MA 591 - HOMEWORK 03

Hayden Outlaw

NCSU Applied Mathematics, 17 Oct 2025

---

1 JAX: Training of Physics Informed Neural Networks

Consider the 1D Burger's equation:

$$\begin{cases} u_t + uu_x - \nu u_{xx} = 0, & (x,t) \in (-1,1) \times (0,1] \\ u(x,0) = -\sin(\pi x) & x \in (-1,1) \\ u(-1,t) = u(1,t) = 0 & t \in (0,1] \end{cases}$$

This is the 'viscous' Burger's equation, with viscosity $\nu = \frac{0.01}{\pi}$. The goal is to use a physics-informed neural network (PINN) to solve this system.

Across all models, we fix:

- The network architecture as depth 9, and width 20 - note all layers will be the same width, and not vary on the size of the training dataset
- All network layers as the tanh activation function
- Optimizer algorithm as the ADAM algorithm with a learning rate of $4 \cdot 10^{-4}$ for $16,000$ iterations - once we are reasonably close to a local minimum on the optimization surface, then the L-BFGS algorithm for 4000 iterations
- Let N_{re}, N_{bd}, N_{ic} be the number of points for the interior, boundary (left and right ends), and initial conditions. We fix $N_{bc} = 200, N_{ic} = 100$.

The goal is primarily to observe the effect of the sampling locations, or collocation points, in the quality of the physics-informed loss functions, specifically as with respect to the discrete L^2 error. We search 9 possible configurations:

- For sampling the interior, we examine a deterministic grid mesh, a uniform random sampling scheme, and a Latin hypercube sampling (LHS) scheme
- We examine three different sample sizes of the interior point sample sizes: $N_{re} = 2000, 4000, 6000$

The models were trained on a M1 Macbook Pro, using the JAX programming language, along with the Equinox NN module. Training each model took < 30 seconds locally, with < 5 minutes required to train all 9 model configurations. Implementation details and source code are available on Github at: *https://github.com/outlawhayden/ncsu-sciml/tree/main/hw3*

First, we can compare the loss trajectories across all model configurations, which are depicted in Figure 1. The switch between training algorithms is visible at $16,000$ iterations, where using a more locally-optimized algorithm (LBFGS) results in a smoother loss function. The LHS sampling scheme introduced large spikes in error in the training, more so than any other mechanism, but only in the ADAM regime (< 16000 iterations).

More visually illustrative is a rendering of the minimum loss across training epochs, which is shown in Figure 2. Here, it is more obvious that at least initially, the LHS sampler is much less efficient in the most sparse setting. Beyond that, past 5000 iterations all of the samplers seem to produce empirically equivalent model accuracy.

Based on these experiments, we then evaluate overall model suitability using the sampling configuration using a uniform random sampling scheme, and $N_{re} = 6000$ collocation points - achieving a minimum discrete L2 loss of $3.82 \cdot 10^{-5}$ across the domain.

