Supplemental material

A. Reduced Noise through Decomposition - Experiment

A.1. Experiment

For this experiment, consider the problem of training a predictor, which given a "positive media reference" $\mathbf{x}$ to a certain stock option, will distribute our assets between the $k = 500$ stocks in the S&P500 index in some manner. One can, again, come up with two rather different strategies for solving the problem.

- An end-to-end approach: train a deep network $N_{\mathbf{w}}$ that given $\mathbf{x}$ outputs a distribution over the k stocks. The objective for training is maximizing the gain obtained by allocating our money according to this distribution.

- A decomposition approach: train a deep network $N_{\mathbf{w}}$ that given $\mathbf{x}$ outputs a single stock, $y \in [k]$, whose future gains are the most positively correlated to $\mathbf{x}$. Of course, we may need to gather extra labeling for training $N_{\mathbf{w}}$ based on this criterion.

We make the (non-realistic) assumption that every instance of media reference is strongly and positively correlated to a single stock $y \in [k]$, and it has no correlation with future performance of other stocks. This obviously makes our problem rather toyish; the stock exchange and media worlds have highly complicated correlations. However, it indeed arises from, and is motivated by, practical problems.

To examine the problem in a simple and theoretically clean manner, we design a synthetic experiment defined by the following optimization problem: Let $X \times Z \subset \mathbb{R}^d \times \{\pm 1\}^k$ be the sample space, and let $y : X \to [k]$ be some labelling function. We would like to learn a mapping $N_{\mathbf{w}} : X \to S^{k-1}$, with the objective being:

$$\min_{\mathbf{w}} L(\mathbf{w}) := \mathop{\mathbb{E}}_{\mathbf{x},\mathbf{z}\sim X\times Z} \left[-\mathbf{z}^\top N_{\mathbf{w}}(\mathbf{x})\right].$$

To connect this to our story, $N_{\mathbf{w}}(\mathbf{x})$ is our asset distribution, $\mathbf{z}$ indicates the future performance of the stocks, and thus, we are seeking minimization of our expected future negative gains, or in other words, maximization of expected profit. We further assume that given $\mathbf{x}$, the coordinate $\mathbf{z}_{y(\mathbf{x})}$ equals 1, and the rest of the coordinates are sampled i.i.d from the uniform distribution over $\{\pm 1\}$.

Whereas in Section 3.1's experiment, the difference between the end-to-end and decomposition approaches could be summarized by a different loss function choice, in this experiment, the difference boils down to the different gradient estimators we would use, where we are again taking as a given fact that exact gradient computations are expensive for large-scale problems, implying the method of choice to be SGD. For the purpose of the experimental discussion, let us write the two estimators explicitly as two unconnected update rules. We will later analyze their (equal) expectation.

For an end-to-end approach, we sample a pair $(\mathbf{x}, \mathbf{z})$, and use $\nabla_{\mathbf{w}}(-\mathbf{z}^\top N_{\mathbf{w}}(\mathbf{x}))$ as a gradient estimate. It is clear that this is an unbiased estimator of the gradient.

For a decomposition approach, we sample a pair $(\mathbf{x}, \mathbf{z})$, *completely ignore* $\mathbf{z}$, and instead, pay the extra costs and gather the required labelling to get $y(\mathbf{x})$. We will then use $\nabla_{\mathbf{w}}(-e_{y(\mathbf{x})}^\top N_{\mathbf{w}}(\mathbf{x}))$ as a gradient estimate. It will be shown later that this too is an unbiased estimator of the gradient.

Figure 7 clearly shows that optimizing using the end-to-end estimator is inferior to working with the decomposition one, in terms of training time and final accuracy, to the extent that for large k, the end-to-end estimator cannot close the gap in performance in reasonable time.

