Sparse Coding and Autoencoders

Akshay Rangamani[*1], Anirbit Mukherjee[†2], Amitabh Basu [‡2], Tejaswini Ganapathy [§3], Ashish Arora [¶1], Sang (Peter) Chin [‖4,5,6], and Trac D. Tran [**1]

[1]ECE Department, Johns Hopkins University
[2]AMS Department, Johns Hopkins University
[3]Salesforce, San Francisco Bay Area
[4]Department of Computer Science , Boston University
[5]Center for Brains, Minds and Machines (CBMM), Dept. of Brain and Cognitive Sciences , MIT
[6]Center of Mathematical Sciences and Applications (CMSA) , Harvard University

[*]Equal Contribution, rangamani.akshay@jhu.edu
[†]Equal Contribution, amukhe14@jhu.edu
[‡]basu.amitabh@jhu.edu
[§]tganapathi@salesforce.com
[¶]aarora8@jhu.edu
[‖]spchin@cs.bu.edu
[**]trac@jhu.edu

Abstract

In *Dictionary Learning* one tries to recover incoherent matrices $A^* \in \mathbb{R}^{n \times h}$ (typically overcomplete and whose columns are assumed to be normalized) and sparse vectors $x^* \in \mathbb{R}^h$ with a small support of size h^p for some $0 < p < 1$ while having access to observations $y \in \mathbb{R}^n$ where $y = A^* x^*$. In this work we undertake a rigorous analysis towards understanding whether gradient based neural training algorithms can solve the dictionary learning problem. We focus on the *Autoencoder* architecture mapping $\mathbb{R}^n \to \mathbb{R}^n$ with a single ReLU activation layer of size h.

Under very mild distributional assumptions on x^*, we prove that the norm of the expected gradient of the standard squared loss function is asymptotically (in sparse code dimension) negligible for *all* points in a small neighborhood of A^*. We support this via experiments using synthetic data. We also conduct experiments to suggest that A^* is a local minimum. Along the way we prove that a layer of ReLU gates can be set up to automatically recover the support of the sparse codes. This property holds independent of the loss function and we believe that it could be of independent interest.

1 Introduction

One of the fundamental themes in learning theory is to consider data being sampled from a generative model and to provide efficient methods to recover the original model parameters exactly or with tight approximation guarantees. Classic examples include learning a mixture of gaussians [28], certain graphical models [5], full rank square dictionaries [35, 13] and overcomplete dictionaries [2, 7, 8, 9] The problem is usually distilled down to a non-convex optimization problem whose solution can be used to obtain the model parameters. With these hard non-convex problems it has been difficult to find any universal view as to why sometimes gradient descent gives very good and sometimes even exact recovery. In recent times progress has been made towards achieving a geometric understanding of the landscape of such non-convex optimization problems [18], [27], [42]. The corresponding question of parameter recovery for neural nets with one layer of activation has been solved in some special cases, [17, 4, 21, 34, 24, 36, 43]. Almost all of these cases are in the supervised setting where it has also been assumed that the labels are being generated from a net of the same architecture as is being trained. In contrast to these works we address an unsupervised learning problem, and possibly more realistically, we do not tie the data generation model (sensing of sparse vectors by an overcomplete incoherent dictionary) to the neural architecture being analyzed except for assuming knowledge of a few parameters about the ground truth. In a related development, it has been shown by two of the authors here in a previous work [6], that for two layer deep nets even the exact global minima can be found deterministically in time polynomial in the data size. This work continues that line of investigation to now make use of generative model assumptions to probe the power of a class of two layer deep nets with ReLU activation.

Here we specialize to the generative model of *dictionary learning/sparse coding* where one receives samples of vectors $y \in \mathbb{R}^n$ that have been generated as $y = A^* x^*$ where $A^* \in \mathbb{R}^{n \times h}$ and $x^* \in \mathbb{R}^h$. We typically assume that the number of non-zero entries in x^* to be no larger than some function of the dimension h and that A^* satisfies certain incoherence properties. The question now is to recover A^* from samples of y. There have been renewed investigations into the hardness of this problem [38] and many former results have recently been reviewed in these lectures [19]. This question has been a cornerstone of learning theory ever since the ground-breaking paper by Olshausen and Field ([31]) (a recent review by the same authors can be found in [32]). Over the years many algorithms have been developed to solve this problem and a detailed comparison among these various approaches can be found in [13].

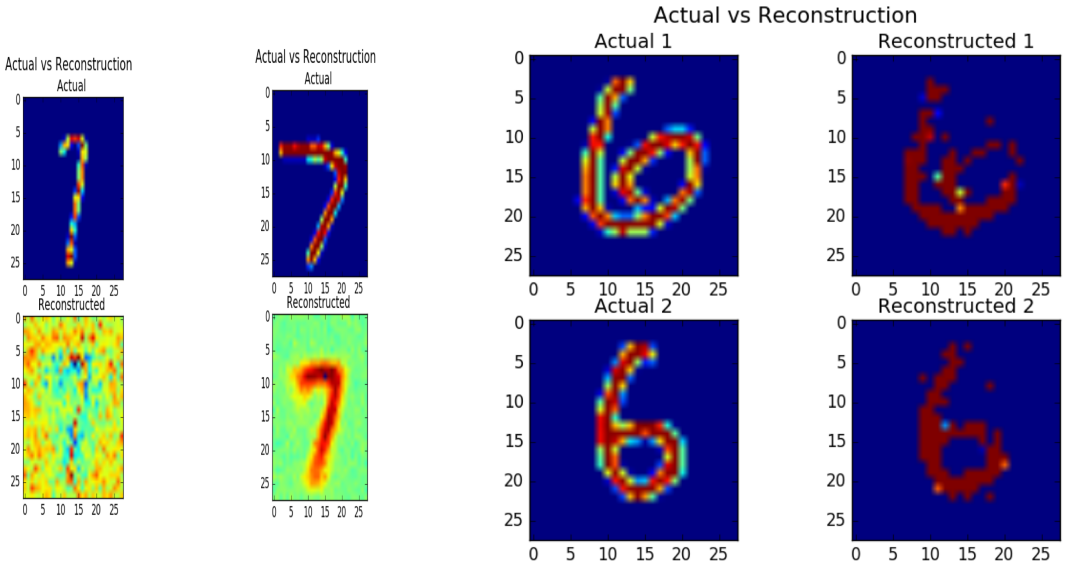
An *autoencoder* is a neural network that maps $\mathbb{R}^n \to \mathbb{R}^n$ with a single hidden layer of Rectified Linear Unit (ReLU) activations. These networks have been used extensively ([11, 12, 33, 40, 41]) in the past for unsupervised feature learning tasks, and have been found to be successful in generating discriminative features

[15]. A number of different autoencoder architectures and regularizers have been proposed which purportedly induce sparsity, at the hidden layer [10, 16, 23, 29]. There has also been some investigation into what autoencoders learn about the data distribution [3].

Olshausen and Field had, as early as 1996, already made the connection between sparse coding and training neural architectures and in today's terminology this problem is very naturally reminiscent of the architecture of an autoencoder [30]. However, to the best of our knowledge, there has not been sufficient progress to rigorously establish whether autoencoders can do sparse coding. In this work, we present our progress towards bridging the above mentioned mathematical gap. To the best of our knowledge, there is no theoretical evidence (even under the usual generative assumptions of sparse coding) that the stationary points of any of the usual squared loss functions (with or without any of the usual regularizers) have any resemblance to the original dictionary that is being sought to be learned. **The main point of this paper is to rigorously prove that for autoencoders with ReLU activation, the standard squared loss function has a neighborhood around the dictionary A^* where the norm of the expected gradient is very small (for large enough sparse code dimension h). Thus, all points in a neighborhood of A^*, including A^*, are all asymptotic critical points of this standard squared loss.** We supplement our theoretical result with experimental evidence for it in Section 6, which also strongly suggests that the standard squared loss function has a local minimum in a neighborhood around A^*. We believe that our results provide theoretical and experimental evidence that the sparse coding problem can be tackled by training autoencoders.

1.1 A motivating experiment on MNIST using TensorFlow

We used TensorFlow [1] to train two ReLU autoencoders mapping $\mathbb{R}^{784} \to \mathbb{R}^{784}$. These networks were trained on a subset of the MNIST dataset of handwritten digits. One of the nets had a single hidden layer of size 10000 and the other one had two hidden layers of size 5000 and 784 (and a fixed identity matrix giving the output from the second layer of activations). In both the cases the weights of the encoder and decoder were maintained as transposes of each other. We trained the autoencoders on the standard squared loss function using RMSProp [37]. The training was done on 6000 images of the digits 6 and 7 from the MNIST dataset. In the following panel we show four pairs (two for each net) of "reconstructed" image i.e output of the trained net when its given as input the "actual" photograph as input.



1 layer Relu Autoencoder

2 layer Relu Autoencoder

In our opinion, the above figures add support to the belief that a single and a double layer ReLU activated $\mathbb{R}^n \to \mathbb{R}^n$ network can learn an implicit high dimensional structure about the handwritten digits dataset. In particular this demonstrates that though adding more hidden layers obviously helps enhance the reconstruction ability, the single hidden layer autoencoder do hold within them significant power for unsupervised learning of representations. Unfortunately analyzing the RMSProp update rule used in the above experiment is currently beyond our analytic means. However, we take inspiration from these experiments to devise a different mathematical set-up which is much more amenable to analysis taking us towards a better understanding of the power of autoencoders.

