Double descent in the condition number

Tomaso Poggio¹, Gil Kur¹, Andy Banburski¹

¹Center for Brains, Minds, and Machines, MIT

Abstract

In solving a system of \( n \) linear equations in \( d \) variables \( Ax = b \), the condition number of the \( n \times d \) matrix \( A \) measures how much errors in the data \( b \) affect the solution \( x \). Bounds of this type are important in many inverse problems. An example is machine learning where the key task is to estimate an underlying function from a set of measurements at random points in a high dimensional space and where low sensitivity to error in the data is a requirement for good predictive performance. Here we report the simple observation that when the columns of \( A \) are random vectors, the condition number of \( A \) is highest, that is worse, if \( d = n \), that is when the inverse of \( A \) exists. An overdetermined system (\( n > d \)) and especially an underdetermined system (\( n < d \)), for which the pseudoinverse must be used instead of the inverse, typically have significantly better, that is lower, condition numbers. Thus the condition number of \( A \) plotted as function of \( d \) shows a double descent behavior with a peak at \( d = n \). We discuss implications to recent machine learning developments.
Double descent in the condition number

Tomaso Poggio\textsuperscript{1}, Gill Kur, Andy Banburski

December 26, 2019

In solving a system of $n$ linear equations in $d$ variables $Ax = b$, the condition number of the $n,d$ matrix $A$ measures how much errors in the data $b$ affect the solution $x$. Bounds of this type are important in many inverse problems. An example is machine learning where the key task is to estimate an underlying function from a set of measurements at random points in a high dimensional space and where low sensitivity to error in the data is a requirement for good predictive performance. Here we discuss the simple observation, which is well-known but surprisingly little quoted (see Theorem 4.2 in \cite{1}) that when the columns of $A$ are random vectors, the condition number of $A$ is highest if $d = n$, that is when the inverse of $A$ exists. An overdetermined system ($n > d$) as well as an underdetermined system ($n < d$), for which the pseudoinverse must be used instead of the inverse, typically have significantly better, that is lower, condition numbers. Thus the condition number of $A$ plotted as function of $d$ shows a double descent behavior with a peak at $d = n$.

The concept of condition number was introduced by Turing in 1948 \cite{2} and has since played a key role in the theory of algorithms. The condition number of a function measures how much the output value of the function can change for a small change in the input argument. The condition number most commonly associated with $Ax = b$ is defined as the ratio of the relative error in $x$ to the relative error in the data $b$. In terms of the $l_2$ norm on $x$ and $b$, this leads to the following definition (see Box1) for the condition number of $A$, denoted by $\kappa(A) = \frac{||A|| ||A^\dagger||}{||A^\dagger||}$, where $||A||$ is the operator norm of the $m,n$ matrix $A$ is defined in terms of the vector norm of $K^n, K^m$ as $||A|| = \sup_{x \in K^n, x \neq 0} \frac{||Ax||}{||x||}$ and $A^\dagger$ is the pseudoinverse. It is easy to see that $\kappa(A) = \frac{\sigma_{\text{max}}(A)}{\sigma_{\text{min}}(A)}$ that is the ratio of the maximal and minimal singular values of $A$.

The plot in the Figure 1 can be easily checked by calling the function “cond” in MatLab. The double descent pattern is apparently quite robust to choices of $d$ and $n$, such that their ratio $\gamma = \frac{n}{d}$ is the same. The fact that the worse conditioning occurs when the inverse exists uniquely ($\gamma = 1$) seems at first surprising. This simple observation must have been realized by many. The proof is also simple because of a well-known characterization of the eigenvalues of random matrices \cite{3}. In fact, consider the $n,d$ random matrix $A$. We characterize its condition number by using the Marchenko–Pastur law, which describes the asymptotic behavior of singular values of large rectangular random matrices. We assume that the entries of $A$ are independent,
identically distributed random variables with mean 0 and variance $\sigma^2$. We consider the limit for $n \to \infty$ with $\frac{n}{d} \to \gamma$. Marchenko–Pastur claims that for $\gamma < 1$ the smallest and the largest singular values of $\frac{1}{n}AA^T$ are, respectively $(1 - \sqrt{\gamma})^2$ and $(1 + \sqrt{\gamma})^2$. For $\gamma > 1$ the largest and the smallest eigenvalues of $\frac{1}{n}A^TA$ are $(1 + \sqrt{\gamma - 1})^2$ and $(1 - \sqrt{\gamma - 1})^2$. When $\gamma = 2$, and the entries are i.i.d. sub-Gaussian, the maximal singular value is concentrated around 2, but the minimal one is $\min\{n^{-1}, d^{-1}\}(\max\{\sqrt{n} - \sqrt{d - 1}, \sqrt{d} - \sqrt{n - 1}\})^2$, as was observed in [4].

