragments

i.e. store a sample of the whole set

A = - x - x x - - x - - - - x -

B = - - x - - x x - - - - x - -

*Which integers do you store?*

- sort and store smallest 10%?

**Problem:** smallest 10% might be very similar for all pages (e.g. “Made with XYZ Software”)

**Solution:** do one-time random permutation and set that as the sort order

**Sequences**

*What if we’re interested in sequences instead of sets?*

Convert sequence to set by considering all sequences of length k for some small value of k (i.e. sliding window of size k).

**Note:** the terms subsequence and ‘shingle’ will be used interchangeably (see http://en.wikipedia.org/wiki/W-shingling).
e.g. sequence of length: 10,000 = 10⁴
alphabet of size 100
k = 3

**How many successor shingles are there?**
# shingles = |alphabet|^3 = 10⁶

**What is the probability that a shingle is in the set?**
\[
\frac{\text{# windows}}{\text{# shingles}} = \frac{1}{\text{# singles}} = \frac{10^4}{10^6} = \frac{1}{100}
\]

**What is the probability that you don’t find >1 successor shingle?**
\[
\left(1 - \frac{1}{100}\right)^{100} = \frac{1}{e} = 0.3
\]
Unfortunately, this means 0.7 probability that you do…

If k = 4:
\[
\left(1 - \frac{1}{10,000}\right)^{100} = 0.99
\]

**Alternative method for finding the number of distinct elements**

x₁, x₂, …, xₙ

n = | sequence |
m = | alphabet |

Let S be a subset of size d, selected uniformly at random from \{1, 2, …, n\}
Let min be the smallest element of S.

**How many distinct elements are in S?**

**Claim:** Knowing min, we can calculate d.

\[
\text{min} = \frac{n}{d + 1}
\]

\[
d = \frac{n}{\text{min}}
\]

\{1, 2, …, m\} \rightarrow_{\text{hash}} \{1, 2, …, M = m^2\} (i.e. birthday problem)
i.e. \{h(x₁), h(x₂), …\}
**Problem:** we’re not really selected uniformly at random (some elements are more likely than others).

*How far are we?*

**Lemma:** With probability at least $\frac{2}{3}$:

$$\frac{d}{6} \leq \frac{m_{\min}}{6d} \leq 6d$$

**Proof:** \(\Pr \left[ \frac{M_{\min}}{\min} > 6d \right] < \frac{1}{6}\)

Define indicator variables for \(k = 1, 2, \ldots, d\) where

\[
Z_k = \begin{cases} 
1 & \text{if } h(a_k) < \frac{m}{6d} \\
0 & \text{otherwise}
\end{cases}
\]

\(Z = \sum_{k=1}^{d} Z_k\)

\[
\Pr \left[ \frac{M_{\min}}{\min} > 6d \right] = \Pr \left[ \min < \frac{M}{6d} \right] = \Pr \left[ \exists k : h(x_k) < \frac{M}{6d} \right] = \Pr [Z \geq 1]
\]

= \Pr [Z \geq 6E(Z)] \leq \frac{1}{6} \text{ by Markov inequality}

where \(E(Z_k) = \frac{1}{6d}\), \(E(Z) = \frac{1}{6}\) and \(\Pr [h(a_k) < \frac{M}{6d}] = \frac{1}{6d}\)

To get \(\frac{M}{\min} \leq 6d\), use a similar proof but use \(\forall\) and Chebyshev’s inequality instead of \(\exists\) and Markov.