A.2. Analysis

We examine the experiment from a SNR perspective. First, let us show that indeed, both estimators are unbiased estimators of the true gradient. As stated above, it is clear, by definition of L, that the end-to-end estimator is an unbiased estimator

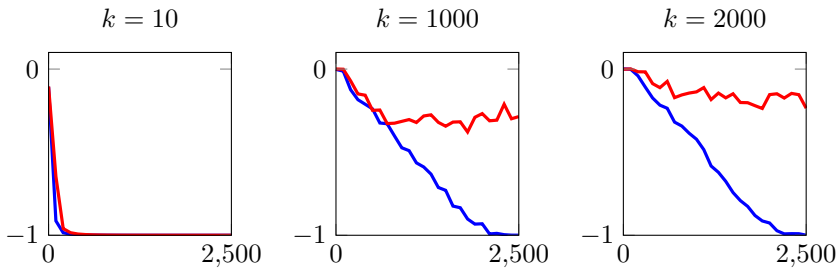

Figure 7. Decomposition vs. end-to-end Experiment: Loss as a function of the number of training iterations, for input dimension $d = 1000$ and for various k values. The red and blue curves correspond to the losses of the end-to-end and decomposition estimators, respectively.

of $\nabla_{\mathbf{w}} L(\mathbf{w})$. To observe this is also the case for the decomposition estimator, we write:

$$\begin{aligned}
&\nabla_{\mathbf{w}} L(\mathbf{w}) = \nabla_{\mathbf{w}} \mathop{\mathbb{E}}_{\mathbf{x},\mathbf{z}}[-\mathbf{z}^\top N_{\mathbf{w}}(\mathbf{x})] \\
&= \mathop{\mathbb{E}}_{\mathbf{x}}[\mathop{\mathbb{E}}_{\mathbf{z}|\mathbf{x}}[\nabla_{\mathbf{w}}(-\mathbf{z}^\top N_{\mathbf{w}}(\mathbf{x}))]] \\
&\overset{(1)}{=} \mathop{\mathbb{E}}_{\mathbf{x}}[\mathop{\mathbb{E}}_{\mathbf{z}|\mathbf{x}}[-\mathbf{z}^\top \nabla_{\mathbf{w}}(N_{\mathbf{w}}(\mathbf{x}))]] \overset{(2)}{=} \mathop{\mathbb{E}}_{\mathbf{x}}[-e_{y(\mathbf{x})}^\top \nabla_{\mathbf{w}}(N_{\mathbf{w}}(\mathbf{x}))]
\end{aligned}$$

where (1) follows from the chain rule, and (2) from the assumption on the distribution of $\mathbf{z}$ given $\mathbf{x}$. It is now easy to see that the decomposition estimator is indeed a (different) unbiased estimator of the gradient, hence the "signal" is the same.

Intuition says that when a choice between two unbiased estimators is presented, we should choose the one with the lower variance. In our context, (Ghadimi & Lan, 2013) showed that when running SGD (even on non-convex objectives), arriving at a point where $\|\nabla_{\mathbf{w}} L(\mathbf{w})\|^2 \leq \epsilon$ requires order of $\bar{\nu}^2/\epsilon^2$ iterations, where

$$\bar{\nu}^2 = \max_t \mathop{\mathbb{E}}_{\mathbf{x},q} \|\nabla_{\mathbf{w}}^t(\mathbf{x}, q)\|^2 - \|\nabla_{\mathbf{w}} L(\mathbf{w}^{(t)})\|^2,$$

$\mathbf{w}^t$ is the weight vector at time t, q is sampled along with $\mathbf{x}$ (where it can be replaced by $\mathbf{z}$ or $y(\mathbf{x})$, in our experiment), and $\nabla_{\mathbf{w}}^t$ is the unbiased estimator for the gradient. This serves as a motivation for analyzing the problem through this lens.