2 Introducing the neural architecture and the distributional assumptions

For any $n, h \in \{1, 2, ..\}$, an autoencoder is a fully connected $\mathbb{R}^n \to \mathbb{R}^n$ neural network with a single hidden layer of h activations. We focus on networks that use the Rectified Linear Unit (ReLU) activation which is the function $\text{ReLU} : \mathbb{R}^h \to \mathbb{R}^h$ mapping $\vec{x} \mapsto (\max\{0, x_i\})_{i=1}^h$. In this case, the autoencoder can be seen as computing the following function $\hat{y}(W, y, \epsilon)$ as follows,

$$\begin{aligned} r &= \text{ReLU}\left(Wy - \epsilon\right) \\ \hat{y} &= W^\top r \end{aligned} \tag{1}$$

Here $y \in \mathbb{R}^n$ is the input to the autoencoder, $W \in \mathbb{R}^{h \times n}$ is the linear transformation implemented by the first layer, $r \in \mathbb{R}^h$ is the output of the layer of activations, $\epsilon \in \mathbb{R}^h$ is the bias vector and $\hat{y} \in \mathbb{R}^n$ is the output of the autoencoder. Note that we impose the condition that the second layer of weights is simply the transpose of the first layer. We shall define the columns of $W^\top$ (rows of W) as $\{W_i\}_{i=1}^h$.

Assumptions on the dictionary and the sparse code. We assume that our signal y is generated using sparse linear combinations of atoms/vectors of an overcomplete dictionary, i.e., $y = A^* x^*$, where $A^* \in \mathbb{R}^{n \times h}$ is a dictionary, and $x^* \in (\mathbb{R}^{\geq 0})^h$ is a non-negative sparse vector, with at most $k = h^p$ (for some $0 < p < 1$) non zero elements. The columns of the original dictionary A^* (labeled as $\{A_i^*\}_{i=1}^h$) are assumed to be normalized and we parameterize its incoherence property as, $\max_{\substack{i,j=1,..,h \\ i \neq j}} |\langle A_i^*, A_j^* \rangle| \leq \frac{\mu}{\sqrt{n}} = h^{-\xi}$ for some $\xi > 0$.

We assume that the sparse code x^* is sampled from a distribution with the following properties. We fix a set of possible supports of x^*, denoted by $\mathbb{S} \subseteq 2^{[h]}$, where each element of $\mathbb{S}$ has at most $k = h^p$ elements. We consider any arbitrary discrete probability distribution $D_{\mathbb{S}}$ on $\mathbb{S}$ such that the probability $q_1 := \mathbb{P}_{S \sim \mathbb{S}}[i \in S]$ is independent of $i \in [h]$, and the probability $q_2 := \mathbb{P}_{S \in \mathbb{S}}[i, j \in S]$ is independent of $i, j \in [h]$. A special case is when $\mathbb{S}$ is the set of all subsets of size k, and $D_{\mathbb{S}}$ is the uniform distribution on $\mathbb{S}$. For every $S \in \mathbb{S}$ there is a distribution say D_S on $(\mathbb{R}^{\geq 0})^h$ which is supported on vectors whose support is contained in S and which is uncorrelated for pairs of coordinates $i, j \in S$. Further, we assume that the distributions D_S are such that each coordinate i is compactly supported over an interval $[a(h), b(h)]$, where $a(h)$ and $b(h)$ are independent of both i and S but will be functions of h. Moreover, $m_1(h) := \mathbb{E}_{x^* \sim D_S}[x_i^*]$, and $m_2(h) := \mathbb{E}_{x^* \sim D_S}[x_i^{*2}]$ are assumed to be independent of both i and S but allowed to depend on h. For ease of notation henceforth we will keep the h dependence of these variables implicit and refer to them as a, b, m_1 and m_2. All of our results will hold in the special case when a, b, m_1, m_2 are constants (no dependence on h).

3 Main Results

3.1 Recovery of the support of the sparse code by a layer of ReLUs

First we prove the following theorem which precisely quantifies the sense in which a layer of ReLU gates is able to recover the support of the sparse code when the weight matrix of the deep net is close to the original

dictionary. We recall that the size of the support of the sparse vector x^* is $k = h^p$ for some $0 < p < 1$. We also recall the parameters a, b as defining the support of the marginal distribution of each coordinate of x^* and m_1 is the expected value of this marginal distribution (recall that none of these depend on the coordinate or the actual support). These parameters will be referenced in the results below.

Theorem 3.1. Let each column of $W^\top$ be within a δ-ball of the corresponding column of A^*, where $\delta = O\left(h^{-p-\nu^2}\right)$ for some $\nu > 0$, such that $p + \nu^2 < \xi$ (where $h^{-\xi}$ is the coherence parameter). We further assume that $a = \Omega\left(bh^{-\nu^2}\right)$. Let the bias of the hidden layer of the autoencoder, as defined in (1) be $\epsilon = 2m_1 k\left(\delta + \frac{\mu}{\sqrt{n}}\right)$. Let r be the vector defined in (1). Then $r_i \neq 0$ if $i \in \text{supp}(x^*)$, and $r_i = 0$ if $i \notin \text{supp}(x^*)$ with probability at least $1 - \exp\left(-\frac{2h^p m_1^2}{(b-a)^2}\right)$ (with respect to the distribution on x^*).

As long as $\frac{h^p m_1^2}{(b-a)^2}$ is large, i.e., an increasing function of h, we can interpret this as saying that the probability of the adverse event is small, and we have successfully achieved support recovery at the hidden layer in the limit of large sparse code dimension.

3.2 Asymptotic Criticality of the Autoencoder around A^*

In this work we analyze the following standard squared loss function for the autoencoder,

$$L = \frac{1}{2}||\hat{y} - y||^2 \tag{2}$$

In the above we continue to use the variables as defined in equation 1. If we consider a generative model in which A^* is a square, orthogonal matrix and x^* is a non-negative vector (not necessarily sparse), it is easily seen that the standard squared reconstruction error loss function for the autoencorder has a global minimum at $W = A^{*\top}$. In our generative model, however, A^* is an incoherent and overcomplete dictionary.

Theorem 3.2. (**The Main Theorem**) Assume that the hypotheses of Theorem 3.1 hold, and $p < \min\{\frac{1}{2}, \nu^2\}$ (and hence $\xi > 2p$). Further, assume the distribution parameters satisfy $\exp\left(\frac{h^p m_1^2}{2(b-a)^2}\right)$ is superpolynomial in h (which holds, for example, when m_1, a, b are $O(1)$). Then for $i = 1, \ldots, h$,

$$\left\| \mathbb{E}\left[\frac{\partial L}{\partial W_i}\right] \right\|_2 \leq o\left(\frac{\max\{m_1^2, m_2\}}{h^{1-p}}\right).$$

Roadmap. We present the proof of the support recovery result, i.e., Theorem 3.1, in Section 4. Section 5 gives the proof of our main result, Theorem 3.2. The argument rests on two critical lemmas (Lemmas 5.1 and 5.2), whose proofs appear in the Supplementary material. In Section 6, we run simulations to verify Theorem 3.2. We also run experiments that strongly suggest that the standard squared loss function has a local minimum in a neighborhood around A^*.

4 A Layer of ReLU Gates can Recover the Support of the Sparse Code (Proof of Theorem 3.1)

Most sparse coding algorithms are based on an alternating minimization approach, where one iteratively finds a sparse code based on the current estimate of the dictionary, and then uses the estimated sparse code to update the dictionary. The analogue of the sparse coding step in an autoencoder, is the passing through the hidden layer of activations of a certain affine transformation (W which behaves as the current estimate of the dictionary) of the input vectors. We show that under certain stochastic assumptions, the hidden layer of ReLU gates in an autoencoder recovers with high probability the support of the sparse vector which corresponds to the present input.

Proof of Theorem 3.1. From the model assumptions, we know that the dictionary A^* is incoherent, and has unit norm columns. So, $|\langle A_i^*, A_j^* \rangle| \leq \frac{\mu}{\sqrt{n}}$ for all $i \neq j$, and $||A_i^*|| = 1$ for all i. This means that for $i \neq j$,

$$\begin{aligned} |\langle W_i, A_j^* \rangle| &= |\langle W_i - A_i^*, A_j^* \rangle| + |\langle A_i^*, A_j^* \rangle| \\ &\leq ||W_i - A_i^*||_2 ||A_j^*||_2 + \frac{\mu}{\sqrt{n}} \leq (\delta + \frac{\mu}{\sqrt{n}}) \end{aligned} \tag{3}$$

Otherwise for $i = j$,

$$\langle W_i, A_i^* \rangle = \langle W_i - A_i^*, A_i^* \rangle + \langle A_i^*, A_i^* \rangle = \langle W_i - A_i^*, A_i^* \rangle + 1,$$

and thus,

$$1 - \delta \leq \langle W_i, A_i^* \rangle \leq 1 + \delta, \tag{4}$$

where we use the fact that $|\langle W_i - A_i^*, A_i^* \rangle| \leq \delta$.

