Lecture 1: VC Dimension

Week 1 - Part II

Lecturer: Tianyu Wang

1 The Task of Classification

1.1 The confusion matrix

Suppose we need to approximate $f : \mathbb{R}^d \to \{0, 1\}$ from dataset $\{(x_i, y_i)\}_{i=1}^n$. Suppose we have an approximate, called $\widehat{f}$. How good is $\widehat{f}$? One standard metric is to use the confusion matrix. The confusion matrix is a 2×2 table arranged as in Table 1.

		Approximated label (by $\widehat{f}$)	
		1 (Positive)	0 (Negative)
True label (by f)	1 (Positive)	True Posotive	False Negative
	0 (Negative)	False Positive	True Negative

Table 1: Confusion matrix. With respect to a dataset $\{(x_i, y_i)\}_{i=1}^n$, the True Positive equals $\sum_{i=1}^n \mathbb{I}_{[f(x_i)=1]}\mathbb{I}_{\left[\widehat{f}(x_i)=1\right]}$; the False Negative equals $\sum_{i=1}^n \mathbb{I}_{[f(x_i)=1]}\mathbb{I}_{\left[\widehat{f}(x_i)=0\right]}$; the False Positive equals $\sum_{i=1}^n \mathbb{I}_{[f(x_i)=0]}\mathbb{I}_{\left[\widehat{f}(x_i)=1\right]}$; the True Negative equals $\sum_{i=1}^n \mathbb{I}_{[f(x_i)=0]}\mathbb{I}_{\left[\widehat{f}(x_i)=0\right]}$.

The more true positive and true negative you have compared to the false positive and false negative, the better you are fitting the data. But is a perfect confusion matrix good enough?

1.2 Overfitting, train/validation/test data split and generalization

If a model memorizes the dataset, it can produce the perfect confusion matrix (no false positive or false negative). However, the model is not "learning" in this case, since it may not be able to "generalize".

Consider fitting/training a model $\widehat{f}$ on a dataset $\{(x_i, y_i)\}_{i=1}^n$ $i.i.d.$ sampled from an unknown distribution. Suppose the model $\widehat{f}$ perfectly fits $\{(x_i, y_i)\}_{i=1}^n$. If you give the model another dataset following the same distribution as $\{(x_i, y_i)\}_{i=1}^n$, this model may not be able to generalize to this new dataset and can drastically fail on this new dataset. The phenomenon of large discrepancy between the model's performance on the $\{(x_i, y_i)\}_{i=1}^n$ and that on the new dataset is called *overfitting*.

To prevent overfitting, a common practice is to partition the dataset $\{(x_i, y_i)\}_{i=1}^n$ into several disjoint subsets. Train and tune your model on some of them (called training and validation sets), and test your model on the rest (called test sets).

2 Measuring classification model complexity by VC dimension

Apart from the train/test split, there are many other schemes to prevent overfitting. A rule of thumb is: simpler models are less likely to overfit. But what is a simpler model? One well-recognized method for measuring complexity of you model is the Vapnik–Chervonenkis dimension (VC dimension).

Definition 2.1 (shattering)**.** Suppose $A \subseteq \mathbb{R}^d$ is a finite set, and H is a class of of sets. The class H shatters the set A if for each subset a of A, there exists some element f of H such that

$$a = f \cap A.$$

Definition 2.2 (VC dimension)**.** Let H be a class of sets. The VC dimension D of H is the largest cardinality of sets shattered by H. If arbitrarily large subsets can be shattered, the VC dimension is ∞.

Usually, complex models have large VC dimension and vice versa.

2.1 Example 1: Axis Aligned Rectangles in $\mathbb{R}^2$

Claim 2.3. *Let H be the set of all axis-aligned rectangles in $\mathbb{R}^2$. The VC dimension of H is 4.*

Proof. We will (i) find a set of four points that is shattered by H, and (ii) prove that no set of five (or more) points can be shattered by H. For (i), consider the four point set $\{(1,0),(0,1),(-1,0),(0,-1)\}$. For (ii), note that no rectangle can intersect the four "extremal" points without intersecting the rest. Here the "extremal" points are points with largest/smallest x-axis/y-axis values, with ties broken arbitrarily. □

2.2 Example 2: the Set of All Half-spaces of $\mathbb{R}^d$

A half-space of $\mathbb{R}^d$ is the set of the form $\{x \in \mathbb{R}^d, w^\top x + b \geq 0\}$ or $\{x \in \mathbb{R}^d, w^\top x + b > 0\}$ for some $w, b \in \mathbb{R}^d$.

Claim 2.4. *Let H be the set of all half-spaces of $\mathbb{R}^d$. The VC dimension of H is $d+1$.*

Proof. We will (i) find a set of $d+1$ points that is shattered by H, and (ii) prove that no set of $d+2$ (or more) points can be shattered by H.