Motivated by (Ghadimi & Lan, 2013)'s result, and by our results regarding Section 3.1, we examine the quantity $\mathbb{E}_{\mathbf{x},q} \|\nabla_{\mathbf{w}}^t(\mathbf{x}, q)\|^2$, or "noise", explicitly. For the end-to-end estimator, this quantity equals

$$\mathop{\mathbb{E}}_{\mathbf{x},\mathbf{z}} \| - \mathbf{z}^\top \nabla_{\mathbf{w}} N_{\mathbf{w}}(\mathbf{x})\|^2 = \mathop{\mathbb{E}}_{\mathbf{x},\mathbf{z}} \| - \sum_{i=1}^k \mathbf{z}_i \nabla_{\mathbf{w}} N_{\mathbf{w}}(\mathbf{x})_i\|^2$$

Denoting by $G_i := \nabla_{\mathbf{w}} N_{\mathbf{w}}(\mathbf{x})_i$, we get:

$$= \mathop{\mathbb{E}}_{\mathbf{x}} \mathop{\mathbb{E}}_{\mathbf{z}|\mathbf{x}} \| - \sum_{i=1}^k \mathbf{z}_i G_i\|^2 = \mathop{\mathbb{E}}_{\mathbf{x}} \sum_{i=1}^k \|G_i\|^2 \tag{3}$$

where the last equality follows from expanding the squared sum, and taking expectation over $\mathbf{z}$, while noting that mixed terms cancel out (from independence of $\mathbf{z}$'s coordinates), and that $\mathbf{z}_i^2 = 1$ for all i.

As for the decomposition estimator, it is easy to see that

$$\mathop{\mathbb{E}}_{\mathbf{x}} \| - e_{y(\mathbf{x})}^\top \nabla_{\mathbf{w}} N_{\mathbf{w}}(\mathbf{x})\|^2 = \mathop{\mathbb{E}}_{\mathbf{x}} \|G_{y(\mathbf{x})}\|^2. \tag{4}$$

Observe that in 3 we are summing up, per $\mathbf{x}$, k summands, compared to the single element in 4. When randomly initializing a network it is likely that the values of $\|G_i\|^2$ are similar, hence we obtain that at the beginning of training, the variance of the end-to-end estimator is roughly k times larger than that of the decomposition estimator.

B. Proofs

B.1. Proof of Theorem 1

Proof Given two square-integrable functions f, g on an Euclidean space $\mathbb{R}^n$, let $\langle f, g\rangle_{L_2} = \mathbb{E}_{\mathbf{x}}[f(\mathbf{x})g(\mathbf{x})]$ and $\|f\|_{L_2} = \sqrt{\mathbb{E}_{\mathbf{x}}[f^2(\mathbf{x})]}$ denote inner product and norm in the L_2 space of square-integrable functions (with respect to the relevant distribution). Also, define the vector-valued function

$$\mathbf{g}(\mathbf{x}) = \frac{\partial}{\partial \mathbf{w}} p_{\mathbf{w}}(\mathbf{x}),$$

and let $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_n(\mathbf{x}))$ for real-valued functions $g_1, \ldots, g_n$. Finally, let $\mathbb{E}_h$ denote an expectation with respect to h chosen uniformly at random from $\mathcal{H}$. Let $|\mathcal{H}| = d$.

We begin by proving the result for the squared loss. To prove the bound, it is enough to show that $\mathbb{E}_h \|\nabla F_h(\mathbf{w}) - \mathbf{a}\|^2 \leq \frac{G^2}{|\mathcal{H}|}$ for any vector $\mathbf{a}$ independent of h. In particular, let us choose $\mathbf{a} = \mathbb{E}_{\mathbf{x}}\left[p_{\mathbf{w}}(\mathbf{x})\mathbf{g}(\mathbf{x})\right]$. We thus bound the following:

$$\begin{aligned}
\mathop{\mathbb{E}}_h \|\nabla F_h(\mathbf{w}) - \mathop{\mathbb{E}}_{\mathbf{x}}\left[p_{\mathbf{w}}(\mathbf{x})\mathbf{g}(\mathbf{x})\right]\|^2 &= \mathop{\mathbb{E}}_h \| \mathop{\mathbb{E}}_{\mathbf{x}}\left[\left(p_{\mathbf{w}}(\mathbf{x}) - h(\mathbf{x})\right)\mathbf{g}(\mathbf{x})\right] - \mathop{\mathbb{E}}_{\mathbf{x}}\left[p_{\mathbf{w}}(\mathbf{x})\mathbf{g}(\mathbf{x})\right]\|^2 \\
&= \mathop{\mathbb{E}}_h \| \mathop{\mathbb{E}}_{\mathbf{x}}\left[h(\mathbf{x})\mathbf{g}(\mathbf{x})\right]\|^2 \;=\; \mathop{\mathbb{E}}_h \sum_{j=1}^{n} \left(\mathop{\mathbb{E}}_{\mathbf{x}}\left[h(\mathbf{x})g_j(\mathbf{x})\right]\right)^2 \\
&= \mathop{\mathbb{E}}_h \sum_{j=1}^{n} \langle h, g_j\rangle_{L_2}^2 \;=\; \sum_{j=1}^{n}\left(\frac{1}{|\mathcal{H}|}\sum_{i=1}^{d}\langle h_i, g_j\rangle_{L_2}^2\right) \\
&\overset{(*)}{\leq} \sum_{j=1}^{n}\left(\frac{1}{|\mathcal{H}|}\|g_j\|_{L_2}^2\right) \;=\; \frac{1}{|\mathcal{H}|}\sum_{j=1}^{n}\mathop{\mathbb{E}}_{\mathbf{x}}[g_j^2(\mathbf{x})] \\
&= \frac{1}{|\mathcal{H}|}\mathop{\mathbb{E}}_{\mathbf{x}}\left[\|\mathbf{g}(\mathbf{x})\|^2\right] \;\leq\; \frac{G(\mathbf{w})^2}{|\mathcal{H}|},
\end{aligned}$$

where $(*)$ follows from the functions in $\mathcal{H}$ being mutually orthogonal, and satisfying $\|h\|_{L_2} \leq 1$ for all $h \in \mathcal{H}$.

To handle a classification loss, note that by its definition and the fact that $h(\mathbf{x}) \in \{-1, +1\}$,

$$\begin{aligned}
\nabla F_h(\mathbf{w}) &= \mathop{\mathbb{E}}_{\mathbf{x}}\left[r'(h(\mathbf{x})p_{\mathbf{w}}(\mathbf{x}))\cdot\frac{\partial}{\partial\mathbf{w}}p_{\mathbf{w}}(\mathbf{x})\right] \\
&= \mathop{\mathbb{E}}_{\mathbf{x}}\left[\left(\frac{r'(p_{\mathbf{w}}(\mathbf{x})) + r'(-p_{\mathbf{w}}(\mathbf{x}))}{2} + h(\mathbf{x})\cdot\frac{r'(p_{\mathbf{w}}(\mathbf{x})) - r'(-p_{\mathbf{w}}(\mathbf{x}))}{2}\right)\cdot\frac{\partial}{\partial\mathbf{w}}p_{\mathbf{w}}(\mathbf{x})\right] \\
&= \mathop{\mathbb{E}}_{\mathbf{x}}\left[\frac{r'(p_{\mathbf{w}}(\mathbf{x})) + r'(-p_{\mathbf{w}}(\mathbf{x}))}{2}\cdot\frac{\partial}{\partial\mathbf{w}}p_{\mathbf{w}}(\mathbf{x})\right] + \mathop{\mathbb{E}}_{\mathbf{x}}\left[h(\mathbf{x})\cdot\left(\frac{r'(p_{\mathbf{w}}(\mathbf{x})) - r'(-p_{\mathbf{w}}(\mathbf{x}))}{2}\right)\cdot\frac{\partial}{\partial\mathbf{w}}p_{\mathbf{w}}(\mathbf{x})\right].
\end{aligned}$$