Let $y = A^* \mathbf{x}^*$ and let S be the support of $\mathbf{x}^*$. Then we define the input to the ReLU activation $Q - \epsilon = W\mathbf{y} - \epsilon$ as

$$\begin{aligned} Q_i &= \sum_{j \in S} \langle W_i, A_j^* \rangle x_j^* \\ &= \langle W_i, A_i^* \rangle x_i^* \mathbf{1}_{i \in S} + \sum_{j \in S \setminus i} \langle W_i, A_j^* \rangle x_j^* \\ &= \langle W_i, A_i^* \rangle x_i^* \mathbf{1}_{i \in S} + Z_i. \end{aligned}$$

First we try to get bounds on Q_i when $i \in \text{supp}(x^*)$. From our assumptions on the distribution of x_i^* we have, $0 \leq a \leq x_i^* \leq b$ and $\mathbb{E}[x_i^*] = m_1$ for all i in the support of x^*. For $i \in \text{supp}(x^*)$,

$$\begin{aligned} & Q_i = \langle W_i, A_i^* \rangle x_i^* + Z_i \\ \implies & Q_i \geq (1 - \delta) a + Z_i \end{aligned}$$

where we use (4). Using (3), Z_i has the following bounds:

$$-bk \left(\delta + \frac{\mu}{\sqrt{n}} \right) \leq Z_i \leq bk \left(\delta + \frac{\mu}{\sqrt{n}} \right)$$

Plugging in the lower bound for Z_i and the proposed value for the bias, we get

$$Q_i - \epsilon \geq (1 - \delta) a - bk \left(\delta + \frac{\mu}{\sqrt{n}} \right) - 2 m_1 k \left(\delta + \frac{\mu}{\sqrt{n}} \right)$$

For $Q_i - \epsilon \geq 0$, we need:

$$a \geq \frac{(b+2m_1)\left(\delta + \frac{\mu}{\sqrt{n}}\right)k}{1-\delta}$$

Now plugging in the values for the various quantities, $\frac{\mu}{\sqrt{n}} = h^{-\xi}$ and $k = h^p$ and $\delta = O\left(h^{-p-\nu^2}\right)$, if we have $a = \Omega\left(bh^{-\nu^2}\right)$, then $Q_i - \epsilon \geq 0$.

Now, for $i \notin \text{supp}(x^*)$ we would like to analyze the following probability:

$$\Pr[Q_i - \epsilon \geq 0 | i \notin \text{supp}(x^*)]$$

We first simplify the quantity $\Pr[Q_i - \epsilon \geq 0 | i \notin \text{supp}(x^*)]$ as follows

$$\begin{aligned}\Pr[Q_i \geq \epsilon | i \notin \text{supp}(x^*)] &= \Pr[Z_i \geq \epsilon] \\ &= \Pr\left[\sum_{j \in S \setminus i} \langle W_i, A_j^* \rangle x_j^* \geq \epsilon\right]\end{aligned}$$

Using the Chernoff's bound, we can obtain

$$\begin{aligned}\Pr[Z_i \geq \epsilon] &\leq \inf_{t \geq 0} e^{-t\epsilon} \mathbb{E}\left[\prod_{j \in S \setminus i}\left[e^{t\langle W_i, A_j^* \rangle x_j^*}\right]\right] \\ &= \inf_{t \geq 0} e^{-t\epsilon} \prod_{j \in S \setminus i} \mathbb{E}\left[e^{t\langle W_i, A_j^* \rangle x_j^*}\right] \\ &\leq \inf_{t \geq 0} e^{-t\epsilon} \mathbb{E}^k\left[e^{t\left(\delta + \frac{\mu}{\sqrt{n}}\right)x_j^*}\right] \\ &\leq \inf_{t \geq 0} e^{-t\epsilon}\left(e^{t\left(\delta+\frac{\mu}{\sqrt{n}}\right)m_1} e^{\frac{t^2\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(b-a)^2}{8}}\right)^k\end{aligned}$$

where the second inequality follows from (3) and the fact that t and x_i^* are both nonnegative, and the third inequality follows from Hoeffding's Lemma. Next, we also have

$$\begin{aligned}\Pr[Z_i \geq \epsilon] &\leq \inf_{t \geq 0} e^{-t\left(\epsilon - k\left(\delta+\frac{\mu}{\sqrt{n}}\right)m_1\right) + t^2 \frac{k}{8}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(b-a)^2} \\ &= e^{-\frac{(\epsilon - k(\delta+\frac{\mu}{\sqrt{n}})m_1)^2}{\frac{k}{2}(\delta+\frac{\mu}{\sqrt{n}})^2(b-a)^2}}.\end{aligned}$$

Finally, since $k = h^p$ and $\epsilon = 2m_1 k\left(\delta + \frac{\mu}{\sqrt{n}}\right)$, we have

$$\exp\left(-\frac{2(\epsilon - km_1(\delta + \frac{\mu}{\sqrt{n}}))^2}{h^p(\delta + \frac{\mu}{\sqrt{n}})^2(b-a)^2}\right) = \exp\left(-\frac{2h^p m_1^2}{(b-a)^2}\right)$$

□

5 Criticality of a neighborhood of A^* (Proof of Theorem 3.2)

It turns out that the expectation of the full gradient of the loss function (2) is difficult to analyze directly. Hence corresponding to the true gradient with respect to the $i^{\text{th}}-$column of $W^\top$ we create a proxy, denoted by $\widehat{\nabla_i L}$), by replacing in the expression for the true expectation $\nabla_i L = \mathbb{E}\left[\frac{\partial L}{\partial W_i}\right]$ every occurrence of the random variable $\text{Th}(W_i^\top y - \epsilon_i) = \text{Th}(W_i^\top A^* x^* - \epsilon_i)$ by the indicator random variable $\mathbf{1}_{i\in \text{supp}(x^*)}$. This proxy is shown to be a good approximant of the expected gradient in the following lemma.

Lemma 5.1. Assume that the hypotheses of Theorem 3.1 hold and additionally let b be bounded by a polynomial in h. Then we have for each i (indexing the columns of $W^\top$),

$$\left|\left|\widehat{\nabla_i L} - \mathbb{E}\left[\frac{\partial L}{\partial W_i}\right]\right|\right|_2 \leq \text{poly}(h)\exp\left(-\frac{h^p m_1^2}{2(b-a)^2}\right)$$

Proof. This lemma has been proven in Section 8 of the Appendix. □

Lemma 5.2.
Assume that the hypotheses of Theorem 3.1 hold, and $p < \min\{\frac{1}{2}, \nu^2\}$ (and hence $\xi > 2p$). Then for each i indexing the columns of $W^\top$, there exist real valued functions α_i and β_i, and a vector e_i such that $\widehat{\nabla_i L} = \alpha_i W_i - \beta_i A_i^* + e_i$, and

$$\begin{aligned}
\alpha_i &= \Theta(m_2 h^{p-1}) + o(m_1^2 h^{p-1}) \\
\beta_i &= \Theta(m_2 h^{p-1}) + o(m_1^2 h^{p-1}) \\
\alpha_i - \beta_i &= o(\max\{m_1^2, m_2\} h^{p-1}) \\
||e_i||_2 &= o(\max\{m_1^2, m_2\} h^{p-1})
\end{aligned}$$

Proof. In subsection 5.1 we first get explicit forms of the above defined quantities α_i, β_i and e_i. Then the proof is completed by estimating these which is done in Appendix 9 □

With the above asymptotic results, we are in a position to assemble the proof of Theorem 3.2.

Proof of Theorem 3.2. Consider any i indexing the columns of $W^\top$. Recall the definition of the proxy gradient $\widehat{\nabla_i L}$ at the beginning of this section. Let us define $\gamma_i = \widehat{\nabla_i L} - \mathbb{E}\left[\frac{\partial L}{\partial W_i}\right]$. Using α_i, β_i and e_i as defined in Lemma 5.2, we can write the expectation of the true gradient as, $\mathbb{E}\left[\frac{\partial L}{\partial W_i}\right] = \alpha_i W_i - \beta_i A_i^* + e_i - \gamma_i$. Further, by Lemma 5.1,

$$\|\gamma_i\| \leq \text{poly}(h)\exp\left(-\frac{h^p m_1^2}{2(b-a)^2}\right).$$

Since $\exp\left(\frac{h^p m_1^2}{2(b-a)^2}\right)$ is superpolynomial in h, we obtain

$$\begin{aligned}
\left|\left|\mathbb{E}\left[\frac{\partial L}{\partial W_i}\right]\right|\right|_2 &= ||\alpha_i W_i - \beta_i A_i^* + e_i - \gamma_i||_2 \\
&= ||\alpha_i(W_i - A_i^*) + (\alpha_i - \beta_i)A_i^* + e_i - \gamma_i||_2 \\
&\leq |\alpha_i|||W_i - A_i^*||_2 + |\alpha_i - \beta_i| + ||e_i - \gamma_i||_2 \\
&\leq \frac{\Theta(m_2 h^{p-1})}{h^{2p+\theta^2}} + o(\max\{m_1^2, m_2\} h^{p-1}) \\
&+ o(\max\{m_1^2, m_2\} h^{p-1}) \\
&= o(\max\{m_1^2, m_2\} h^{p-1})
\end{aligned}$$

□

5.1 Simplifying the proxy gradient of the autoencoder under the sparse-coding generative model - to get explicit forms of the coefficients α, β and e as required towards proving Lemma 5.2

To recap we imagine being given as input signals $y \in \mathbb{R}^n$ (imagined as column vectors), which are generated from an overcomplete dictionary $A^* \in \mathbb{R}^{n \times h}$ of fixed incoherence. Let $x^* \in \mathbb{R}^h$ (imagined as column vectors) be the sparse code that generates y. The model of the autoencoder that we now have is $\hat{y} = W^\top \text{ReLU}(Wy - \epsilon)$. W is a $h \times n$ matrix and the i^{th} column of $W^\top$ is to be denoted as the column vector W_i.

