Notes 16: Neural networks

What is the VC dimension of a neural network?

Define neural network $N$ as directed acyclic graph $G$ with LTFs at internal nodes

$G$ specifies the network architecture and is fixed

$G$ has $n$ input nodes $1, \ldots, n$ and $s$ internal nodes $v_1, \ldots, v_s$

Input nodes (those without incoming edges) receive input signals $x_1, \ldots, x_n \in \mathbb{R}$

Node/neuron $v$ is internal if it has at least one incoming edge

Internal neuron $v$ computes a linear threshold function on its predecessor neurons

$$x_v = 1 \left( \sum_{u \in \text{Pred}(v)} w_{uv} \cdot x_u \geq \theta_v \right)$$

where $\text{Pred}(v) = \{ \text{predecessors of } v \}$

$v$ is activated (i.e. $x_v = 1$) if the weighted sum of incoming signals exceeds threshold $\theta_v$

When $G$ has a single output node (that has no outgoing edges)

the network $N$ computes a function $f_N : \mathbb{R}^n \to \{0, 1\}$ (given $w_{uv}$ and $\theta_v$)

If learning algorithm $A$ searches for weights and thresholds to minimize training error

$A$’s hypothesis class is $\mathcal{H}_N = \{ f_N \mid w_{uv} \in \mathbb{R}, \theta_v \in \mathbb{R} \}$

$\text{VCDim}(\mathcal{H}_N) \leq ?$

Will answer this question for a more general class of neural networks:

Redefine neural network $N$ as directed acyclic graph $G$ with concept classes at internal nodes

$\mathcal{C}_j$ over $\mathbb{R}^{\text{Pred}(v_j)}$ is the concept class at internal node $v_j$

Internal neuron $v_j$ computes $x_{v_j} = 1 \left( x_{\text{Pred}(v_j)} \in c_j \right)$ for some $c_j \in \mathcal{C}_j$

Original definition has $\mathcal{C}_j = \{ \text{LTFs} \}$ for all $v_j$: New definition allows other activation functions

Hypothesis class $\mathcal{H}_N = \{ f_N \mid c_j \in \mathcal{C}_j \}$ (now $f_N : \mathbb{R}^n \to \{0, 1\}$ implicitly depends on $c_j$’s)

Theorem 1. Growth function of $\mathcal{H}_N$ is at most the product of growth functions of $\mathcal{C}_j$ over internal nodes $v_1, \ldots, v_s$ of $G$,

$$\Pi_{\mathcal{H}_N}(m) \leq \Pi_{\mathcal{C}_1}(m) \cdots \Pi_{\mathcal{C}_s}(m)$$

for all $m \in \mathbb{N}$

Proof. Order internal nodes $v_1, \ldots, v_s$ by the order they get evaluated (i.e. topological order)

e.g. in above diagram, $v_4$ comes after $v_1, \ldots, v_3$ because $x_{v_4}$ depends on $x_{v_1}, \ldots, x_{v_3}$

Fix $m$ input samples $S = \{ x^1, \ldots, x^m \}$ where every $x^i \in \mathbb{R}^n$

How many different labelings/dichotomies $T \in \Pi_{\mathcal{H}_N}(S)$ are induced as $c_j \in \mathcal{C}_j$ vary?

Imagine choosing $c_1, \ldots, c_s$ sequentially and suppose $c_1, \ldots, c_{j-1}$ have been fixed

For every $u \in \text{Pred}(v_j)$, the function $f_u : \mathbb{R}^n \to \mathbb{R}$ of the subnetwork ending at $u$ is fixed

Every sample $x^i$ yields a vector $(f_u(x^i))_{u \in \text{Pred}(v_j)}$ of evaluations of these functions

Call this vector $f_{\text{Pred}(v_j)}(x^i)$; It belongs to $\mathbb{R}^{\text{Pred}(v_j)}$

Collection of these vectors $S_j = \{ f_{\text{Pred}(v_j)}(x^i) \mid x^i \in S \}$ has size $\leq m$

Varying $c_j$ may induce different dichotomies $T_j \in \Pi_{\mathcal{C}_j}(S_j)$ on $S_j$

Choosing all $c_1, \ldots, c_s$ yields a labeling $T$ of $S$, together with a sequence $(T_1, \ldots, T_s)$ as above

Distinct labelings $T$ and $T'$ must correspond to different sequences $(T_1, \ldots, T_s)$ and $(T'_1, \ldots, T'_s)$

Because a sequence $(T_1, \ldots, T_s)$ contains enough information to recover $T$

via computing $f_{v_j}(x^i) = 1 \left( f_{\text{Pred}(v_j)}(x^i) \in T_j \right)$ iteratively for $j = 1, \ldots, s$

Every $T_j$ is induced by $c_j \in \mathcal{C}_j$ on $S_j$ of size $\leq m$ $\implies$ At most $\Pi_{\mathcal{C}_j}(m) \cdots \Pi_{\mathcal{C}_s}(m)$ sequences

Corollary 2. If $\text{VCDim}(\mathcal{C}_j) \leq d$ for all $1 \leq j \leq s$, then $\text{VCDim}(\mathcal{H}_N) \leq 2ds \log(es)$ when $s \geq 2$
Proof. By above Theorem and Sauer–Shelah lemma, when \( m \geq d \),

\[
\Pi_{H_N}(m) \leq \Pi_{C_1}(m) \cdots \Pi_{C_s}(m) \leq \left( \frac{em}{d} \right)^d \cdot \cdots \cdot \left( \frac{em}{d} \right)^d
\]

VCDim(\( H_N \)) < m \iff \Pi_{H_N}(m) < 2^m$, so we want \( \left( \frac{em}{d} \right)^d > 2^m \iff ds \log \left( \frac{em}{d} \right) < m \)

How to choose \( m \)?

Clearly \( m \geq ds \) is needed, but then \( \log(\frac{em}{d}) \geq \log(es) \), so \( m \geq ds \log(es) \)

Turns out \( m = 2ds \log(es) \) suffices when \( s \geq 2 \) (exercise)

Back to original question, if \( G \) has fan-in \( r \) (i.e. every internal node takes signals from \( r \) other nodes)

\[
\text{VCDim}((\text{LTFs over } \mathbb{R}^r)) = r + 1 \iff \text{VCDim}(H_N) \leq 2(r + 1)s \log(es)
\]

Neural networks in practice typically have internal nodes with real-valued outputs, not just \{0, 1\}

Above Theorem does not apply to these networks

The end of Notes15 considers

\[
H_R = \left\{ \text{sign} \left( \sum_{1 \leq t \leq R} \alpha_t h_t \right) \mid \alpha_t \in \mathbb{R}, h_t \in H \text{ for } 1 \leq t \leq R \right\}
\]

where \( H \) denotes the hypothesis class of weak learner \( A \) in AdaBoost

Proposition in Notes15 can be proved using above Theorem and calculations in above Corollary

Question: Which neural network corresponds to \( H_R \)? What are the \( C_j \)’s?