Letting $\mathbf{g}(\mathbf{x}) = \left(\frac{r'(p_{\mathbf{w}}(\mathbf{x})) - r'(-p_{\mathbf{w}}(\mathbf{x}))}{2}\right)\cdot\frac{\partial}{\partial\mathbf{w}}p_{\mathbf{w}}(\mathbf{x})$ (which satisfies $\mathbb{E}_{\mathbf{x}}[\|\mathbf{g}(\mathbf{x})\|^2] \leq G^2$ since r is 1-Lipschitz) and $\mathbf{a} = \mathbb{E}_{\mathbf{x}}\left[\frac{r'(p_{\mathbf{w}}(\mathbf{x})) + r'(-p_{\mathbf{w}}(\mathbf{x}))}{2}\cdot\frac{\partial}{\partial\mathbf{w}}p_{\mathbf{w}}(\mathbf{x})\right]$ (which does not depend on h), we get that

$$\mathop{\mathbb{E}}_h \|\nabla F_h(\mathbf{w}) - \mathbf{a}\|^2 = \mathop{\mathbb{E}}_h \|\mathop{\mathbb{E}}_{\mathbf{x}}[h(\mathbf{x})\mathbf{g}(\mathbf{x})]\|^2.$$

Proceeding now exactly in the same manner as the squared loss case, the result follows. ■

B.2. Proof of Theorem 3

Proof We first state and prove two auxiliary lemmas.

Lemma 1 *Let $h_1, \ldots, h_n$ be real-valued functions on some Euclidean space, which belong to some weighted L_2 space. Suppose that $\|h_i\|_{L_2} = 1$ and $\max_{i\neq j}|\langle h_i, h_j\rangle_{L_2}| \leq c$. Then for any function g on the same domain,*

$$\frac{1}{n}\sum_{i=1}^{n}\langle h_i, g\rangle_{L_2}^2 \;\leq\; \|g\|_{L_2}^2\left(\frac{1}{n} + c\right).$$

Proof For simplicity, suppose first that the functions are defined over some finite domain equipped with a uniform distribution, so that $h_1, \ldots, h_n$ and g can be thought of as finite-dimensional vectors, and the L_2 inner product and norm reduce to the standard inner product and norm in Euclidean space. Let $H = (h_1, \ldots, h_n)$ denote the matrix whose i-th column is h_i. Then

$$\sum_{i=1}^{n} \langle h_i, g \rangle^2 = g^\top \left(\sum_{i=1}^{n} h_i h_i^\top \right) g \;=\; g^\top H H^\top g \leq \|g\|^2 \|H H^\top\| \;=\; \|g\|^2 \|H^\top H\|,$$

where $\|\cdot\|$ for a matrix denotes the spectral norm. Since $H^\top H$ is simply the $n \times n$ matrix with entry $\langle h_i, h_j \rangle$ in location i, j, we can write it as $I + M$, where I is the $n \times n$ identity matrix, and M is a matrix with 0 along the main diagonal, and entries of absolute value at most c otherwise. Therefore, letting $\|\cdot\|_F$ denote the Frobenius norm, we have that the above is at most

$$\|g\|^2 \left(\|I\| + \|M\| \right) \;\leq\; \|g\|^2 \left(1 + \|M\|_F \right) \;=\; \|g\|^2 \left(1 + cn \right),$$

from which the result follows.

Finally, it is easily verified that the same proof holds even when $h_1, \ldots, h_n, g$ are functions over some Euclidean space, belonging to some weighted L_2 space. In that case, H is a bounded linear operator, and it holds that $\|H^* H\| = \|H\|^2 = \|H^*\|^2 = \|H H^*\|$ where H^* is the Hermitian adjoint of H and the norm is the operator norm. The rest of the proof is essentially identical. ■

Lemma 2 *If $\mathbf{w}, \mathbf{v}$ are two unit vectors in $\mathbb{R}^d$, and $\mathbf{x}$ is a standard Gaussian random vector, then*

$$\left| \mathop{\mathbb{E}}_{\mathbf{x}} \left[\mathit{sign}(\mathbf{w}^\top \mathbf{x}) \mathit{sign}(\mathbf{v}^\top \mathbf{x}) \right] \right| \leq |\langle \mathbf{w}, \mathbf{v} \rangle|$$