Using the above notation the squared loss of the autoencoder is $\frac{1}{2}||\hat{y} - y||^2$. But we introduce a dummy constant $D = 1$ to be multiplied to y because this helps read the complicated equations that would now follow. This marker helps easily spot those terms which depend on the sensing of x^* (those with a factor of D) as opposed to the terms which are "purely" dependent on the neural net (those without the factor of D). Thus we think of the squared loss L of our autoencoder as,

$$L = \frac{1}{2}||\hat{y} - Dy||^2 = \frac{1}{2}(W^\top \text{ReLU}(Wy - \epsilon) - Dy)^\top (W^\top \text{ReLU}(Wy - \epsilon) - Dy) = \frac{1}{2} f^T f$$

where we have defined $f \in \mathbb{R}^n$ as,

$$f = W^\top \text{ReLU}(Wy - \epsilon) - Dy$$

Then we have,

$$J_{W_i}(f)_{ab} = \frac{\partial f_a}{\partial W_{ib}} = \text{ReLU}(W_i^\top y - \epsilon)\delta_{ab} + \text{Th}(W_i^T y - \epsilon) W_{ia} y_b$$

In the form of a $n \times n$ derivative matrix this means,

$$J_{W_i}(f) = \left[\frac{\partial f_a}{\partial W_{ib}}\right] = \text{ReLU}(W_i^\top y - \epsilon) I + \text{Th}(W_i^\top y - \epsilon) W_i y^\top$$

This helps us write,

$$\begin{aligned}
\frac{\partial L}{\partial W_i} &= J_{W_i}(f))^\top f \\
&= (\text{ReLU}(W_i^\top y - \epsilon) I + \text{Th}(W_i^\top y - \epsilon) W_i y^\top)^\top [W^\top \text{ReLU}(Wy - \epsilon) - Dy] \\
&= \text{Th}(W_i^\top y - \epsilon_i) \left[(W_i^\top y - \epsilon_i) I + y W_i^\top\right] \left(\sum_{j=1}^{h} \text{ReLU}(W_j^\top y - \epsilon_j) W_j - Dy\right)
\end{aligned}$$

Now going over to the proxy gradient $\widehat{\nabla_i L}$ corresponding to this term and we define the vector G_i as,

$$\begin{aligned}
\widehat{\nabla_i L} &= \mathbb{E}_{S \in \mathbb{S}} \left[\mathbf{1}_{i \in S} \times \mathbb{E}_{x_S^*} \left[\left[(W_i^\top y - \epsilon_i) I + y W_i^\top\right] \left(\sum_{j \in S} (W_j^\top y - \epsilon_j) W_j - Dy\right) \right] \right] \\
&= \mathbb{E}_{S \in \mathbb{S}} \left[\mathbf{1}_{i \in S} \times G_i\right]
\end{aligned}$$

Thus we have,

$$
\begin{aligned}
G_i &= \mathbb{E}_{x_S^*}\left[\left[(W_i^\top A^* x^* - \epsilon_i)I + (A^* x^*)W_i^\top\right]\left(\sum_{j\in S}(W_j^\top A^* x^* - \epsilon_j)W_j - DA^* x^*\right)\right] \\
&= \underbrace{\mathbb{E}_{x_S^*}\left[(W_i^\top A^* x^* - \epsilon_i)\left(\sum_{j\in S}(W_j^\top A^* x^* - \epsilon_j)W_j - DA^* x^*\right)\right]}_{\text{Term 1}} \\
&\quad + \underbrace{\mathbb{E}_{x_S^*}\left[(A^* x^*)W_i^\top\left(\sum_{j\in S}(W_j^\top A^* x^* - \epsilon_j)W_j - DA^* x^*\right)\right]}_{\text{Term 2}}
\end{aligned}
$$

which can be decomposed into the following convenient parts,

$$
\begin{aligned}
G_i &= \underbrace{\mathbb{E}_{x_S^*}\left[\sum_{j\in S}\epsilon_i\epsilon_j W_j - \sum_{j,k\in S}\epsilon_i(W_j^\top A_k^*)W_j x_k^* - \sum_{j,k\in S}\epsilon_j(W_i^\top A_k^*)W_j x_k^* + \sum_{j,k,l\in S}(W_i^\top A_k^*)(W_j^\top A_l^*)W_j x_l^* x_k^*\right]}_{\text{From Term 1}} \\
&\quad + \underbrace{\mathbb{E}_{x_S^*}\left[-D\sum_{j,k\in S}(W_i^\top A_k^*)A_j^* x_k^* x_j^* + D\sum_{j\in S}\epsilon_i A_j^* x_j^*\right]}_{\text{From Term 1}} + \underbrace{\mathbb{E}_{x_S^*}\left[-D\sum_{j,k\in S}(A_k^{*\top} W_i)A_j^* x_k^* x_j^*\right]}_{\text{From Term 2}} \\
&\quad + \underbrace{\mathbb{E}_{x_S^*}\left[-\sum_{j,k\in S}\epsilon_j A_k^*(W_i^\top W_j)x_k^*\right]}_{\text{From Term 2}} + \underbrace{\mathbb{E}_{x_S^*}\left[\sum_{j,k,l\in S}(W_i^\top W_j)(W_j^\top A_l^*)A_k^* x_k^* x_l^*\right]}_{\text{From Term 2}}
\end{aligned}
$$

Now we invoke the distributional assumption about i.i.d sampling of the coordinates for a fixed support and the definition of m_1 and m_2 to write, $\mathbb{E}_{x_S^*}[x_i^* x_j^*] = \mathbb{E}^2_{x_S^*}[x_i^*] = m_1^2$ for all $i \neq j$ and for $i = j$, $m_2 = \mathbb{E}_{x_S^*}[x_i^* x_j^*]$. Thus we get,

$$
\begin{aligned}
G_i = & \underbrace{\sum_{j\in S}\epsilon_i\epsilon_j W_j - m_1\sum_{j,k\in S}(W_j^\top A_k^*)W_j\epsilon_i - m_1\sum_{j,k\in S}\epsilon_j(W_i^\top A_k^*)W_j}_{G_i^1 \text{ From Term 1}} \\
& \underbrace{+ m_2\sum_{j,k\in S}(W_i^\top A_k^*)(W_j^\top A_k^*)W_j + m_1^2\sum_{\substack{j,k,l\in S\\ k\neq l}}(W_i^\top A_k^*)(W_j^\top A_l^*)W_j}_{G_i^2 \text{ From Term 1}} \\
& + \underbrace{\left[-Dm_1^2\sum_{\substack{j,k\in S\\ j\neq k}}(W_i^\top A_k^*)A_j^* - Dm_2\sum_{j\in S}(W_i^\top A_j^*)A_j^* + m_1 D\sum_{j\in S}\epsilon_i A_j^*\right]}_{G_i^3 \text{ From Term 1}} \\
& - \underbrace{\left[Dm_1^2\sum_{\substack{j,k\in S\\ j\neq k}}(A_k^{*\top}W_i)A_j^* + Dm_2\sum_{j\in S}(A_j^{*\top}W_i)A_j^*\right]}_{G_i^4 \text{ From Term 2}} \\
& \underbrace{- m_1\left[\sum_{j,k\in S}\epsilon_j(W_i^\top W_j)A_k^*\right] + \left[m_2\sum_{j,k\in S}(W_i^\top W_j)(W_j^\top A_k^*)A_k^* + m_1^2\sum_{\substack{j,k,l\in S\\ k\neq l}}(W_i^\top W_j)(W_j^\top A_l^*)A_k^*\right]}_{G_i^5 \text{ From Term 2}}
\end{aligned}
$$

Each term in the above sum is a vector. Now we separate out from the sums the terms which are in the directions of W_i or A_i^* and the rest. We remember that this is being under the condition that $i \in S$. To make this easy to read we do this separation for each line of the above equation separately in a different equation block. Also inside every block we do the separation for each summation term in a separate line.