Figure 1: Loss across Model Configurations

Figure 2: Minimum Loss across Model Configurations

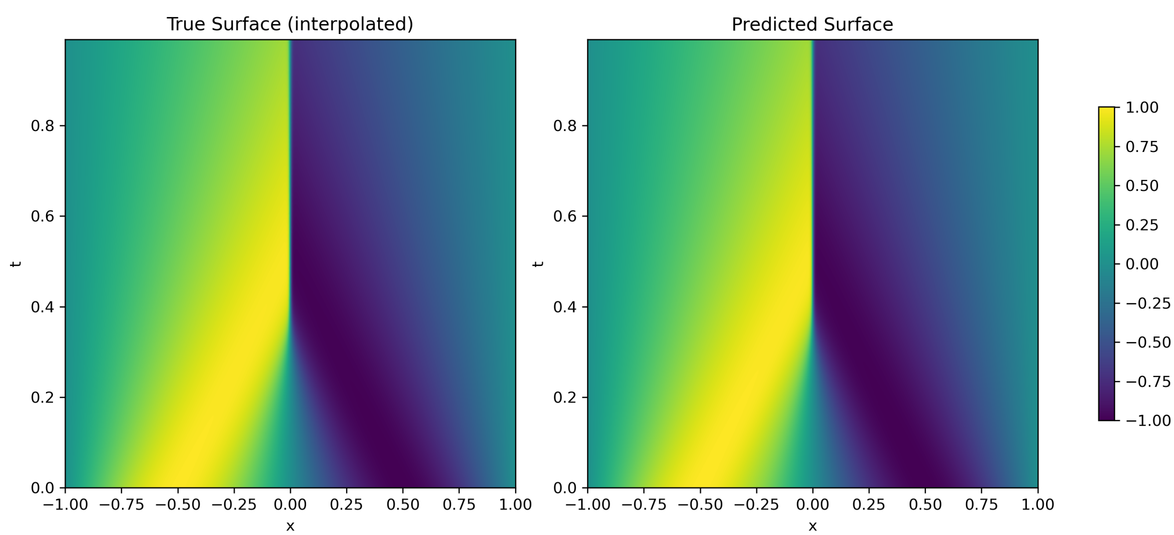

Figure 3: Prediction vs Ground Truth

Using this trained model, we can compare the predicted solution surface against the known true surface, which is shown in Figure 3[1]. The predicted solution visually matches the known interpolated solution, including the proper scale of the resulting wave, and a sufficiently fine delineation of the shockwave in the middle of the domain, where the problem is the most stiff. A visual examination of the errors in Figure 4 shows that for most of the domain, the error is well within suitable levels - the one

[1]Note: Using cubic interpolation to generate a surface

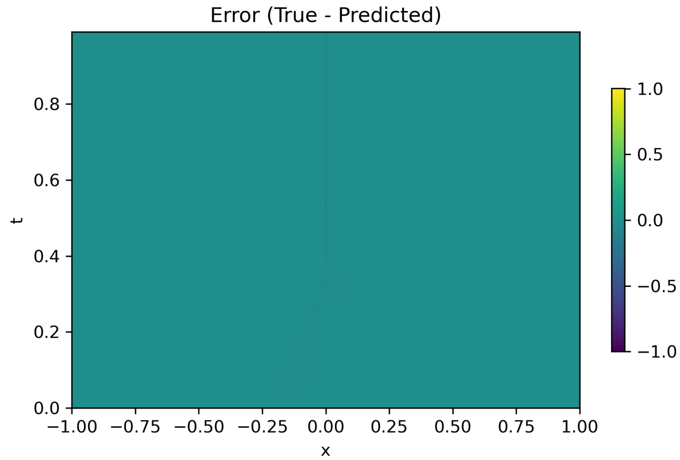

Figure 4: Prediction Error

area where there are still faint inaccuracies is on the shockwave down the middle of the domain, which is expected behavior.

2 L-Lipschitz Neural Network Models

2.1 ReLU is 1-Lipschitz

Consider the ReLU activation function: $\phi(x) = \max\{0, x\}$. We can show that ϕ is 1-Lipschitz. Given $x, y \in \mathbb{R}$:

$$\begin{aligned} |\phi(x) - \phi(y)| &= |\max\{x, 0\} - \max\{y, 0\}| \\ &\leq |x - y| \\ &= 1|x - y| \end{aligned}$$

so ϕ is Lipschitz on $\mathbb{R}$ with Lipschitz constant 1.

2.2 Lipschitz NN Models are Equicontinuous Class

Consider a class of two-layer neural networks with L-Lipschitz activation ϕ, and bounded weights and biases, i.e.,

$$\mathcal{NN}_n^\phi(B) = \left\{ \sum_{i=1}^n c_i \phi(w_i x + b_i) + c_0 \middle| |c_0|, |c_i|, |w_i|, |b_i| \leq B \; \forall \; i \right\}$$

We can show that this model class is equicontinuous, i.e., for any ε, there exists $\delta > 0$ such that:

$$|v(x) - v(y)| < \varepsilon \;\; \forall \;\; x, y \text{ such that } |x - y| < \delta \;\; \forall \; v \in \mathcal{NN}_n^\phi(B)$$

Let $v \in \mathcal{NN}_n^\phi(B)$. For $\varepsilon > 0$:

$$
\begin{aligned}
|v(x)-v(y)| &= \left|\sum_{i=1}^{n} c_i\phi(w_i x+b_i)+c_0-\left(\sum_{i=1}^{n} c_i\phi(w_i y+b_i)+c_0\right)\right| \\
&= \left|\sum_{i=1}^{n} c_i\phi(w_i x+b_i)-\sum_{i=1}^{n} c_i\phi(w_i y+b_i)\right| \\
&= \left|\sum_{i=1}^{n} \left(c_i\phi(w_i x+b_i)-c_i\phi(w_i y+b_i)\right)\right| \\
&= \left|\sum_{i=1}^{n} c_i\left(\phi(w_i x+b_i)-\phi(w_i y+b_i)\right)\right| \\
&\leq \left|B\sum_{i=1}^{n} \left(\phi(w_i x+b_i)-\phi(w_i y+b_i)\right)\right| \quad (c_i \text{ bounded}) \\
&\leq B\sum_{i=1}^{n} \left|\phi(w_i x+b_i)-\phi(w_i y+b_i)\right| \\
&\leq B\sum_{i=1}^{n} L\left|w_i(x)+b_i-w_i(y)-b_i\right| \quad (\phi \text{ is } L\text{-Lipschitz}) \\
&= BL\sum_{i=1}^{n} |w_i(x-y)| \\
&\leq BL\sum_{i=1}^{n} B|(x-y)| \quad (w_i \text{ bounded}) \\
&= (B^2Ln)|x-y|
\end{aligned}
$$

