

## Lecture 10: Overfitting and Concentration

*Lecturer: Matthew Wicker*

### 1 Learning Objectives

In our previous lecture, we covered Bayesian learning and in doing so completed a relatively comprehensive view of learning in linear models.<sup>1</sup> In the next lectures, we will turn from learning models to evaluating models that we have already learned. At the end of last lecture we briefly covered the marginal likelihood as a principled Bayesian approach to model comparison; however, for the next few lectures we will focus on the mathematical tools developed for evaluating and validating frequentist models, though the bounds and theorems we prove are general tools that apply to any probability distribution. While we will focus on using these tools to reason about model performance over an unknown probability distribution, there is no need to restrict use of the developed theorems and inequalities to a frequentist analysis.

### 2 Universal Function Approximation\*

*NOT IN EXAM*

As a starting point, we have seen in our first lectures that we can choose a basis expansions such as  $\phi(x) := [x, 1]^\top$  to model affine functions or  $\phi'(x) := [x^2, x, 1]^\top$  to model quadratic functions. But how do we choose one or the other? One tempting but flawed approach is to consider the model that fits your data the best. One reason this approach is flawed is that increasing the complexity of the models we consider will almost always lead to a decrease in the loss (interpreted as a better fit). In fact, it is the case that no matter what function our data is drawn from, we can always pick a polynomial basis function such that the function fits that data. To make this more formal, we introduce the first of two universal approximation theorems:

**Definition 2.1. Weierstrass Theorem** Let  $f$  be a continuous function on a closed interval  $[a, b]$ . For any  $\varepsilon > 0$ , there exists a polynomial function  $P(x)$  such that

$$\|f - P\|_\infty = \sup_{x \in [a, b]} |f(x) - P(x)| < \varepsilon.$$

In other words, the polynomial  $P(x)$  can uniformly approximate  $f$  on the interval  $[a, b]$ .

---

<sup>1</sup>A potential next step is for interested students to look into generalized linear models.

The Weierstrass theorem seems to be a compelling reason to select high-order polynomial basis expansion models: we can fit any function we want! But, in fact, the Weierstrass theorem should also give us some pause when picking such models. Consider again the case where we are fitting noisy observations, the high-order polynomial will fit to any such noise which can lead to highly unintuitive model behavior. See, for example, the different linear models given in our slides. Before discussing how to deal with the issue of fitting noise, let us spend a little more time discussing the interesting and powerful universal approximation theorem we have just seen. If polynomial basis expansion is so mathematically powerful, then why is it the case that in practice we use neural networks and other sophisticated models, do they also have similar universal function approximating behavior? Indeed they do, let us recall a classical result for MLPs that we have seen in this class:

**Definition 2.2. Universal Approximation Theorem (Cybenko, 1989)** Let  $\sigma(x)$  be a sigmoidal activation function, i.e., a function that is nonconstant, bounded, and continuously differentiable. For any continuous function  $f$  on a compact subset of  $\mathbb{R}^n$ , and any  $\varepsilon > 0$ , there exist positive integers  $N$ , weights  $w_i$ , and biases  $b_i$ , and a sum of  $N$  sigmoidal functions such that:

$$F(x) = \sum_{i=1}^N w_i \sigma(w_i^T x + b_i)$$

satisfies  $|F(x) - f(x)| < \varepsilon$  for all  $x$  in the compact subset.

This is a very similar result to the the Weierstrass theorem we just observed and we call such results universal function approximation results as they tell us that a particular model class can approximate any continuous function up to arbitrary precision. And, it turns out, that many models are universal function approximators. A general result about universal approximation can be given:

**Theorem 2.3** (Stone-Weierstrass Theorem (limited version)). *Let  $F$  be a class of functions defined on a compact set  $S \subseteq \mathbb{R}^d$ . If  $F$  satisfies:*

