1 Setting

We can mostly explain symmetry breaking and tiling, and so we finally want to explain localization. To do this, I focus on a single-neuron model with ReLU activation.

The task is to discriminate between two classes of inputs:

$$X_1 \sim p(\xi_1), \quad X_0 \sim p(\xi_0), \tag{1}$$

which are n-dimensional vectors. We only assume that p is *translation-invariant* (this is explained more below). Every input X_i has scalar label Y_i. The distribution p is paramaterized by $\xi > 0$, which defines the length-scale of correlations in the input. Specifically, we construct p so that

$$\mathrm{Cov}(X) = \Sigma(\xi), \quad \Sigma(\xi)_{ij} = \exp(-(i-j)^2/\xi^2). \tag{2}$$

We consider one neuron without bias and ReLU activation.

$$\hat{y}(x) = \mathrm{ReLU}(\langle w, x\rangle), \tag{3}$$

where w is our receptive field.

2 Dynamics

The dynamics of w is given by

$$\tau \frac{d}{dt} w = -\frac{\partial \mathcal{L}}{\partial w} = \frac{1}{2} \underbrace{\left[\frac{\partial}{\partial w} \mathbb{E}_{X|Y=Y_i}\left[\mathrm{ReLU}(\langle w, x\rangle)\right]\right]}_{\triangleq f(w)} - \frac{1}{2}(\Sigma_0 + \Sigma_1) w, \tag{4}$$

where i is the class with output label 1, i.e. $Y_i = 1$ and $Y_{1-i} = 0$. This elucidates some universal structure. We could also write f as

$$f(w) = \mathbb{E}_{X|Y=Y_i}\left[\mathbb{1}(\langle w, X\rangle \geq 0) X\right]. \tag{5}$$

This lets us establish some properties of f.

1. *It is invariant to scaling w*: $f(w) = f(\alpha w)$ for $\alpha > 0$.
2. *It is sign equivariant*: $f(-w) = -f(w)$.
3. *It is translation equivariant*[1]: $f(\mathcal{C}\, w) = \mathcal{C}\, f(w)$, where $\mathcal{C}$ is a circular shift.
4. *It can preserve symmetry in p*: If p is symmetric w.r.t. some invertible linear transformation A, i.e. $p_\xi(x) = p_\xi(Ax)$ for all x, then $f(Aw) = A^{-\top} f(w)$ [2].

(Note property 4 captures properties 2 and 3.) Thus, f only depends on the shape of w, not it's magnitude or position. So, it is precisely the object we need to understand.

The above equation also makes precise why a Gaussian approximation empirically always holds early in training. Early on, with w initialized as standard Gaussian, the second term dominates the dynamics. Note

[1] Here, "translation" refers to shifts of the *entries* of a vector. So, if $\mathcal{C}$ is a shift down by 1, $x_i = (\mathcal{C}\, x)_{i+1}$

[2] If A is orthogonal, then $f(Aw) = Af(w)$.

this term looks a lot like the update rule if we assume p is Gaussian. This term shrinks the receptive field, specifically shrinking lower-frequency oscillations faster than higher-frequency ones[3]. Once w becomes sufficiently small in norm so that the second term is on the order of $f(w)$, the first term is no longer negligible, and the dynamics are no longer Gaussian.

For certain distributions, we can evaluate $f(w)$ exactly, or at least find its form and interpret it. For Gaussian data,

$$f(w) = \frac{1}{\sqrt{2\pi}}(w^\top \Sigma_i w)^{-\frac{1}{2}}\Sigma_i w.$$

For elliptical data more generally,

$$f(w) = g(w^\top \Sigma_i w)\Sigma_i w,$$

where $g : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is some scalar function that scales like $\sim x^{-\frac{1}{2}}$ for inputs x. Given this form, we can write the update in Fourier space, given by $u = P^\top w$, where P is the (real) DFT matrix.

$$\tau \frac{d}{dt} u = g(u^\top \Lambda_i u)\Lambda_i u - \frac{1}{2}(\Lambda_0 + \Lambda_1)u, \tag{6}$$

where the Λ_i are diagonal matrices of eigenvalues of Σ_i. Because this is diagonal, we can compute steady states, which lets us show that the limiting solutions are a superposition of just one or two modes (i.e. u is sparse)[4]. This is insufficient for localization, which requires u to be sparse in the *spatial* domain, not the Fourier domain.

In general, it's hard to say much more about f. I don't know what it looks like for Alessandro's data model. **I want to empirically investigate how it behaves.** I think we can do this using Jax, but I'm not sure what the right questions to ask are or how to answer them.