Proof Note that $\mathbf{w}^\top \mathbf{x}, \mathbf{v}^\top \mathbf{x}$ are jointly zero-mean Gaussian, each with variance 1 and with covariance $\mathbb{E}[\mathbf{w}^\top \mathbf{x} \mathbf{x}^\top \mathbf{v}] = \mathbf{w}^\top \mathbf{v}$. Therefore,

$$\begin{aligned}
\mathop{\mathbb{E}}_{\mathbf{x}} \left[\mathrm{sign}(\mathbf{w}^\top \mathbf{x}) \mathrm{sign}(\mathbf{v}^\top \mathbf{x}) \right] &= \Pr(\mathbf{w}^\top \mathbf{x} \geq 0, \mathbf{v}^\top \mathbf{x} \geq 0) + \Pr(\mathbf{w}^\top \mathbf{x} \leq 0, \mathbf{v}^\top \mathbf{x} \leq 0) \\
&\quad - \Pr(\mathbf{w}^\top \mathbf{x} \geq 0, \mathbf{v}^\top \mathbf{x} \leq 0) - \Pr(\mathbf{w}^\top \mathbf{x} \leq 0, \mathbf{v}^\top \mathbf{x} \geq 0) \\
&= 2 \Pr(\mathbf{w}^\top \mathbf{x} \geq 0, \mathbf{v}^\top \mathbf{x} \geq 0) - 2 \Pr(\mathbf{w}^\top \mathbf{x} \geq 0, \mathbf{v}^\top \mathbf{x} \leq 0),
\end{aligned}$$

which by a standard fact on the quadrant probability of bivariate normal distributions, equals

$$\begin{aligned}
&2 \left(\frac{1}{4} + \frac{\sin^{-1}(\mathbf{w}^\top \mathbf{v})}{2\pi} \right) - 2 \left(\frac{\cos^{-1}(\mathbf{w}^\top \mathbf{v})}{2\pi} \right) = \frac{1}{2} + \frac{1}{\pi} \left(\sin^{-1}(\mathbf{w}^\top \mathbf{v}) - \cos^{-1}(\mathbf{w}^\top \mathbf{v}) \right) \\
&= \frac{1}{2} + \frac{1}{\pi} \left(2 \sin^{-1}(\mathbf{w}^\top \mathbf{v}) - \frac{\pi}{2} \right) \;=\; \frac{2 \sin^{-1}(\mathbf{w}^\top \mathbf{v})}{\pi} \,.
\end{aligned}$$

The absolute value of the above can be easily verified to be upper bounded by $|\mathbf{w}^\top \mathbf{v}|$, from which the result follows. ■

With these lemmas at hand, we turn to prove our theorem. By a standard measure concentration argument, we can find d^k unit vectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{d^k}$ in $\mathbb{R}^d$ such that their inner product is at most $O(\sqrt{k \log(d)/d})$ (where the $O(\cdot)$ notation is w.r.t. d). This induces d^k functions $h_{\mathbf{u}_1}, h_{\mathbf{u}_2}, \ldots, h_{\mathbf{u}_{d^k}}$ where $h_{\mathbf{u}}(\mathbf{x}_1, \ldots, \mathbf{x}_k) = \prod_{l=1}^{k} \mathrm{sign}(\mathbf{u}^\top \mathbf{x}_l)$. Their L_2 norm (w.r.t. the distribution over $\mathbf{x}_1^k = (\mathbf{x}_1, \ldots, \mathbf{x}_k)$) is 1, as they take values in $\{-1, +1\}$. Moreover, since $\mathbf{x}_1, \ldots, \mathbf{x}_k$ are i.i.d.

standard Gaussian, we have by Lemma 2 that for any $i \neq j$,

$$\begin{aligned}\langle h_{\mathbf{u}_i}, h_{\mathbf{u}_j}\rangle_{L_2} &= \left|\mathbb{E}\left[\prod_{l=1}^{k}\text{sign}(\mathbf{u}_i^\top \mathbf{x}_l)\prod_{l=1}^{k}\text{sign}(\mathbf{u}_j^\top \mathbf{x}_l)\right]\right| \\ &= \left|\prod_{l=1}^{k}\mathbb{E}\left[\text{sign}(\mathbf{u}_i^\top \mathbf{x}_l)\text{sign}(\mathbf{u}_j^\top \mathbf{x}_l)\right]\right| \\ &= \left|\mathbb{E}\left[\text{sign}(\mathbf{u}_i^\top \mathbf{x}_l)\text{sign}(\mathbf{u}_j^\top \mathbf{x}_l)\right]\right|^k \\ &\leq |\mathbf{u}_i^\top \mathbf{u}_j|^k \;\leq\; O\left(\sqrt{\frac{k\log(d)}{d}}\right)^k.\end{aligned}$$