$$
\begin{aligned}
G_i^1 &= \sum_{j\in S} \epsilon_i \epsilon_j W_j - m_1 \sum_{j,k\in S} (W_j^\top A_k^*) W_j \epsilon_i - m_1 \sum_{j,k\in S} \epsilon_j (W_i^\top A_k^*) W_j \\
&= \left[\epsilon_i^2 W_i + \sum_{\substack{j\in S \\ j\neq i}} \epsilon_i \epsilon_j W_j \right] \\
&\quad - m_1 \left[\sum_{k\in S} \epsilon_i (W_i^\top A_k^*) W_i + \sum_{\substack{j,k\in S \\ j\neq i}} (W_j^\top A_k^*) W_j \epsilon_i \right] \\
&\quad - m_1 \left[\sum_{k\in S} \epsilon_i (W_i^\top A_k^*) W_i + \sum_{\substack{j,k\in S \\ j\neq i}} \epsilon_j (W_i^\top A_k^*) W_j \right]
\end{aligned}
$$

$$
\begin{aligned}
G_i^2 &= m_2 \sum_{j,k\in S} (W_i^\top A_k^*)(W_j^\top A_k^*) W_j + m_1^2 \sum_{\substack{j,k,l\in S \\ k\neq l}} (W_i^\top A_k^*)(W_j^\top A_l^*) W_j \\
&= m_2 \left[\sum_{k\in S} (W_i^\top A_k^*)(W_i^\top A_k^*) W_i + \sum_{\substack{j,k\in S \\ j\neq i}} (W_i^\top A_k^*)(W_j^\top A_k^*) W_j \right] \\
&\quad + m_1^2 \left[\sum_{\substack{k,l\in S \\ k\neq l}} (W_i^\top A_k^*)(W_i^\top A_l^*) W_i + \sum_{\substack{j,k,l\in S \\ j\neq i \\ k\neq l}} (W_i^\top A_k^*)(W_j^\top A_l^*) W_j \right]
\end{aligned}
$$

$$
\begin{aligned}
G_i^3 &= -D\left[m_1^2\sum_{\substack{j,k\in S\\ j\neq k}}(W_i^\top A_k^*)A_j^* + m_2\sum_{j\in S}(W_i^\top A_j^*)A_j^* - m_1\sum_{j\in S}\epsilon_i A_j^*\right]\\
&= -D\left[m_1^2\sum_{\substack{k\in S\\ k\neq i}}(W_i^\top A_k^*)A_i^* + m_1^2\sum_{\substack{j,k\in S\\ j\neq i\\ j\neq k}}(W_i^\top A_k^*)A_j^*\right]\\
&\quad - D\left[m_2(W_i^\top A_i^*)A_i^* + m_2\sum_{\substack{j\in S\\ j\neq i}}(W_i^\top A_j^*)A_j^*\right]\\
&\quad - D\left[-m_1\epsilon_i A_i^* - m_1\sum_{\substack{j\in S\\ j\neq i}}\epsilon_i A_j^*\right]
\end{aligned}
$$

$$
\begin{aligned}
G_i^4 &= -\left[Dm_1^2\sum_{\substack{j,k\in S\\ j\neq k}}(A_k^{*\top}W_i)A_j^* + Dm_2\sum_{j\in S}(A_j^{*\top}W_i)A_j^*\right]\\
&= -D\left[m_1^2\sum_{\substack{k\in S\\ k\neq i}}(A_k^{*\top}W_i)A_i^* + m_1^2\sum_{\substack{j,k\in S\\ j\neq k\\ j\neq i}}(A_k^{*\top}W_i)A_j^*\right]\\
&\quad - D\left[m_2(A_i^{*\top}W_i)A_i^* + m_2\sum_{\substack{j\in S\\ j\neq i}}(A_j^{*\top}W_i)A_j^*\right]
\end{aligned}
$$

$$G_i^5 = -m_1\left[\sum_{j,k\in S}\epsilon_j(W_i^\top W_j)A_k^*\right] + \left[m_2\sum_{j,k\in S}(W_i^\top W_j)(W_j^\top A_k^*)A_k^* + m_1^2\sum_{\substack{j,k,l\in S\\k\neq l}}(W_i^\top W_j)(W_j^\top A_l^*)A_k^*\right]$$

$$= -m_1\sum_{j\in S}\epsilon_j(W_i^\top W_j)A_i^* - m_1\sum_{\substack{j,k\in S\\k\neq i}}\epsilon_j(W_i^\top W_j)A_k^*$$

$$+ m_2\sum_{j\in S}(W_i^\top W_j)(W_j^\top A_i^*)A_i^* + m_2\sum_{\substack{j,k\in S\\k\neq i}}(W_i^\top W_j)(W_j^\top A_k^*)A_k^*$$

$$+ m_1^2\sum_{\substack{j,l\in S\\l\neq i}}(W_i^\top W_j)(W_j^\top A_l^*)A_i^* + m_1^2\sum_{\substack{j,k,l\in S\\k\neq i,l}}(W_i^\top W_j)(W_j^\top A_l^*)A_k^*$$

Thus combining the $G_i^1,\ldots,G_i^5$ above we have, $\widehat{\nabla_i L} = \alpha_i W_i - \beta_i A_i^* + e_i$ where,

$$\alpha_i = \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{m_2\sum_{k\in S}(W_i^\top A_k^*)(W_i^\top A_k^*) + m_1^2\sum_{\substack{k,l\in S\\k\neq l}}(W_i^\top A_k^*)(W_i^\top A_l^*) - 2m_1\sum_{k\in S}\epsilon_i(W_i^\top A_k^*) + \epsilon_i^2\right\}\right]$$

$$\beta_i = \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{2Dm_1^2\sum_{\substack{k\in S\\k\neq i}}(W_i^\top A_k^*) + 2Dm_2(W_i^\top A_i^*) - Dm_1\epsilon_i + m_1\sum_{j\in S}\epsilon_j(W_i^\top W_j)\right.\right.$$

$$\left.\left. - m_2\sum_{j\in S}(W_i^\top W_j)(W_j^\top A_i^*) - m_1^2\sum_{\substack{j,l\in S\\l\neq i}}(W_i^\top W_j)(W_j^\top A_l^*)\right\}\right]$$

$$e_i = \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{\sum_{\substack{j\in S\\j\neq i}}\epsilon_i\epsilon_j W_j - m_1\sum_{\substack{j,k\in S\\j\neq i}}\epsilon_i(W_j^\top A_k^*)W_j - m_1\sum_{\substack{j,k\in S\\j\neq i}}\epsilon_j(W_i^\top A_k^*)W_j\right.\right.$$

$$+ m_2\sum_{\substack{j,k\in S\\j\neq i}}(W_i^\top A_k^*)(W_j^\top A_k^*)W_j + m_1^2\sum_{\substack{j,k,l\in S\\j\neq i\\k\neq l}}(W_i^\top A_k^*)(W_j^\top A_l^*)W_j$$

$$- 2Dm_1^2\sum_{\substack{j,k\in S\\j\neq i\\j\neq k}}(W_i^\top A_k^*)A_j^* - 2Dm_2\sum_{\substack{j\in S\\j\neq i}}(W_i^\top A_j^*)A_j^* + Dm_1\sum_{\substack{j\in S\\j\neq i}}\epsilon_i A_j^*$$

$$\left.\left. - m_1\sum_{\substack{j,k\in S\\k\neq i}}\epsilon_j(W_i^\top W_j)A_k^* + m_2\sum_{\substack{j,k\in S\\k\neq i}}(W_i^\top W_j)(W_j^\top A_k^*)A_k^* + m_1^2\sum_{\substack{j,k,l\in S\\k\neq i,l}}(W_i^\top W_j)(W_j^\top A_l^*)A_k^*\right\}\right]$$

Thus we have laid the groundwork of finding a convenient decomposition of the proxy-gradient in terms of the quantities α_i, β_i and e_i. Now we can go over to Appendix 9 where their magnitudes are estimated towards completing the proof of Lemma 5.2.

6 Simulations

We conduct some experiments on synthetic data in order to check whether the gradient norm is indeed small within the columnwise δ-ball of A^*. We also make some observations about the landscape of the squared

loss function, which has implications for being able to recover the ground-truth dictionary A^*.

Data Generation Model We generate random gaussian dictionaries (A^*) of size $n \times h$ where $n = 50$, and $h = 256, 512, 1024, 2048$ and 4096. For each h, we generate a dataset containing $N = 5000$ sparse vectors with h^p non-zero entries, for various $p \in [0.01, 0.5]$. In our experiments, the coherence parameter ξ was approximately 0.1. The support of each sparse vector x^* is drawn uniformly from all sets of indices of size h^p, and the non-zero entries in the sparse vectors are drawn from a uniform distribution between $a = 1$ and $b = 10$. Once we have generated the sparse vectors, we collect them in a matrix $X^* \in \mathbb{R}^{h \times N}$ and then compute the signals $Y = A^*X^*$. We set up the autoencoder as defined through equation 1. We analyze the squared loss function in (2) and its gradient with respect to a column of W through their empirical averages over the signals in Y.