1. *Each  $f \in F$  is continuous.*
2. *For every  $x$ , there exists  $f \in F$  such that  $f(x) \neq 0$ .*
3. *For every  $x, x_1$  with  $x \neq x_1$ , there exists  $f \in F$  such that  $f(x) \neq f(x_1)$  ( $F$  separates points).*
4.  *$F$  is closed under multiplication ( $\forall f, g \in F, \exists h \in F$  such that  $h(x) = f(x)g(x)$ ) and vector space operations ( $F$  is an algebra).*


*Then, for every continuous function  $g : \mathbb{R}^d \rightarrow \mathbb{R}$  and any  $\epsilon > 0$ , there exists  $f \in F$  such that  $\|f - g\|_\infty \leq \epsilon$ . In other words,  $F$  is a universal approximator.*

## 2.1 No-Free Lunch Theorems

So, returning to a previous point: why is it that we often use deep learning in practice if polynomial basis expansion and neural networks are both universal function approximators? Here we do not provide a technical treatment, but simply provide a high-level intuition. We often pick neural networks or other models because they are easy to optimize for the given problem. That is, they have low sample complexity (require few examples) or low computational complexity. In large part, characterizing why this is the case is an open research problem. However, a series of formal mathematical results known as *no free lunch* theorems have been developed. These results can be summarized as saying that given two models that perfectly fit our data (e.g., a polynomial and a neural network) we have no a priori reason to choose one over the other because it is impossible to concretely say which will perform best on data that is unlike the data the model was trained on. In the remainder of the next few lectures, we will instead be focused on cases where we would like to compare models on data that are similar to the data they have seen at training insofar as the data we compare the model with are from the same data (joint) distribution.

## 3 Test Sets

To compare two models, a sensible idea would be to put both models through a trial deployment and measure how well each model does. To do so, we typically model how *well* a model performs using a risk function  $R : \mathbb{R}^{m \times m} \rightarrow \mathbb{R}$  which takes in two outputs (the desired output and the model predicted output) and returns how bad the prediction is with larger values indicating worse errors. We can then consider how a model would perform after letting it make  $N$  predictions:


$$\hat{R}_N(f^\theta) := \frac{1}{N} \sum_{i=1}^N R(f^\theta(\mathbf{x}^{(i)}), \mathbf{y})$$

where above we have abused notation and overloaded  $R$  to take a classifier  $f^\theta$ . We could also consider another model  $g^\theta$  and then it stands to reason that if  $\hat{R}_N(f^\theta) < \hat{R}_N(g^\theta)$  we should pick  $f$  over  $g$ . But, how long of a trial  $N$  should we use before we conclusively say that  $f$  is better (or worse) than  $g$ ? In order to understand this question we will need to develop mathematical tools known as concentration inequalities.

## 4 Concentration and Weak Law of Large Numbers

### 4.1 Concentration Inequalities

To reason about the estimators of the above estimators, let's consider an infinitely long trial ( $N = \infty$ ) in this case, we arrive at exactly the expectation of  $R(f^{\theta^*})$ . Stepping away from our testing example, let us consider the expectation of a discrete random variable:

$$\mathbb{E}[Z] = \sum_z z P(Z = z)$$

which is simply the weighted sum of all of the values of  $z$ . Now it is clear that if we consider only part of this weighted sum, then the part of the sum will be less than the whole sum, that is:

$$\sum_z zP(Z = z) \geq \sum_{z > a} zP(Z = z)$$

Further, we can consider taking the latter sum and rather than summing all of the values of  $z$  just summing the lower bound,  $a$  which will yield a lower bound on the latter sum from above:

$$\sum_{z > a} zP(Z = z) \geq \sum_{z > a} aP(Z = z)$$

with these two very elementary steps we have actually defined a useful and powerful inequality:

$$\mathbb{E}[Z] \geq aP(Z > a) \quad \text{or} \quad P(Z > a) \leq \frac{\mathbb{E}[Z]}{a}$$