For the system of linear equations $Ax = b$, the implication is that is better to have more variables than data: the condition number associated with the minimum norm solution $x = A^\dagger b$ is usually much better – that is closer to 1 – than the condition number of a well-determined system with $n=d$, if the matrix $A$ is random (see for instance [5]).

There are interesting observations for machine learning. The most obvious is that kernel methods (see Box1), which are a popular workhorse in machine learning, do not require regularization in order to be well-conditioned, if the kernel matrices are based on high dimensional i.i.d data, especially when $\gamma < 1$. This claim follows from recent results on kernels. The simplest form of the kernel matrix $K(x_j, x_i)$ is $K = XX^T$. We consider random matrices whose entries are $K(x_i^T, x_j)$ with i.i.d. vectors $x_i$ in $\mathbb{R}^p$ with normalized distribution (in Figure [3] we consider a radial kernel $K(||x_i - x_j||^2)$ for which similar arguments are likely to hold). Assuming that $f$ is sufficiently smooth and the distribution of $x_i$’s is sufficiently nice, El Karoui [6] showed that the spectral distributions of kernel dot-product matrices $K(x_i, x_j) = f(XX^T)$ behave as if $f$ is linear in the Marchenko–Pastur limit. In fact, El Karoui showed that under mild conditions, the kernel matrix is asymptotically equivalent to a linear combination of $XX^T$, the all-1’s matrix, and the identity, and hence the limiting spectrum is Marchenko-Pastur. As a consequence, the claims about the condition number of a random matrix $A$ also apply to kernel matrices with random data, see Figure [3].

More intriguing is the fact that the behavior of the condition number of $K^\dagger$ is similar to the double descent behavior of the test error by linear and kernel interpolants, which after initial work by Belkin (2, see also [8]) has recently attracted much attention [9, 10, 7, 11, 12, 13, 14]. It is natural to expect that some measure of stability of the interpolant solution should play a key role in determining the prediction error. We know that in the “classical” regime of fixed hypothesis space and $n \to \infty$, stability, defined as cross-validation leave-one-out ($CV_{\text{loo}}$) error, is important. We expect a similar notion of stability to be required in the “modern” high dimensional regime of $\frac{n}{d} \to \infty$, in which the minimum norm pseudoinverse plays a key role. In both cases, well-posedness, that is existence, uniqueness and especially stability of the solution, are the key requirement for predictivity. Stability is usually guaranteed during minimization of the empirical loss by complexity control under the form of vanishing regularization (as in the definition of the pseudoinverse) or as implicitly provided by iterative gradient descent [13].

Defining $CV_{\text{loo}}$ stability as the difference between the error made by the predictor obtained by using ERM on the training set $S$ vs. the error of the predictor obtained from a slightly perturbed training set $S'$ we can derive a simplified bound for interpolating kernel regressors as proportional to (assuming Lipschitz properties of the loss function):

$$\|\frac{1}{n}k(x, X)K^{-1}Y\| \leq \frac{1}{n}\|k(x, X)\|\|K^{-1}\|\|Y\| = k(x, x)\sigma^{-1}_{\min}Y \leq K^2C\sigma^{-1}_{\min}$$ (1)
where \( k \) and \( C \) do not depend on \( n, d \) of the training set \( S \) but \( \sigma_{\min}^{-1} \) does (when \( \frac{n}{d} < 1 \) then \( \sigma_{\min} = 1 - \sqrt{\frac{2}{d}} \)). Thus \( \beta \leq k^2 C \sigma_{\min}^{-1} \). Using Equation 1.3 in [16], we have with probability \( 1 - \delta \) that the expected error of the algorithm is