Results Once we have generated the data, we compute the empirical average of the gradient of the loss function in (2) at 200 random points which are columnwise $\frac{\delta}{2} = \frac{1}{2h^{2p}}$ away from A^*. We average the gradient over the 200 points which are all at the same distance from A^*, and compare the average column norm of the gradient to h^{p-1}. Our experimental results shown in Table 1 demonstrate that the average column norm of the gradient is of the order of h^{p-1} (and thus falling with h for any fixed p) as expected from Theorem 3.2.

h \ p	0.01	0.02	0.05	0.1	0.2
256	(0.0137, 0.0041)	(0.0138, 0.0044)	(0.0126, 0.0052)	(0.0095, 0.0068)	(0.0284, 0.0118)
512	(0.0058, 0.0021)	(0.0058, 0.0022)	(0.0054, 0.0027)	(0.0071, 0.0036)	(0.0104, 0.0068)
1024	(0.0025, 0.0010)	(0.0024, 0.0011)	(0.0026, 0.0014)	(0.0079, 0.0020)	(0.0078, 0.0039)
2048	(0.0011, 0.0005)	(0.0012, 0.0006)	(0.0025, 0.0007)	(0.0031, 0.0010)	(0.0032, 0.0022)
4096	(0.0006, 0.0003)	(0.0012, 0.0003)	(0.0013, 0.0004)	(0.0026, 0.0006)	(0.0020, 0.0013)

h \ p	0.3	0.5
256	(0.0464, 0.0206)	(0.0343, 0.0625)
512	(0.0214, 0.0127)	(0.0028, 0.0442)
1024	(0.0099, 0.0078)	(0.00, 0.0313)
2048	(0.0036, 0.0048)	(0.00, 0.0221)
4096	(0.0008, 0.0030)	(0.00, 0.0156)

Table 1: Average gradient norm for points that are columnwise $\frac{\delta}{2}$ away from A^*. For each h and p we report $\left(||\mathbb{E}\left[\frac{\partial L}{\partial W_i}\right]||, h^{p-1}\right)$. We note that the gradient norm and h^{p-1} are of the same order, and for any fixed p the gradient norm is decreasing with h as expected from Theorem 3.2

We also plot the squared loss of the autoencoder along a randomly chosen direction to understand the geometry of the landscape of the loss function around A^*. We draw a matrix ΔW from a standard normal distribution, and normalize its columns. We then plot $f(t) = L((A^* + t\Delta W)^\top)$, as well as the gradient norm averaged over all the columns. For purposes of illustration, we show these plots for $p = 0.01, 0.1, 0.3$. The plots for $h = 256$ are in Figure 1, and those for $h = 4096$ in Figure 2. From the plots for $p = 0.01$ and 0.1, we can observe that the loss function value, and the gradient norm keep decreasing as we get close to A^*. Figure 1 and 2 are representative of the shapes obtained for every direction, ΔW that we checked. This suggests that A^* might conveniently lie at the bottom of a well in the landscape of the loss function. For the value of $p = 0.3$, (which is much larger than the coherence parameter ξ), Theorem 3.1 is no longer valid. We see that the value of the loss function decreases a little as we move away from A^*, and then increases. We suspect that A^* is here in a region where $\text{ReLU}(A^{*\top}y - \epsilon) = 0$, which means the function is flat in a small neighborhood of A^*.

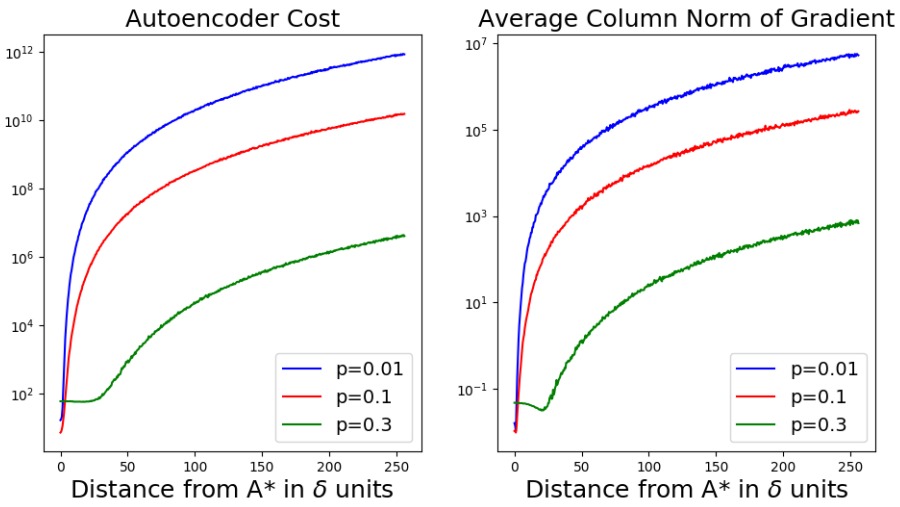


Figure 1: Loss function plot for $h = 256, n = 50$

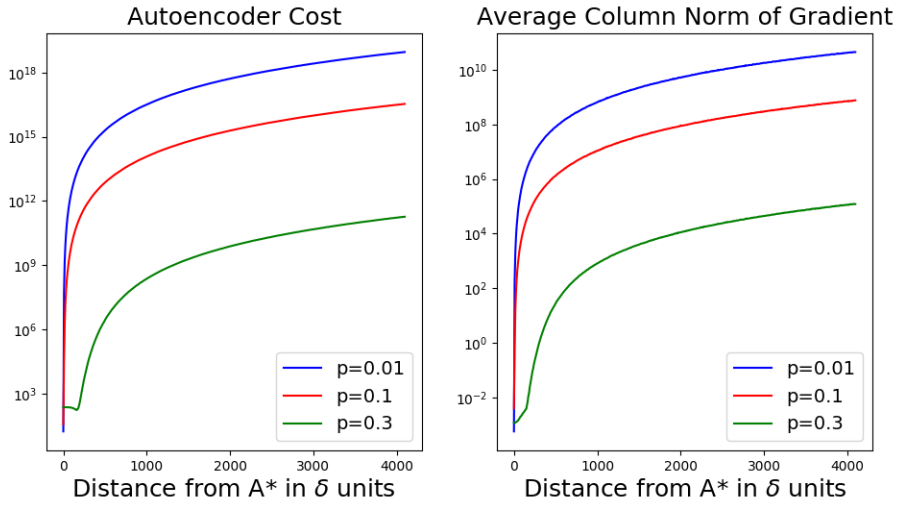


Figure 2: Loss function plot for $h = 4096, n = 50$

We also tried to minimize the squared loss of the autoencoder using gradient descent. In these experiments, we initialized $W^\top$ far away from A^* (precisely at a columnwise distance of $\frac{h}{5} \times \delta$), and did gradient descent until the gradient norm dropped below a factor of 2×10^{-5} of the initial norm of the gradient. We then computed the average columnwise distance between $W_{\text{final}}^\top$ and A^*, and report the $\%$ decrease in the average columnwise distance from the initial point. These results are reported in Table 2 below. These experiments suggest that there is a neighborhood of A^* (the radius of which is increasing with h), such that gradient descent initialized at the edge of that neighborhood, greatly reduces the average columnwise distance between $W^\top$ and A^*.

h	$p = 0.05$	$p = 0.1$
256	97.7%	96.9%
512	98.6%	98.2%
1024	99%	98.8%
2048	99.2%	99%
4096	99.4%	99.2%

Table 2: Fraction of initial columnwise distance covered by the gradient descent procedure

7 Conclusion

In this paper we have undertaken a rigorous analysis of the loss function of the squared loss of an autoencoder when the data is assumed to be generated by sensing of sparse high dimensional vectors by an overcomplete dictionary. **We have shown that the expected gradient of this loss function is very close to zero in a neighborhood of the generating overcomplete dictionary.**

Our simulations complement this theoretical result by providing further empirical support. Firstly, they show that the gradient norm in this $\delta-$ball of A^* indeed falls with h and is of the same order as $\frac{1}{h^{1-p}}$ as expected from our proof. Secondly, the experiments also strongly suggest ranges of values of h and p where A^* is a local minima of this loss function and that it has a neighborhood where the reconstruction error is low.

This suggests sparse coding problems can be solved by training autoencoders using gradient descent based algorithms. Further, recent investigations have led to the conjecture/belief that many important unsupervised learning tasks, e.g. recognizing handwritten digits, are sparse coding problems in disguise [25, 26]. Thus, our results could shed some light on the observed phenomenon that gradient descent based algorithms train autoencoders to low reconstruction error for natural data sets, like MNIST.

It remains to rigorously show whether a gradient descent algorithm can be initialized randomly (may be far away from A^*) and still be shown to converge to this neighborhood of critical points around the dictionary. Towards that it might be helpful to understand the structure of the Hessian outside this neighborhood. Since our analysis applies to the expected gradient, it remains to analyze the sample complexities where these nice results will become prominent.

The possibility also remains open that this standard loss or some other loss functions exist for the autoencoder with the provable property of having a global minima/minimum at the ground truth dictionary. We have mentioned one example of such in a special case (when A^* is square orthogonal and x^* is nonnegative) and even in this special case it remains open to find a provable optimization algorithm.

On the simulation front we have a couple of open challenges yet to be tackled. Firstly, it is left to find efficient implementations of the iterative update rule based on the exact gradient of the proposed loss function which has been given in (2). This would open up avenues for testing the power of this loss function on real data rather than the synthetic data used here. Secondly, a simulation of the main Theorem 3.2 that can probe deeper into its claim would need to be able to sample A^* for different h at a fixed value of the incoherence parameter ξ. This sampling question of A^* with these constraints is an unresolved one that is left for future work.

Autoencoders with more than one hidden layer have been used for unsupervised feature learning [22] and recently there has been an analysis of the sparse coding performance of convolutional neural networks with one layer [20] and two layers of nonlinearities [39]. The connections between neural networks and sparse coding has also been recently explored in [14]. It remains an exciting open avenue of research to try to do a similar study as in this work to determine if and how deeper architectures under the same generative model might provide better means of doing sparse coding.