known as Markov's inequality. The key thing to observe about the above inequality is that it relates the expectation of a random variable to probabilities about the value of the random value. You could use the above inequality to say that if the expectation of  $Z$  is small then so is the probability that  $Z$  is large. The above reasoning can also be extended to any random variable continuous or discrete. For example, we can use the random variable  $(Z - \mu)^2$  which is simply taking the square of our random variable minus some constant. By replacing  $a$  arbitrarily with  $a^2$  the above inequality becomes:

$$\begin{aligned} \mathbb{E}[(Z - \mu)^2] &\geq a^2 P((Z - \mu)^2 \geq a^2) \\ \underbrace{\mathbb{E}[(Z - \mu)^2]}_{\text{Var}(Z) = \sigma^2} &= a^2 P(|Z - \mu| \geq a) \end{aligned}$$

*as before arbitrary const  
∴ don't make a diff*

Now, taking  $\mu$  to be the (finite) mean of the random variable we can see that the expectation we just bounded is simply the definition of the variance of a random variable, and dividing the  $a^2$  to the other side we have:

$$\frac{\sigma^2}{a^2} \geq P(|Z - \mu| \geq a) \quad \mathbb{E}[(Z - \mu)^2] = \text{Var}(Z)$$

A final simplifying step one will often see is selecting  $a = k\sigma$  in which case we have what is known as Chebychev's inequality:

$$\frac{1}{k^2} \geq P(|Z - \mu| \geq k\sigma)$$

An intuitive interpretation of this inequality, similarly to the interpretation of Markov's inequality is that if the variance of a random variable is small then so is the probability of sampling a value far from the mean.

## 4.2 Weak Law of Large Numbers

Okay, so we have now developed some tools for reasoning about the expectation of a random variable in terms of probabilities, but how does this help us in understanding the testing set up we

have for a machine learning model? Well, consider that we can only really think about the above expectations in the infinite trial run case. However, we only have a finite amount of time to trial each model. So how quickly will our risk estimate converge to the true estimate? In other words: how long should we run our trial? In our slides we state and give the intuition of convergence of a random variable, which students ought to look through. In what follows here, we will consider the random variable  $\hat{Z} = \hat{R}_N(f^{theta})$  which is itself a sum of random variables:

$$\hat{Z} = \frac{Z_1 + Z_2 + \dots + Z_N}{N}$$

We therefore might ask, what is the mean of this random variable? By linearity of expectation and under the i.i.d. assumption we have that:

$$\begin{aligned} \mathbb{E}[\hat{Z}] &= \frac{\mathbb{E}[Z_1] + \mathbb{E}[Z_2] + \dots + \mathbb{E}[Z_N]}{N} \\ &= \frac{\mu_1 + \mu_2 + \dots + \mu_N}{N} = \mu \end{aligned}$$

$\text{Var}(\hat{Z}) = \text{Var}\left(\frac{Z_1}{N}\right) + \text{Var}\left(\frac{Z_2}{N}\right) + \dots$   
 $= \frac{1}{N^2} \text{Var}(Z_1) + \frac{1}{N^2} \text{Var}(Z_2) + \dots$   
 $= \frac{1}{N^2} (\sigma^2 + \sigma^2 + \dots)$   
 $= \frac{1}{N^2} N \sigma^2 = \frac{\sigma^2}{N}$

and by similar logic we have that  $\mathbb{V}[\hat{Z}] = \frac{N\sigma^2}{N^2} = \frac{\sigma^2}{N}$ , and finally, by plugging these values into (the equation before the final statement of) Chebychev's inequality and letting  $a = \epsilon$  we get:

$$P(|\hat{Z} - \mu| \geq \epsilon) \leq \frac{\sigma^2}{\epsilon^2} \rightarrow P(|\hat{Z} - \mu| \geq \epsilon) \leq \frac{\mathbb{V}[\hat{Z}]}{\epsilon^2} = \frac{\sigma^2}{N\epsilon^2}$$

Now, this statement, known as the weak law of large numbers, tells us that as  $N$  grows, the probability that our finite trial risk estimate is far away from the true (infinite trial) risk shrinks. At the start of our next lecture we will introduce a more sophisticated concentration inequality and derive an exact length for our trial run.