So, we have that $|v(x)-v(y)| \leq (B^2Ln)|x-y|$. Therefore, if we choose $\delta = \frac{\varepsilon}{B^2Ln}$, $|x-y| \leq \delta = \frac{\varepsilon}{B^2Ln}$ implies that:

$$
\begin{aligned}
&|v(x)-v(y)| \leq (B^2Ln)|x-y| \implies \\
&|v(x)-v(y) \leq (B^2Ln)\frac{\varepsilon}{B^2Ln} \implies \\
&|v(x)-v(y)| \leq \varepsilon
\end{aligned}
$$

Therefore, since δ is parameterized exclusively by ε and $\mathcal{NN}_n^{\phi}(B)$ (ie by ϕ, n, B), we can conclude $\mathcal{NN}_n^{\phi}$ is an equicontinous set of functions.

2.3 L-Lipschitz Function on Partition of $[0,1]$

Let $f : [0,1] \to \mathbb{R}$ be L-Lipschitz. For a positive integer N, let $h = \frac{1}{N+1}$ and consider a partition $\{I_i\}_{i=1}^{N}$ of $[0,1]$ (ie $I_i = [(i-1)h, ih)$ for $i = 1, \dots, N$ and $I_{N+1} = [Nh, 1]$). Let p_N be the piecewise constant such that for all i,$p_N(x) = f((i-1h))$ for $x \in I_i$. We can show that:

$$\sup_{x\in[0,1]} |p_N(x) - f(x)| \leq Lh$$

$p_N(x)$ is f evaluated at the left boundary of whichever partition x is in on the interval. For any $x \in [0,1]$, let $x \in I_i$ - we know it must be in exactly one subinterval, since I_i partitions $[0,1]$. The problem can be reduced to the distance to the nearest left boundary of whatever subinterval x is in. In other words, the corresponding $p_N(x) = f((i-1)h)$, and:

$$\sup_{x\in[0,1]} |p_N(x) - f(x)| = \sup_{x\in I_i} |f((i-1)h) - f(x)|$$

Since f is L-Lipschitz, we know:

$$
\begin{aligned}
\sup_{x\in I_i} |f((i-1)h) - f(x)| &\leq \sup_{x\in I_i} L|(i-1)h - x| \\
&= L\left[\sup_{x\in I_i} |(i-1)h - x|\right]
\end{aligned}
$$

Since $x \in I_i = [(i-1h), ih)$, the supremum of this distance is at $x = ih$[2], but then $\sup_{x \in I_i} |(i-1)h - x| = |(i-1)h - ih| = h$. And so we can conclude:

$$\sup_{x \in [0,1]} |p_N(x) - f(x)| \leq Lh$$

3 Analysis of Physics Informed Neural Networks on ODE

Consider the ODE: $\begin{cases} x'(x) = -\pi \sin(\pi x), x \in (0,1) \\ u(0) = 1 \end{cases}$ with known analytical solution $u(x) = \cos(\pi x)$. Given a class of bounded two-layer ReLU networks:

$$\mathcal{NN}_n^{\text{ReLU}}(B) = \left\{ \sum_{i=1}^{n} c_i \phi(x - b_i) + 1 \middle| 0 < |c_i|, b_i \leq B \; \forall i \right\}$$

consider the PINN problem:

$$\min_{u \in \mathcal{NN}_n^{\text{ReLU}}(B)} \mathcal{L}_m(u) = \frac{1}{m} \sum_{i=1}^{m} \Big(u'(x_i) + \pi sin(\pi x_i) \Big)^2 + (u(0) - 1)^2$$

where $x_i = ih$, and $h = \frac{1}{m+1}$, ie an even grid. Note that $\mathcal{L}_m(u)$ is dependent on m, the number of collocation points.