Acknowledgements

Akshay Rangamani and Trac Tran are partially supported by the US Air Force under contract FA8651-17-C-0017. Akshay Rangamani and Peter Chin are also supported by the AFOSR grant FA9550-12-1-0136. Peter Chin is also supported by National Science Foundation grant DMS 1737897 and National Institute of Health grant R21 EY028381-01 Amitabh Basu and Anirbit Mukherjee gratefully acknowledges support from the

National Science Foundation grant CMMI1452820 and Office of Naval Research grant N000141812096. We would like to thank Raman Arora (JHU), and Siva Theja Maguluri (Georgia Institute of Technology) for illuminating comments and discussion.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[2] A. Agarwal, A. Anandkumar, P. Jain, P. Netrapalli, and R. Tandon. Learning sparsely used overcomplete dictionaries. In *COLT*, pages 123–137, 2014.

[3] G. Alain and Y. Bengio. What regularized auto-encoders learn from the data-generating distribution. *Journal of Machine Learning Research*, 15(1):3563–3593, 2014.

[4] Z. Allen-Zhu. Natasha 2:faster non-convex optimization than sgd. *arXiv preprint arXiv:1708.08694*, 2017.

[5] A. Anandkumar, R. Ge, D. J. Hsu, S. M. Kakade, and M. Telgarsky. Tensor decompositions for learning latent variable models. *Journal of Machine Learning Research*, 15(1):2773–2832, 2014.

[6] R. Arora, A. Basu, P. Mianjy, and A. Mukherjee. Understanding deep neural networks with rectified linear units. *arXiv preprint arXiv:1611.01491*, 2016.

[7] S. Arora, A. Bhaskara, R. Ge, and T. Ma. More algorithms for provable dictionary learning. *arXiv:1401.0579*, 2014.

[8] S. Arora, R. Ge, T. Ma, and A. Moitra. Simple, efficient, and neural algorithms for sparse coding. In *COLT*, pages 113–149, 2015.

[9] S. Arora, R. Ge, and A. Moitra. New algorithms for learning incoherent and overcomplete dictionaries. In *COLT*, pages 779–806, 2014.

[10] D. Arpit, Y. Zhou, H. Ngo, and V. Govindaraju. Why regularized auto-encoders learn sparse representation? In *International Conference on Machine Learning*, pages 136–144, 2016.

[11] P. Baldi. Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, pages 37–49, 2012.

[12] Y. Bengio, L. Yao, G. Alain, and P. Vincent. Generalized denoising auto-encoders as generative models. In *Advances in Neural Information Processing Systems*, pages 899–907, 2013.

[13] J. Błasiok and J. Nelson. An improved analysis of the er-spud dictionary learning algorithm. *arXiv:1602.05719*, 2016.

[14] A. Bora, A. Jalal, E. Price, and A. G. Dimakis. Compressed sensing using generative models. *arXiv preprint arXiv:1703.03208*, 2017.

[15] A. Coates, A. Ng, and H. Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223, 2011.

[16] A. Coates and A. Y. Ng. The importance of encoding versus training with sparse coding and vector quantization. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 921–928, 2011.

[17] S. S. Du, J. D. Lee, and Y. Tian. When is a convolutional filter easy to learn? *arXiv preprint arXiv:1709.06129*, 2017.

[18] R. Ge, C. Jin, and Y. Zheng. No spurious local minima in nonconvex low rank problems: A unified geometric analysis. *arXiv preprint arXiv:1704.00708*, 2017.

[19] A. Gilbert. Cbms conference on sparse approximation and signal recovery algorithms, may 22-26, 2017 and 16th new mexico analysis seminar, may 21. *https://www.math.nmsu.edu/ jlakey/cbms2017/ cbms_lecture˙notes.html*.

[20] A. C. Gilbert, Y. Zhang, K. Lee, Y. Zhang, and H. Lee. Towards understanding the invertibility of convolutional neural networks. *arXiv preprint arXiv:1705.08664*, 2017.

[21] M. Janzamin, H. Sedghi, and A. Anandkumar. Beating the perils of non-convexity: Guaranteed training of neural networks using tensor methods. *arXiv preprint arXiv:1506.08473*, 2015.

[22] Q. V. Le. Building high-level features using large scale unsupervised learning. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 8595–8598. IEEE, 2013.

[23] J. Li, T. Zhang, W. Luo, J. Yang, X.-T. Yuan, and J. Zhang. Sparseness analysis in the pretraining of deep neural networks. *IEEE transactions on neural networks and learning systems*, 2016.

[24] Y. Li and Y. Yuan. Convergence analysis of two-layer neural networks with relu activation. *arXiv preprint arXiv:1705.09886*, 2017.

[25] A. Makhzani and B. Frey. K-sparse autoencoders. *arXiv preprint arXiv:1312.5663*, 2013.

[26] A. Makhzani and B. J. Frey. Winner-take-all autoencoders. In *Advances in Neural Information Processing Systems*, pages 2791–2799, 2015.

[27] S. Mei, Y. Bai, and A. Montanari. The landscape of empirical risk for non-convex losses. *arXiv preprint arXiv:1607.06534*, 2016.

[28] A. Moitra and G. Valiant. Settling the polynomial learnability of mixtures of gaussians. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 93–102. IEEE, 2010.

[29] A. Ng. Sparse autoencoder. 2011.

[30] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607, 1996.

[31] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

[32] B. A. Olshausen and D. J. Field. How close are we to understanding v1? *Neural computation*, 17(8):1665–1699, 2005.

[33] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 833–840, 2011.

[34] H. Sedghi and A. Anandkumar. Provable methods for training neural networks with sparse connectivity. *arXiv preprint arXiv:1412.2693*, 2014.

[35] D. A. Spielman, H. Wang, and J. Wright. Exact recovery of sparsely-used dictionaries. In *COLT*, pages 37–1, 2012.

[36] Y. Tian. An analytical formula of population gradient for two-layered relu network and its applications in convergence and critical point analysis. *arXiv preprint arXiv:1703.00560*, 2017.

[37] T. Tieleman and G. Hinton. RMSprop Gradient Optimization.

[38] A. M. Tillmann. On the computational intractability of exact and approximate dictionary learning. *IEEE Signal Processing Letters*, 22(1):45–49, 2015.

[39] P. Vardan, Y. Romano, and M. Elad. Convolutional neural networks analyzed via convolutional sparse coding. *arXiv preprint arXiv:1607.08194*, 2016.

[40] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.

[41] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(Dec):3371–3408, 2010.

[42] L. Wu, Z. Zhu, et al. Towards understanding generalization of deep learning: Perspective of loss landscapes. *arXiv preprint arXiv:1706.10239*, 2017.

[43] Q. Zhang, R. Panigrahy, S. Sachdeva, and A. Rahimi. Electron-proton dynamics in deep learning. *arXiv preprint arXiv:1702.00458*, 2017.

Appendix

8 The proxy gradient is a good approximation of the true expectation of the gradient (Proof of Lemma 5.1)

Proof. To make it easy to present this argument let us abstractly think of the function f (defined for any $i \in \{1, 2, 3, .., h\}$) as $f(y, W, X) = \frac{\partial L}{\partial W_i}$ where we have defined the random variable $X = \text{Th}[W_i^T y - \epsilon_i]$. It is to be noted that because of the ReLU term and its derivative this function f has a dependency on $y = A^* x^*$ even outside its dependency through X. Let us define another random variable $Y = \mathbf{1}_{i \in \text{Support}(x^*)}$. Then we have,

$$\begin{aligned}
&\|\mathbb{E}_{x^*}[f(y, W, X)] - \mathbb{E}_{x^*}[f(y, W, Y)]\|_{\ell_2} \\
\leq &\mathbb{E}_{x^*}[|f(y, W, X) - f(y, W, Y)|_{\ell_2}] \\
\leq &\mathbb{E}_{x^*}[|f(y, W, X)(\mathbf{1}_{X=Y} + \mathbf{1}_{X\neq Y}) - f(y, W, Y)(\mathbf{1}_{X=Y} + \mathbf{1}_{X\neq Y})|_{\ell_2}] \\
\leq &\mathbb{E}_{x^*}[|(f(y, W, X) - f(y, W, Y))|_{\ell_2} \mathbf{1}_{X\neq Y}] \\
\leq &\sqrt{\mathbb{E}_{x^*}[|f(y, W, X) - f(y, W, Y)|_2^2]} \sqrt{\mathbb{E}_{x^*}[\mathbf{1}_{X\neq Y}]}
\end{aligned}$$

In the last step above we have used the Cauchy-Schwarz inequality for random variables. We recognize that $\mathbb{E}_{x^*}[f(y, W, Y)]$ is precisely what we defined as the proxy gradient $\widehat{\nabla_i L}$. Further for such W as in this lemma the support recovery theorem (Theorem 3.1) holds and that is precisely the statement that the term, $\mathbb{E}_{x^*}[\mathbf{1}_{X\neq Y}]$ is small. So we can rewrite the above inequality as,

$$\left\| \mathbb{E}_{x^*}[\frac{\partial L}{\partial W_i}] - \widehat{\nabla_i L} \right\|_2 \leq \sqrt{\mathbb{E}_{x^*}[|f(y, W, X) - f(y, W, Y)|_2^2]} \exp\left(-\frac{h^p m_1^2}{2(b-a)^2} \right)$$