\[
R(A) \leq (\sqrt{\beta T} + \frac{T}{\sqrt{n}}) \sqrt{\log(\frac{1}{\delta})}
\]

where \( T \) is a uniform bound on the loss function. Thus we obtain

\[
R(A) \leq (\sqrt{T k^2 C (1 - \frac{n^2}{d})^{-1}} + \frac{T}{\sqrt{n}}) \sqrt{\log(\frac{1}{\delta})}.
\]

It is interesting to observe that \( \beta \) in CVloo stability and \( R(A) \) are controlled by the minimum norm solution for any given \( n, d \). This is true in the modern and also in the classical regime, as it is clear by rewriting the bound on the expected error for the general case of regularized kernel regression with a regularization parameter \( \lambda \):

\[
R(A) - R_{\text{emp}}(A) \leq (\sqrt{T k^2 C ||(K + n\lambda I)^d||} + \frac{T}{\sqrt{n}}) \sqrt{\log(\frac{1}{\delta})}.
\]

For \( d \) fixed, \( \lambda \) has to go to zero slower than \( \frac{1}{n} \) in order for the right-hand side to go to zero. In this regime \( \beta \to 0 \) with \( n \to \infty \) implies uniform convergence and compactness of the hypothesis space. For \( \frac{n}{d} \) fixed and \( n \to \infty \), \( \lambda \) can be set to \( \lambda = 0 \); the minimum expected error then follows from the minimum of \( \beta \), which itself follows from the minimum norm property of the pseudoinverse \( ||K^\dagger|| \). Thus the property of CVloo stability provides the crucial link between (minimum) norm and expected error.

In the modern regime, the expected error \( R(A) \) remains different from the zero empirical error (interpolation) also for \( n \to \infty \). An example is shown in Figure 3, demonstrating that even this simple estimate seems to capture the “double-descent” behavior of the test error.

The condition number of the kernel matrix is not sufficient by itself to accurately estimate the out-of-sample error. An elegant and complex estimate of the test error for kernel interpolators has been recently given [17] adding intriguing details to the basic double descent behavior described by our simple analysis.

The open question is whether a general stability condition such as the CVloo stability of which \( \beta \) stability is a special case, can also provide the expected behavior for the case of overparametrized deep neural networks. It has been shown recently [18, 19, 20] that with the exponential loss, gradient descent induces a dynamics such that the weight matrix for each layer of the network converges to a minimum norm solution.\(^1\)

\(^1\)In the case of a deep network a bound on \( \beta \) is given by \( C ||W_K|| \cdots ||W_1|| \) where \( W_1 \cdots W_K \) are the minimum norm solutions (for margin greater or equal to 1) for each layer weight matrix obtained via gradient descent. Then the expected error – assuming interpolation on the data – is \( R(A) \leq (\sqrt{T C ||W_K|| \cdots ||W_1||} + \frac{T}{\sqrt{n}}) \sqrt{\log(\frac{1}{\delta})} \), where the bound is expected to be loose.
Figure 1: Typical double descent of the condition number (y axis) of a random data matrix distributed as $\mathcal{N}(0, 1)$: the condition number is worse when $n = d$, better if $n > d$ (on the right of $n = d$) and also better if $n < d$ (on the left of $n = d$).
Figure 2: Typical double descent of the condition number (y axis) of a radial basis function kernel $K(x, x') = \exp\left(-\frac{||x-x'||^2}{2\sigma^2}\right)$ built from a random data matrix distributed as $\mathcal{N}(0, 1)$: as in the linear case, the condition number is worse when $n = d$, better if $n > d$ (on the right of $n = d$) and also better if $n < d$ (on the left of $n = d$). The parameter $\sigma$ was chosen to be 5.
Figure 3: Plot of $||K(x, X)|| ||K(X, X)^\dagger||$, where $x$ is an out-of-sample data point (not in $X$) and $K$ is a radial basis function kernel $K(x, x') = \exp\left(-\frac{||x-x'||^2}{2\sigma^2}\right)$, plotted with $\sigma = 8$. Both the dataset $X$ and the out-of-sample point were generated from the $\mathcal{N}(0,1)$ distribution.