3.1 Consistent Initial Condition

First, we can show that $\mathcal{NN}_n^{\text{ReLU}}(B)$ contains models consistent with the initial condition. More specifically, for any $v \in \mathcal{NN}_n^{\text{ReLU}}(B)$:

$$\begin{aligned} v(0) &= \sum_{i=1}^{n} c_i \phi(0 - b_i) + 1 \\ &= \sum_{i=1}^{n} c_i \max\{0, -b_i\} + 1 \\ &= \sum_{i=1}^{n} c_i (0) + 1 \\ &= 1 \\ &= u(0) \end{aligned}$$

3.2 Existence of Optimal Model

We can show that there exists $B > 0$ such that for any m, there is an n (potentially dependent on N) and $u_m^* \in \mathcal{NN}_n^{\text{ReLU}}(B)$, such that $\mathcal{L}_N(u_m^*) = 0$.

Since we established that $u(0) = 1$ for all u in the model class, we know that for any u, $(u(0) - 1)^2$ is always 0. So, we have the optimization problem:

$$\min_{u \in \mathcal{NN}_n^{\text{ReLU}}(B)} \mathcal{L}_m(u) = \frac{1}{m} \sum_{i=1}^{m} \Big(u'(x_i) + \pi sin(\pi x_i) \Big)^2$$

Since all of the elements in the sum are nonnegative (as squares), we have:

$$\mathcal{L}_m(u) = 0 \iff u'(x_i) + \pi \sin(\pi x_i) = 0 \; \forall \; x_i$$

We know the structure of u in the model class, so we can compute u':

[2] Note this isn't in I_i to form a proper partition, hence the use of supremum

$$
\begin{aligned}
u'(x) &= \frac{d}{dx}\left[\sum_{i=1}^{n} c_i \phi(x-b_i)+1\right] \\
&= \sum_{i=1}^{n} c_i \frac{d}{dx}\left[\mathrm{ReLU}(x-b_i)\right]
\end{aligned}
$$

Now, $\mathrm{ReLU}(x-b_i) = \max\{x-b_i, 0\} = \begin{cases} x-b_i, x \geq b_i \\ 0, x < b_i \end{cases}$. We can then take the piecewise derivative as[3]:

$$
\begin{aligned}
\frac{d}{dx}\mathrm{ReLU}(x-b_i) &= \begin{cases} 1, x \geq b_i \\ 0, x < b_i \end{cases} \\
&= H(x-b_i) \;\; (H \text{ is Heaviside function})
\end{aligned}
$$

and so, $u'(x) = \sum_{i=1}^{n} c_i H(x-b_i)$. Now, we require:

$$
\sum_{i=1}^{n} c_i H(x_j - b_i) = \pi \sin(\pi x_j) \; \forall \; x_j
$$

First, let $n = m-1$, the number of collocation points. If we define $y_j = \pi \sin(\pi x_j)$ for simplicity, we know we can use Heaviside functions to interpolate data, by setting $c_i = (y_i - y_{i+1}), b_i = x_i$ such that:

$$
\begin{aligned}
u_m'(x) &= \sum_{i=1}^{m-1} c_i H(x-b_i) \\
&= \sum_{i=1}^{m-1} (y_i - y_{i+1}) H(x - x_i)
\end{aligned}
$$

Consider now:

$$
\begin{aligned}
u_m'(x_j) &= \sum_{i=1}^{m-1} (y_i - y_{i-1}) H(x_j - x_i) \\
&= \sum_{i \geq j}^{m-1} (y_i - y_{i+1}) \\
&= y_j - y_{j+1} + \sum_{i \geq j+1}^{m-1} (y_i - y_{i+1}) \\
&= y_j - y_m
\end{aligned}
$$

But, we can just extend the network one more sum in the term to include $c_m = y_m$. If we do this, we then have:

$$
\begin{aligned}
u_m'(x_j) &= \sum_{i=1}^{m-1} (y_i - y_{i-1}) H(x_j - x_i) \\
&= y_j - y_m + y_m \\
&= y_j
\end{aligned}
$$

Therefore, this network u_m parameterized in this way of width m interpolates all data points! Is this a member of the model class that's bounded?

We can check:

[3] Note that ReLU is not actually differentiable at $x = b_i$, but we define the derivative as 1 at $x = b_i$ to make a continuous sub-derivative.

$$
\begin{aligned}
|c_i| &= |\pi \sin(\pi x_i) - \pi \sin(\pi x_{i+1})| \\
&= \pi |\sin(\pi x_i) - \sin(\pi x_{i+1})| \\
&\leq \pi
\end{aligned}
$$

And since $b_i = x_i$, and we know the domain is $(0, 1)$, we have $b_i \in (0, 1)$ for any index. Therefore, we can bound the components above by $B = \pi$.