For (i), consider the set $A = \{\mathbf{0}, \boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_d\}$, where $\mathbf{0}$ is the origin, and $\boldsymbol{e}_i$ is the point with one on the i-th coordinate and zeros on all other coordinates. Let a be an arbitrary subset of A. If a is $\{\boldsymbol{e}_{j_1}, \boldsymbol{e}_{j_2}, \cdots, \boldsymbol{e}_{j_k}\}$ for some $k \leq d$, then we pick $h \in \boldsymbol{H}$ so that $h := \{x \in \mathbb{R}^d, \sum_{i=1}^k \boldsymbol{e}_{j_i}^\top x \geq 1/2\}$. If a is $\{\mathbf{0}\} \cup \left(A \setminus \{\mathbf{0}\} \setminus \{\boldsymbol{e}_{j_1}, \boldsymbol{e}_{j_2}, \cdots, \boldsymbol{e}_{j_{k'}}\}\right)$ for some $k \leq d$, then we pick $h \in H$ so that $h := \{x \in \mathbb{R}^d, 1/2 - \sum_{i=1}^k \boldsymbol{e}_{j_i}^\top x \geq 0\}$.

For (ii), we show that for any set $A \subseteq \mathbb{R}^d$ of cardinality $d+2$, we can divide A into two disjoint subsets A_1 and A_2 so that the convex hull of A_1 and A_2 intersect. Let $A = \{x_1, x_2, \cdots, x_{d+2}\}$, and consider the system of equations in $\lambda_1, \lambda_2, \cdots, \lambda_{d+2}$ defined by

$$\sum_{i=1}^{d+2} \lambda_i x_i = 0, \quad \text{and} \quad \sum_{i=1}^{d+2} \lambda_i = 0. \tag{1}$$

There are $d+2$ unknowns and $d+1$ equations (d from $\sum_{i=1}^{d+2} \lambda_i x_i = 0$ and 1 from $\sum_{i=1}^{d+2} \lambda_i = 0$). Thus the system of equations (1) must have at least one non-zero solution. Let λ_i^* be a non-zero solution, then we have, for some index set I and J such that $I \cup J = \{1, 2, \cdots, d+2\}$, the convex hulls of $A_1 = \{x_i \in A, i \in I\}$ and that of $A_2 = \{x_i \in A, i \in J\}$ both contain the point $p = \frac{\sum_{i \in I} \lambda_i^* x_i}{\sum_{i \in I} \lambda_i^*}$. [1]

□

3 VC Dimension of Classifiers and back to the Task of Classification

So far, the VC dimension is defined in a set-theoretical sense. To connect the above definition to classifiers, one can define the VC dimension of a set of classifiers by the VC dimension of the sets where the classifiers take positive value. More specifically, for a set of classifiers $\mathcal{H} := \{f : \mathbb{R}^d \to \{-1, +1\}\}$, the VC dimension of $\mathcal{H}$ is the VC dimension of $\{S \subset \mathbb{R}^d : f(x) = +1 \text{ iff } x \in S \text{ for some } f \in \mathcal{H}\}$.

In machine learning tasks, one usually first specify a set of classifiers $\mathcal{H}$ and search for a best classifier within $\mathcal{H}$. The (pre-specified) set of classifier $\mathcal{H}$ is called the *hypothesis class*. As discussed previously, $\mathcal{H}$ can be the set of all linear classifiers, neural networks classifiers (of a certain structure), ...

If VC dimension of the hypothesis class is large, the model is more likely to overfit the dataset. [2]

3.1 The Growth Function

Given $x_1, \cdots, x_n$ and a hypothesis class $\mathcal{H}$, we define

$$\mathcal{H}_{x_1, \cdots, x_n} = \{f(x_1), \cdots, f(x_n) : f \in \mathcal{H}\},$$

which is the set of ways the data $x_1, \cdots, x_n$ are classified by functions from $\mathcal{H}$. Since the functions f can only take two values, this set will always be finite, no matter how big $\mathcal{H}$ is.

[1] Without loss of generality, we assume $\sum_{i \in I} \lambda_i^* > 0$.

[2] The model complexity for deep neural networks is more involved, and is out of the scope of this course.

Definition 3.1 (Growth Function)**.** The growth function of a hypothesis class $\mathcal{H}$ is the maximum number of ways into which m points can be classified by functions in $\mathcal{H}$. Formally, the growth function of $\mathcal{H}$ is $S_{\mathcal{H}} : \mathbb{N}_+ \to \mathbb{N}_+$, so that for any $m \in \mathbb{N}_+$,

$$S_{\mathcal{H}}(m) = \max_{x_1, x_2, \cdots, x_m} |\mathcal{H}_{x_1, \cdots, x_n}|$$

Acknowledgement

A recommended general reference is the machine learning textbook by Kevin Murphy. TW largely used lecture notes by Cynthia Rudin to compile this notes. The proof of the second example is due to Radon. The VC dimension is due to Vapnik and Chervonenkis, as self-explained by its name. Also, a thank you to wikipedia contributors.