**Acknowledgments** We thank Gil Strang and Felipe Cucker and Lorenzo Rosasco and especially Silvia Valle. This material is based upon work supported by the Center for Minds, Brains and Machines (CBMM), funded by NSF STC award CCF-1231216, and part by C-BRIC, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA. This research was also sponsored by grants from the National Science Foundation (NSF-0640097, NSF-0827427), and AFSOR-THRL (FA8650-05-C-7262).

**Competing Interests** The authors declare that they have no competing financial interests.

**Correspondence** Correspondence and requests for materials should be addressed to T. Poggio (email: tp@ai.mit.edu).

**Author Contribution** All developed the basic theory.
Boxes
Consider “learning the function $f$ from data $S = \{x_1, y_1; x_2, y_2; \ldots; x_n, y_n\}$ by computing

$$\min_{f \in B_R} \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2. \quad (5)$$

We assume that $f(x) = \sum_{i=1}^{n} c_i K(x_i, x)$ and that $f$ is in the ball $B_R$ of radius $R$ in $\mathcal{H}$ (e.g. $\|f\|_K \leq R$). Then $\mathcal{H} = I_K(B_R)$ is compact – where $I_K : \mathcal{H}_K \rightarrow C(X)$ is the inclusion and $C(X)$ is the space of continuous functions with the sup norm [21]. In this case the minimizer of the generalization error $I[f]$ is well-posed. Minimization of the empirical risk (Equation (5)) is also well-posed: it provides a set of linear equations to compute the coefficients $c$ of the solution $f$ as

$$Kc = y \quad (6)$$

where $y = (y_1, \ldots, y_n)$ and $(K)_{i,j} = K(x_i, x_j)$. Notice that this last set of linear equations is well-posed even without the constraint $\|f\|_K \leq R$: if $K$ is symmetric and positive definite and the $x_i$ are distinct the $K^{-1}$ exists and $\|f\|_K^2$ is automatically bounded, with a bound that increase with $n$. For any fixed $n$ the condition number of $K$ is finite. A regularized form of ERM is

$$\min_{f \in H} \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|_K^2, \quad (7)$$

which gives the following set of equations for $c$ (with $\lambda \geq 0$)

$$(K + n \lambda I)c = y, \quad (8)$$

which for $\lambda = 0$ reduces to Equation (6). In both cases, stability of the empirical risk minimizer provided by Equation (7) can be characterized using the classical notion of condition number of the problem. The change in the solution $f$ due to a variation in the data $y$ can be bounded as

$$\|\Delta f\| \leq \|K + n \lambda I\| (K + n \lambda I)^{-1} \|\Delta y\|,$$

where the condition number $\|K + n \lambda I\| (K + n \lambda I)^{-1}$ is controlled by $n \lambda$. A large value of $n \lambda$ gives condition numbers close to 1, whereas ill-conditioning may result if $\lambda = 0$ and the ratio of the largest to the smallest eigenvalue of $K$ is large. Though this was the classical argument, it is now clear (because of recent results such as El Karoui [6]) that random $K$ matrices are typically well-conditioned even for $\lambda = 0$. In other words, for i.i.d high-dimensional data,

$$\frac{\|\Delta f\|}{\|f\|} \leq \|K\| \|(K)^\dagger\| \|\Delta y\| \|y\|^{-1}, \quad (9)$$

and the condition number $\kappa(K) = \|K\| \|(K)^\dagger\|$ is close to 1. Bounding the test error under the typical settings of machine learning is more complicated. The test error for $x \in S$ is zero (interpolation). We would like to bound the test error for a new $x$ in terms of $k(x, X)$ and $Y$ and assumptions on the range of $y$. Let us call $\|e\| = \|f(x) - f^*(x)\|$ in the case of noiseless training data $y$. We are interested in how much worse can thi test error be if the training data are affected by noise $\Delta y$. Bounding $\|\Delta f(x)\| = \|k(x, X)K^{-1}\Delta y\|$ is difficult in general because both the first and the second term depend on the i.i.d. training set. It seems clear however that the bound should depend critically on the condition number of $K$. A careful and difficult calculations to estimate the bias and variance of the estimator $k(x, X)K^{-1}\Delta y$ has been carried out in [17].