3.3 Convergence to True Solution as $m \to \infty$

Consider the loss function based on infinitely smooth sampling:

$$
\mathcal{L}_\infty(u_m^*) := \int_0^1 \left(\frac{d}{dx} u_m^*(x) + \pi \sin(\pi x) \right)^2 + (u_m^*(0) - 1)^2
$$

We can demonstrate $\mathcal{L}_\infty(u_m^*) \leq \left(\frac{\pi^2}{m+1}\right)^2$, which directly implies that $\lim_{m\to\infty} \mathcal{L}_\infty(u_m^*) = 0$.

We already know from 3.1 that $u_m^*(0) = 1$, so $(u_m^*(0) - 1)^2 = 0 \; \forall \; m$. Next, consider just the error between the interpolation network and the true value. Since we know the points x_i partition the domain $(0, 1)$, any x must be in some interval $[x_i, x_{i+1})$. However, we showed in 3.2 that $\frac{d}{dx} u_m^*(x)$ interpolates x_i exactly. So, let x_i be the closest collocation point included in the interval to x. Then:

$$
\begin{aligned}
\left| \frac{d}{dx} u_m^*(x) - \pi \sin(\pi x) \right| &\leq \left| \frac{d}{dx} u_m^*(x_i) - \pi \sin(\pi x) \right| \\
&= |\pi \sin(\pi x_i) - \pi \sin(\pi x)| \\
&= \pi \,|\sin(\pi x_i) - \sin(\pi x)| \\
&=\leq \pi(1)|\pi x_i - \pi x| \quad (sin \text{ is 1-Lipschitz}) \\
&= \pi^2 |x_i - x| \\
&\leq \pi^2 h \\
&= \frac{\pi^2}{m+1}
\end{aligned}
$$

Using this, we then have:

$$
\begin{aligned}
\mathcal{L}_\infty(u_m^*) &= \int_0^1 \left(\frac{d}{dx} u_m^*(x) + \pi \sin(\pi x) \right)^2 + (u_m^*(0) - 1)^2 \\
&\leq \int_0^1 \left(\frac{\pi^2}{m+1} \right)^2 dx + 0 \\
&= \left(\frac{\pi^2}{m+1} \right)^2
\end{aligned}
$$

Since this error is also always nonnegative, we have that $\lim_{m\to\infty} \mathcal{L}_\infty(u_m^*) = 0$. Since both functions are C^0, this also implies $\lim_{m\to} \frac{d}{dx} u_m^*(x) = -\pi \sin(\pi x)$

3.4 Convergence of Minimizers to True Solution

We can now show that the sequence $\{u_m^*\}_{m=1,\ldots}$ converges to the true solution $\cos(\pi x)$ in C_0.

We know immediately that these functions u_m^* are real valued, and defined on a compact interval $[0, 1]$ that's a subset of the real line[4]. Next, since we operate on the bounded model class $\mathcal{NN}_n^{\text{ReLU}}(B)$, we also have that these models are bounded operators. Since ReLU is 1-Lipschitz (ie $|\text{ReLU}(x) - \text{ReLU}(y)| \leq |x - y|$), we showed in 2.2 that this model class is equicontinuous such that for any model v, $|v(x) - v(y)| \leq (B^2 n)|x - y|$. Therefore, by the Arzela-Ascoli theorem, there is a subsequence $\{u_{m_k}^*\} \subseteq \{u_m^*\}$ that converges uniformly. Without proof, if we accept that this space of functions is a metric space, then sequential compactness is equivalent to compactness, and therefore $\{u_m^*\}$ converges uniformly.

[4] even if the collocation points are not on $x = 0, 1$

What is the limit of this sequence? Evaluating:

$$\begin{aligned}
\lim_{m\to\infty} u_m^*(x) &= \lim_{m\to\infty}\left[1+\int_0^x \frac{d}{ds}u_m^*(s)ds\right] \\
&= 1+\int_0^x \lim_{m\to\infty}\left[\frac{d}{dx}u_m^*(s)\right]ds \\
&= 1+\int_0^x \pi\sin(\pi s)ds \quad \text{(via 3.3)} \\
&= 1+\cos(\pi x)-\cos(\pi 0) \\
&= \cos(\pi x)
\end{aligned}$$

This is the known analytical solution to the original ODE.