Extra fact One additional thing we can say about f is that

$$\frac{\partial}{\partial w} f(w)_i \perp w \quad \forall i, \qquad \text{and} \qquad \frac{\partial}{\partial w_j} f(w) \perp w \quad \forall j.$$

This is either something important about interpreting f or entirely obvious. Not sure which, but this seems like a special consequence of using ReLU activation. To see why:

$$\begin{aligned}
\frac{\partial}{\partial w_j} f(w)_i &= \frac{\partial}{\partial w_j}\frac{\partial}{\partial w_i}\int_{\mathbb{R}^n} p_\xi(x)\,\mathrm{ReLU}(\langle w, x\rangle)dx \\
&= \frac{\partial}{\partial w_j}\int_{\mathbb{R}^n} p_\xi(x)\mathbb{1}(\langle w, x\rangle \geq 0)x_i dx \\
&= \int_{\mathbb{R}^n} p_\xi(x)\delta(\langle w, x\rangle)x_i x_j dx,
\end{aligned}$$

where δ is the Dirac delta function.

3 Signals on the hypercube

We will try to come up with some general sufficient conditions for localization. Let us introduce the data model of Alessandro as a starting point. There,

$$p_\xi(x) = \mathrm{Law}(X), \quad X_i = \frac{1}{\sqrt{\mathcal{Z}(g)}}\,\mathrm{erf}(gZ_i), \quad X \sim \mathcal{N}(0, \Sigma(\xi)), \tag{7}$$

[3] Does this make sense?

[4] I need to confirm this with simulations.

where $g > 0$ is our gain parameter and $\mathcal{Z}(g)$ is a normalization constant that ensures $\mathrm{Var}(X_i) = 1$ for all i. (Note that $\mathrm{Cov}(X) \neq \Sigma(\xi)$, but it's pretty close.) Importantly, as $g \to 0$, $X \overset{d}{\to} Z$, i.e. the data is approximately Gaussian. However, as $g \to \infty$, X becomes supported on the vertices of the hypercube $\{\pm 1\}^n$. After staring at equation (5) for a while, I think that this is the key to understanding localization. I'll explain this below, but first, some analytical examples.

Single bump If $w = e_i$, then

$$f(w) = \Sigma e_i. \tag{8}$$

Balanced bumps If $w = e_i + e_j$, then

$$f(w) = \Sigma(e_i + e_j). \tag{9}$$

Imbalanced bumps If $w = \alpha e_i + e_j$ for $\alpha > 1$, then

$$f(w) = \Sigma e_i. \tag{10}$$

Interesting!

If we flip the sign of the smaller bump so that If $w = \alpha e_i - e_j$ for $\alpha > 1$, then

$$f(w) = \Sigma e_i. \tag{11}$$

Cool!

Three bumps If $w = \alpha e_i + \beta e_j + e_k$ for $\alpha > \beta > 1$, then

$$f(w) \approx \Sigma e_i. \tag{12}$$

More generally? Assume $w_1 > 0$.

$$\mathbb{1}(\langle w, x \rangle \geq 0) = \mathbb{1}(x_1 \geq -\sum_{i=2}^{n} (\tfrac{w_i}{w_1}) x_i).$$

If $\sum_{i=2}^{n} |\frac{w_i}{w_1}| < 1$, this is equivalent to $\mathbb{1}(x_1 \geq 0) = \mathbb{1}(x_1 \geq 1)$. This is a pretty strong condition on w that is not usually true. However, it gives us a starting point for how we might be able to generally cut away a lot of the complexity of f when the data is supported on the vertices of the hypercube.

Let's consider some separation point k, (recall we assume $|w_1| > \ldots > |w_n|$, but this is just to make the sums easier to write—we just need to partition the entries of w into two sets).

$$\mathbb{1}(\langle w, x \rangle \geq 0) = \mathbb{1}(\sum_{i=1}^{k} w_i x_i \geq -\sum_{i=k+1}^{n} w_i x_i).$$

What is the smallest positive value the LHS produces? Define

$$\delta \triangleq \min_{x \in \{\pm 1\}^k} \left| \sum_{i=1}^{k} w_i x_i \right|. \tag{13}$$

Then, if $\left|\sum_{i=k+1}^{n} w_i x_i\right| < \delta$ for all x, which in this case is equivalent to $\sum_{i=k+1}^{n} |w_i| < \delta$, we have

$$\mathbb{1}(\langle w, x\rangle \geq 0) = \mathbb{1}(\sum_{i=1}^{k} w_i x_i \geq 0).$$

We want $\mathcal{X}$ s.t. we can make k small in the following inequality.

$$\min_{x\in\mathcal{X}} \left|\sum_{i=1}^{k} w_i x_i\right| > \max_{x\in\mathcal{X}} \left|\sum_{i=k+1}^{n} w_i x_i\right|$$

Again, we're making *universal statements* about f without assuming the underlying probability distribution (other than that it's supported on the hypercube). In practice, we can do better than having x in equation (13) range across the entire k-dimensional hypercube, *considering instead some subset of it that occurs with high probability under p.* This would allow us to cut away even more of the complexity of f. Obviously, we'd want to consider the smallest k such that the condition above holds.