---

**Box 1: ERM and Kernel Machines**

Consider “learning the function $f$ from data $S = \{x_1, y_1; x_2, y_2; \ldots; x_n, y_n\}$ by computing
Box2: Classical Learning Theory

In the classical setting, a key property of a learning algorithm is generalization: the empirical error must converge to the expected error when the number of examples $n$ increases to infinity, while the class of functions $\mathcal{H}$, called the hypothesis space, is kept fixed. An algorithm that guarantees good generalization will predict well, if its empirical error on the training set is small. Empirical risk minimization (ERM) on $\mathcal{H}$ represents perhaps the most natural class of learning algorithms: the algorithm selects a function $f \in \mathcal{H}$ that minimizes the empirical error – as measured on the training set.

One of the main achievements of the classical theory was a complete characterization of the necessary and sufficient conditions for generalization of ERM, and for its consistency (consistency requires asymptotic convergence of the expected risk to the minimum risk achievable by functions in $\mathcal{H}$; for ERM generalization is equivalent to asymptotic consistency). It turns out that consistency of ERM is equivalent to a precise property of the hypothesis space: $\mathcal{H}$ has to be a uniform Glivenko-Cantelli (uGC) class of functions.

Later work showed that an apparently separate requirement – the well-posedness of ERM – is in fact equivalent to consistency of ERM. Well-posedness usually means existence, uniqueness and stability of the solution. The critical condition is stability of the solution. Stability is equivalent to some notion of continuity of the learning map (induced by ERM) that maps training sets into the space of solutions, eg $L : Z^n \to \mathcal{H}$. In particular, it was proved $^{22, 23}$ that CV$_{\text{loo}}$ stability guarantees generalization and in the case of ERM is in fact equivalent to consistency.

We recall the definition of leave-one-out cross-validation (in short, CV$_{\text{loo}}$) stability:

$$\forall i \in \{1, \ldots, n\} \ P_S \{|V(f_S, z_i) - V(f_{S_i}, z_i)| \leq \beta_{\text{CV}}\} \geq 1 - \delta_{\text{CV}},$$

where $V(f, z)$ is a loss function that is Lipschitz and bounded for the range of its arguments and $z = ((x, y))$. CV$_{\text{loo}}$ stability measures the difference between the errors at a point $z_i$ when it is in the training set $S$ of $f_S$ wrt when it is not. The definition of CV$_{\text{loo}}$ was introduced to deal with general situations in which $\mathcal{H}$ may not have a norm. The definition of leave-one-out stability is simpler in the framework of inverse problems when $\mathcal{H}$ is a RKHS and the noise in the data can be assumed to affect only the “outputs” $y_i$. Then a condition number can be defined: a good condition number close to 1 implies then good CV$_{\text{loo}}$ stability. Both definitions capture the basic idea of stability of a well-posed problem: the function “learned” from a training set should, with high probability, change little in its pointwise predictions for a small change in the training set, such as deletion of one of the examples or label noise affecting some of the training data.

In the modern regime, in which both $n$ and $d$ (or number of parameters) grow to infinity, generalization is not expected. The classical approach – of asymptotic generalization and then consistency – cannot be used because there is no fixed hypothesis space. However, the requirement of well-posedness and stability remains. We would expect that the appropriate definitions of stability for the “modern” regime should involve the condition number. An appropriate definition should provides a bound on the test error wrt to perturbations in the data, similarly to the definition of CV$_{\text{loo}}$ stability.
References