Using this and Lemma 1, we have that for any function g,

$$\frac{1}{d^k}\sum_{i=1}^{d^k}\langle h_{\mathbf{u}_i}, g\rangle_{L_2}^2 \;\leq\; \|g\|_{L_2}^2\cdot\left(\frac{1}{d^k}+O\left(\sqrt{\frac{k\log(d)}{d}}\right)^k\right) \;\leq\; \|g\|_{L_2}^2\cdot O\left(\sqrt{\frac{k\log(d)}{d}}\right)^k.$$

Moreover, since this bound is derived based only on an inner product condition between $\mathbf{u}_1,\ldots,\mathbf{u}_{d^k}$, the same result would hold for $U\mathbf{u}_1,\ldots,U\mathbf{u}_{d^k}$ where U is an arbitrary orthogonal matrix, and in particular if we pick it uniformly at random:

$$\mathop{\mathbb{E}}_{U}\left[\frac{1}{d^k}\sum_{i=1}^{d^k}\langle h_{U\mathbf{u}_i}, g\rangle_{L_2}^2\right] \;\leq\; \|g\|_{L_2}^2\cdot\left(\frac{1}{d^k}+O\left(\sqrt{\frac{k\log(d)}{d}}\right)\right).$$

Now, note that for any fixed i, $U\mathbf{u}_i$ is uniformly distributed on the unit sphere, so the left hand side simply equals $\mathbb{E}_{\mathbf{u}}\left[\langle h_{\mathbf{u}}, g\rangle_{L_2}^2\right]$, and we get

$$\mathop{\mathbb{E}}_{\mathbf{u}}\left[\langle h_{\mathbf{u}}, g\rangle_{L_2}^2\right] \;\leq\; \|g\|^2\cdot O\left(\sqrt{\frac{k\log(d)}{d}}\right)^k. \tag{5}$$

With this key inequality at hand, the proof is now very similar to the one of Theorem 1. Given the predictor $p_{\mathbf{w}}(\mathbf{x}_1^k)$, where $\mathbf{w}\in\mathbb{R}^n$, define the vector-valued function $\mathbf{g}(\mathbf{x}_1^k)=\frac{\partial}{\partial \mathbf{w}}p_{\mathbf{w}}(\mathbf{x}_1^k)$, and let $\mathbf{g}(\mathbf{x}_1^k)=(g_1(\mathbf{x}_1^k),g_2(\mathbf{x}_1^k),\ldots,g_n(\mathbf{x}_1^k))$ for real-valued functions $g_1,\ldots,g_n$. To prove the bound, it is enough to upper bound $\mathbb{E}_{\mathbf{u}}\|\nabla F_{\mathbf{u}}(\mathbf{w})-\mathbf{a}\|^2$ for any vector $\mathbf{a}$ independent of $\mathbf{u}$. In particular, let us choose $\mathbf{a}=\mathbb{E}_{\mathbf{x}_1^k}\left[p_{\mathbf{w}}(\mathbf{x}_1^k)\mathbf{g}(\mathbf{x}_1^k)\right]$. We thus bound the following:

$$\begin{aligned}\mathop{\mathbb{E}}_{\mathbf{u}}\|\nabla F_{\mathbf{u}}(\mathbf{w})-\mathop{\mathbb{E}}_{\mathbf{x}_1^k}\left[p_{\mathbf{w}}(\mathbf{x}_1^k)\mathbf{g}(\mathbf{x}_1^k)\right]\|^2 &= \mathop{\mathbb{E}}_{\mathbf{u}}\|\mathop{\mathbb{E}}_{\mathbf{x}_1^k}\left[\left(p_{\mathbf{w}}(\mathbf{x}_1^k)-h_{\mathbf{u}}(\mathbf{x}_1^k)\right)\mathbf{g}(\mathbf{x}_1^k)\right]-\mathop{\mathbb{E}}_{\mathbf{x}_1^k}\left[p_{\mathbf{w}}(\mathbf{x}_1^k)\mathbf{g}(\mathbf{x}_1^k)\right]\|^2 \\ &= \mathop{\mathbb{E}}_{\mathbf{u}}\|\mathop{\mathbb{E}}_{\mathbf{x}_1^k}\left[h_{\mathbf{u}}(\mathbf{x}_1^k)\mathbf{g}(\mathbf{x}_1^k)\right]\|^2 \;=\; \mathop{\mathbb{E}}_{\mathbf{u}}\sum_{j=1}^{n}\left(\mathop{\mathbb{E}}_{\mathbf{x}_1^k}\left[h_{\mathbf{u}}(\mathbf{x}_1^k)g_j(\mathbf{x}_1^k)\right]\right)^2 \\ &= \mathop{\mathbb{E}}_{\mathbf{u}}\sum_{j=1}^{n}\langle h_{\mathbf{u}}, g_j\rangle_{L_2}^2 \;=\; \sum_{j=1}^{n}\mathop{\mathbb{E}}_{\mathbf{u}}\langle h_{\mathbf{u}}, g_j\rangle_{L_2}^2 \\ &\overset{(*)}{\leq} \sum_{j=1}^{n}\|g_j\|^2\cdot O\left(\sqrt{\frac{k\log(d)}{d}}\right)^k \;=\; \sum_{j=1}^{n}\mathop{\mathbb{E}}_{\mathbf{x}_1^k}[g_j^2(\mathbf{x}_1^k)]\cdot O\left(\sqrt{\frac{k\log(d)}{d}}\right)^k \\ &= \mathop{\mathbb{E}}_{\mathbf{x}_1^k}\|\mathbf{g}(\mathbf{x}_1^k)\|^2\cdot O\left(\sqrt{\frac{k\log(d)}{d}}\right)^k \;\leq\; G(\mathbf{w})^2\cdot O\left(\sqrt{\frac{k\log(d)}{d}}\right)^k,\end{aligned}$$

where $(*)$ follows from (5). By definition of $\text{Var}(\mathcal{H}, F, \mathbf{w})$, the result follows. Generalization for the classification loss is obtained in the exact same way to the one used in the proof of Theorem 1.

■

C. Technical Lemmas

Lemma 3 *Any parity function over d variables is realizable by a network with one fully connected layer of width $\tilde{d} > \frac{3d}{2}$ with ReLU activations, and a fully connected output layer with linear activation and a single unit.*

Proof Let the weights entering each of the first $\frac{3d}{2}$ hidden units be set to $\mathbf{v}^*$, and the rest to 0. Further assume that for $i \in [d/2]$, the biases of the first $3i + \{1, 2, 3\}$ units are set to $-(2i - \frac{1}{2})$, $-2i$, $-(2i + \frac{1}{2})$ respectively, and that their weights in the output layer are 1, -2, and 1. It is not hard to see that the weighted sum of those triads of neurons is $\frac{1}{2}$ if $\langle \mathbf{x}, \mathbf{v}^* \rangle = 2i$, and 0 otherwise. Observe that there's such a triad defined for each even number in the range $[d]$. Therefore, the output of this net is 0 if $y = -1$, and $\frac{1}{2}$ otherwise. It is easy to see that scaling of the output layer's weights by 4, and introduction of a -1 bias value to it, results in a perfect predictor. ■

D. Command Lines for Experiments

Our experiments are implemented in a simple manner in python. We use the tensorflow package for optimization.

To run experiment 2.1, use:

```
python ./parity.py d
```

where d is the desired dimension.

To run experiment A, use:

```
python ./dec_vs_e2e.py [e2e|dec] N k
```

where $e2e|dec$ is the desired experiment, N is the desired input dimension, k is the dimension of the predicted distribution.