Estimating m(w)

- $m(w)$ for an equidistant algorithm
  - For every address $a \in A$, assume periodic access every d_a time units
  - Let $H_a(w)$ = 1 if the address a is accessed during a time window of size w
    - $H_a(w)$ = 0 otherwise
    - This is a random variable depending on the placement of the window
  - The expected value of $H_a(w)$ is:
    - $\mathbf{E}\big(H_a(w)\big) = \min\left(\frac{w}{d_a}, 1\right)$
  - Let $N(w) = \sum_{a \in A} H_a(w)$, i.e. the number of different addresses accessed
  - $m(w)$ is just the average of $N(w)$ across all window placements
    - $m(w) = \mathbf{E}\big(N(w)\big) = \sum_{a \in A} \mathbf{E}(H_a(w)) = \sum_{a \in A} \min\left(\frac{w}{d_a}, 1\right)$

Estimating m(w)

- $m(w)$ in general
  - The intervals between adjacent accesses to the same address may vary
  - The d_a is, in general, a random variable dependent on window placement
  - The correct general formula for the expected value of $H_a(w)$ is:
    - $\mathbf{E}\big(H_a(w)\big) = \frac{\mathbf{E}(\min(w,d_a))}{\mathbf{E}(d_a)}$
      - Based on the fact that wide d_a is encountered more frequently
    - $m(w) = \mathbf{E}\big(N(w)\big) = \sum_{a\in A} \mathbf{E}(H_a(w)) = \sum_{a\in A} \frac{\mathbf{E}(\min(w,d_a))}{\mathbf{E}(d_a)}$
  - Note: If the random variables $H_a(w)$ are independent for different $a \in A$
    - This is not a realistic assumption for most algorithms, but it still works here
    - Then, for large $|A|$, $N(w)$ can be approximated by a normal distribution (by CLT)
      - The variance will be relatively low, $\sigma^2 \leq |A|/4$, i.e. the std. dev. $\sigma \leq \sqrt{|A|}/2$
      - This observation will soon be useful...

Estimating the frequency of cache misses

- Estimating number of cache misses
  - C – the size of the cache
  - For an access to an address $b \in A$
    - assuming the previous access is at the distance d_b
    - the address b will be evicted and thus a cache miss will occur if $N(d_b) \geq C$
      - $N(d_b)$ is a random variable dependent on the position of the access
      - However, due to the narrow variance of $N(w)$, the formula $N(d_b) \geq C$...
      - ... may be simplified to $m(d_b) \geq C$, which is still random due to d_b
  - The total frequency of cache misses (wrt. unit of time) is then estimated as
    - $X(C) = \sum_{b \in A} \frac{\mathbf{P}(m(d_b) \geq C)}{\mathbf{E}(d_b)}$
      - the $\mathbf{E}(d_b)$ factor accounts for the frequency of memory accesses to b

Estimating the frequency of cache misses

- Computing $X(C)$ from $m(w)$
  - Trick: Compute the derivative of $m(w)$:
    - $\frac{\partial}{\partial w} m(w) = \sum_{a \in A} \frac{\frac{\partial}{\partial w}\mathbf{E}(\min(w, d_a))}{\mathbf{E}(d_a)} = \sum_{a \in A} \frac{\mathbf{P}(w \le d_a)}{\mathbf{E}(d_a)}$
  - $m(d_b)$ is increasing (except when equal to $|A|$)
    - therefore $w \le d_a$ is equivalent to $m(w) \le m(d_a)$
  - Combined:
    - $\frac{\partial}{\partial w} m(w) = \sum_{a \in A} \frac{\mathbf{P}(m(w) \le m(d_a))}{\mathbf{E}(d_a)}$
  - This is similar to the definition of $X(C)$:
    - $X(C) = \sum_{b \in A} \frac{\mathbf{P}(m(d_b) \ge C)}{\mathbf{E}(d_b)}$
      - with the substitution $C = m(w)$
  - Finally:
    - $X(C) = \frac{\partial m(w)}{\partial w}(m^{-1}(C))$
    - This is only an approximative formula
      - not applicable for small $C \ll \sqrt{|A|}$