If k is sufficiently small, then we can cut away a lot of the complexity of f. More specifically, if w_i is sufficiently small, the indicator function treats it as if it were zero. However, we need to understand how the remaining terms, which cannot be treated like zero, affect the indicator function.

SDP

Equation (13) is related to the integer optimization problem

$$\delta_{\text{INT}} \triangleq \max_{x\in\{\pm 1\}^k} \sum_{i,j=1}^{k} A_{ij} x_i x_j,$$

where $A = -ww^\top$. This is because $\sum_{i,j=1}^{k} A_{ij} x_i x_j = (x^\top A x) = -(\langle w, x\rangle)^2$. So,

$$\delta_{\text{INT}} = \max_{x\in\{\pm 1\}^k} -(\langle w, x\rangle)^2 = -\min_{x\in\{\pm 1\}^k} (\langle w, x\rangle)^2 = -\sqrt{\delta}.$$

It would be cool to bound δ from below in terms of w. Grothendieck's inequality might let us do this, albeit with a rather loose bound.

3.1 Simulations

In the few examples above, we've been able to show analytically that, because X has support on the hypercube, f extracts the maximum value of w. This makes a lot of sense! But we'd like to show this in a more general setting.

Something like X_j is approximately independent of $\mathbb{1}(\langle w, X\rangle \geq 0)$ when j does not correspond to the maximum absolute entry in w. Otherwise, $X_j \approx \text{sgn}(w_j)$.

Let's consider the set of x

$$\begin{aligned}
\mathbb{E}_X[\mathbb{1}(\langle w, x\rangle \geq 0)X] &= \mathbb{E}_X\left[\sum_{x'\in\Theta} \mathbb{1}(X = x')X\right] && (14)\\
&= \mathbb{E}_X\left[\sum_{x'\in\Theta}\left[\prod_{i=1}^{n} \mathbb{1}(X_i = x_i')\right]X\right] && (15)\\
&= \mathbb{E}_X\left[\sum_{x'\in\Theta}\left[\prod_{i=1}^{n} \frac{\operatorname{sgn}(x_i')}{2}(X_i + x_i')\right]X\right] && (16)\\
&= \sum_{x'\in\Theta}\mathbb{E}_X\left[\left(\prod_{i=1}^{n} \frac{\operatorname{sgn}(x_i')}{2}(X_i + x_i')\right)X\right]. && (17)
\end{aligned}$$

Entrywise,

$$\sum_{x'\in\Theta}\mathbb{E}_X\left[\left(\prod_{i=1}^{n} \frac{\operatorname{sgn}(x_i')}{2}(X_i + x_i')\right)X_j\right] = \sum_{x'\in\Theta}\mathbb{E}_X\left[\left(\prod_{i=1}^{n} \frac{\operatorname{sgn}(x_i')}{2}(X_i + x_i')X_j\right)\right]$$

First, I want to understand how we could rigorously apply this intuition to simplify f. Let's start by defining the event

$$\Theta = \left\{x = (x_j)_{j\in[n]} \in \{\pm 1\}^n \mid \langle w, x\rangle \geq 0\right\}. \quad (18)$$

Then,

$$\mathbb{1}(\langle w, x\rangle \geq 0) = \sum_{x'\in\Theta}\mathbb{1}(x = x') = \sum_{x'\in\Theta}\prod_{j\in[n]}\mathbb{1}(x_j = x_j') \quad (19)$$

So,

$$\mathbb{E}_{X|Y=1}\left[\mathbb{1}(\langle w, X\rangle \geq 0)X\right] = \sum_{j\in[n]}\mathbb{E}_{X|Y=1}\left[\underbrace{\left(\sum_{x'\in\Theta}\mathbb{1}(X_j = x_j')\right)}_{\triangleq g_j(X)}X\right]. \quad (20)$$

With this perspective, we want to show

1. $g_j(x) \approx \mathbb{1}(x_j = \operatorname{sign}(w_j))$ when w_j is sufficiently large, and
2. $g_j(x) \approx 1$ when w_j is not sufficiently large.

We also want to understand what it means for w_j to be sufficiently large. Hopefully, we can show there is a pretty clear divide between the two cases, and that nothing falls in between. **Now, how do I do this?**

We'll start with the case where $w_j > 0$ for the sufficiently large case.

Let's consider the case where only one w_j is sufficiently large. WLOG, let's say this happens for $j = 1$. As we make it larger, how do the $g_j(x)$ change?

Note that

$$\begin{aligned}
g_j(x) &= \sum_{x'\in\Theta}\mathbb{1}(x_j = x_j') && (21)\\
&= |\{x' \in \Theta \mid x_j' = 1\}|\mathbb{1}(x_j = 1) + |\{x' \in \Theta \mid x_j' = -1\}|\mathbb{1}(x_j = -1) && (22)
\end{aligned}$$