Occam Algorithms

- Let $X$ be a domain of examples and $C, H$ concept classes over $X$.
- **Theorem 0:** Let $A$ be a learning algorithm such that $\forall c \in C$ $A$ takes a sample of $m$ examples and outputs a hypothesis $h \in H$ consistent with $S$
  Then when using a sample of size $m \geq \frac{1}{\epsilon} \ln \frac{|H|}{\delta}$
  $A$ is a PAC learning algorithm for $C$ using $H$.

- Definition: For $\alpha \geq 0$ and $0 \leq \beta < 1$
  Algorithm $A$ is an $(\alpha, \beta)$-Occam algorithm for learning $C$ by $H$
  if on any sample $S \subset X_n$ of $m$ examples labeled by $c \in C_n$
  $A$ outputs $h \in H_n$ consistent with $S$ and such that
  $\text{size}(h) \leq [n \cdot \text{size}(c)]^\alpha m^\beta$
- In the above, $\text{size}(x)$ measures the number of bits used to represent $x$ in the respective representation language.
- **Theorem 2:** If $A$ is an $(\alpha, \beta)$-Occam algorithm for learning $C$ by $H$,
  Then when using a sample of size $m \geq \max \left\{ \frac{2}{\epsilon} \ln \frac{1}{\delta} \left( \frac{2[n \cdot \text{size}(c)]^\alpha m^\beta}{\epsilon} \right) \right\}$
  $A$ is a PAC learning algorithm for $C$ using $H$.

- **Theorem 3 (VC Dimension):** Let $C$ be a concept class with $\text{VC}D(C) = d$ then:
  1. for $m \leq d$ we have
     $\pi_C(m) = 2^d$
  2. for $m > d$ we have
     $\pi_C(m) \leq \left( \frac{\epsilon m}{d} \right)^d = O(m^d)$
  2. for an algorithm that takes IID sample and returns a consistent hypothesis in $C$:
     for $m \geq \max \left\{ \frac{6}{\epsilon} \ln \frac{2}{\delta}, \frac{8d}{\epsilon} \ln \frac{30}{\delta} \right\}$ we have
     $\Pr[\text{err} > \epsilon] < 2\pi_C(2m) \left( \frac{1}{2} \right)^{\epsilon m} < \delta$
  3. if an algorithm PAC learns $C$ then it must use
     $m = \Omega \left( \frac{d}{\epsilon} \ln \frac{1}{\delta} \right)$