We remember that f is a polynomial in h because its h dependency is through Frobenius norms of submatrices of W and ℓ_2 norms of projections of Wy. But the ℓ_∞ norm of the training vectors y (that is b) have been assumed to be bounded by poly(h). Also we have the assumption that the columns of $W^\top$ are within a $\frac{1}{h^{p+\nu 2}}-$ball of the corresponding columns of A^* which in turn is a $n \times h$ dimensional matrix of bounded norm because all its columns are normalized. So summarizing we have,

$$\left\| \mathbb{E}_{x^*}[\frac{\partial L}{\partial W_i}] - \widehat{\nabla_i L} \right\|_2 \leq \text{poly}(h) \exp\left(-\frac{h^p m_1^2}{2(b-a)^2} \right)$$

The above inequality immediately implies the claimed lemma. □

9 The asymptotics of the coefficients of the gradient of the squared loss (Proof of Lemma 5.2)

We will pick up from where subsection 5.1 left and will now estimate bounds on each of the terms $\alpha_i, \beta_i, ||e_i||$, which were defined at the end of that segment. We will separate them as $\alpha_i = \tilde{\alpha_i} + \hat{\alpha_i}$ (similarly for the other terms). Where the tilde terms are those that come as a coefficient of m_2, and the hat terms are the ones that come as coefficient of m_1 or ϵ or both. (Note : Given the previous definitions of q_1 and q_2 it is obvious from context as to how the quantities q_i, q_{ij}, q_{ijk} and q_S mean and we shall use this notation in this Appendix.)

9.1 Estimating the m_2 dependent parts of the derivative

Since $||A_i^*||= 1$ and W_i is being assumed to be within a $0 < \delta < 1$ ball of A_i^* we can use the following inequalities:

$$
\begin{aligned}
||W_i|| &= ||W_i - A_i^* + A_i^*|| \leq ||W_i - A_i^*|| + ||A_i^*|| = \delta + 1 \\
||W_i|| &\geq 1 - \delta \\
\langle W_i, A_i^* \rangle &= \langle W_i - A_i^*, A_i^* \rangle + \langle A_i^*, A_i^* \rangle \leq ||W_i - A_i^*||||A_i^*|| + 1 \leq \delta + 1 \\
\langle W_i, A_i^* \rangle &\geq 1 - \delta \\
|\langle W_j, A_i^* \rangle| &= |\langle W_j - A_j^*, A_i^* \rangle + \langle A_j^*, A_i^* \rangle| \leq \frac{\mu}{\sqrt{n}} + ||W_j - A_j^*||||A_i^*|| = \frac{\mu}{\sqrt{n}} + \delta \\
|\langle W_i, W_j \rangle| &= |\langle W_i - A_i^*, W_j \rangle + \langle A_i^*, W_j \rangle| \leq \delta(1+\delta) + (\delta + \frac{\mu}{\sqrt{n}}) = \delta^2 + 2\delta + \frac{\mu}{\sqrt{n}} \\
\langle W_i, W_i \rangle &= ||W_i||^2 \geq (1-\delta)^2 \\
\langle W_i, W_i \rangle &= ||W_i||^2 \leq (1+\delta)^2
\end{aligned}
$$

Bounding $\tilde{\beta}_i$

$$
\begin{aligned}
\tilde{\beta}_i &= \mathbb{E}_{S \in \mathbb{S}} \left[\mathbf{1}_{i \in S} \left\{ 2Dm_2 (W_i^\top A_i^*) - m_2 \sum_{j \in S} (W_i^\top W_j)(W_j^\top A_i^*) \right\} \right] \\
&= \mathbb{E}_{S \in \mathbb{S}} \left[\mathbf{1}_{i \in S} \left\{ 2Dm_2 \langle W_i, A_i^* \rangle - m_2 ||W_i||^2 \langle W_i, A_i^* \rangle - m_2 \sum_{\substack{j \in S \\ j \neq i}} \langle W_i, W_j \rangle \langle W_j, A_i^* \rangle \right\} \right]
\end{aligned}
$$

Evaluating the outer expectation we get,

$$
\begin{aligned}
\tilde{\beta}_i &= \sum_{\{S \in \mathbb{S}: i \in S\}} q_S 2Dm_2 \langle W_i, A_i^* \rangle - \sum_{\{S \in \mathbb{S}: i \in S\}} q_S m_2 ||W_i||^2 \langle W_i, A_i^* \rangle - m_2 \sum_{\substack{j=1 \\ j \neq i}}^{h} \langle W_i, W_j \rangle \langle W_j, A_i^* \rangle \sum_{\{S \in \mathbb{S}: i,j \in S, i \neq j\}} q_S \\
&= 2Dq_i m_2 \langle W_i, A_i^* \rangle - q_i m_2 ||W_i||^2 \langle W_i, A_i^* \rangle - m_2 \sum_{\substack{j=1 \\ j \neq i}}^{h} q_{ij} \langle W_i, W_j \rangle \langle W_j, A_i^* \rangle
\end{aligned}
$$

Upper bounding the above we get,

$$
\begin{aligned}
\tilde{\beta}_i &\leq 2Dm_2 h^{p-1}(1+\delta) - m_2 h^{p-1}(1-\delta)^3 + m_2 h^{2p-1} \left(\delta + \frac{\mu}{\sqrt{n}} \right) \left(\delta^2 + 2\delta + \frac{\mu}{\sqrt{n}} \right) \\
&= 2Dm_2 h^{p-1}(1 + h^{-p-\nu^2}) - m_2 h^{p-1}(1 - 3h^{-p-\nu^2} + 3h^{-2p-2\nu^2} - h^{-3p-3\nu^2}) \\
&+ m_2 h^{2p-1}(h^{-3p-3\nu^2} + 2h^{-2p-2\nu^2} + h^{-2p-2\nu^2-\xi} + 3h^{-p-\nu^2-\xi} + h^{-2\xi}) \qquad (5)
\end{aligned}
$$

Similarly for the lower bound on β_i we get,

$$
\begin{aligned}
\tilde{\beta}_i &\geq 2Dm_2 h^{p-1}(1-\delta) - m_2 h^{p-1}(1+\delta)^3 - m_2 h^{2p-1} \left(\delta + \frac{\mu}{\sqrt{n}} \right) \left(\delta^2 + 2\delta + \frac{\mu}{\sqrt{n}} \right) \\
&= 2Dm_2 h^{p-1}(1 - h^{-p-\nu^2}) - m_2 h^{p-1}(1 + 3h^{-p-\nu^2} + 3h^{-2p-2\nu^2} + h^{-3p-3\nu^2}) \\
&- m_2 h^{2p-1}(h^{-3p-3\nu^2} + 2h^{-2p-2\nu^2} + h^{-2p-2\nu^2-\xi} + 3h^{-p-\nu^2-\xi} + h^{-2\xi}) \qquad (6)
\end{aligned}
$$

Thus for $0 < p < 2\xi$ and $D = 1$, we have $\beta = \Theta\left(m_2 h^{p-1}\right)$

Bounding $\tilde{\alpha_i}$

$$\begin{aligned}
\tilde{\alpha_i} &= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\left\{m_2\sum_{k\in S}(W_i^\top A_k^*)^2\right\}\right] \\
&= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\left\{m_2\langle W_i, A_i^*\rangle^2 + m_2\sum_{\substack{k\in S\\k\neq i}}\langle W_i, A_k^*\rangle^2\right\}\right] \\
&= \sum_{\{S\in\mathbb{S}:i\in S\}} m_2\langle W_i, A_i^*\rangle^2 q_S + \sum_{\substack{k=1\\k\neq i}}^{h}\sum_{\{S\in\mathbb{S}:i,k\in S\}}\langle W_i, A_k^*\rangle^2 q_S \\
&= m_2\langle W_i, A_i^*\rangle^2\sum_{\{S\in\mathbb{S}:i\in S\}} q_S + m_2\sum_{\substack{k=1\\k\neq i}}^{h}\langle W_i, A_k^*\rangle^2\left(\sum_{\{S\in\mathbb{S}:i,k\in S,i\neq k\}} q_S\right) \\
&= q_i m_2\langle W_i, A_i^*\rangle^2 + m_2\sum_{\substack{k=1\\k\neq i}}^{h} q_{ik}\langle W_i, A_k^*\rangle^2 \\
&= h^{p-1}m_2\langle W_i, A_i^*\rangle^2 + m_2 h^{2p-1}\max\langle W_i, A_k^*\rangle^2
\end{aligned}$$

The above implies the following bounds,

$$h^{p-1}m_2(1-h^{-p-\nu^2})^2 \leq \tilde{\alpha_i} \leq h^{p-1}m_2(1+h^{-p-\nu^2})^2 + m_2h^{2p-1}(h^{-p-\nu^2}+h^{-\xi})^2 \tag{7}$$

As long as $0 < p < 2\xi$, $\tilde{\alpha_i} = \Theta\left(m_2h^{p-1}\right)$

Bounding $||\tilde{e_i}||_2$

$$\begin{aligned}
\tilde{e_i} = \mathbb{E}_{S\in\mathbb{S}}&\left[\mathbf{1}_{i\in S}\times\left\{m_2\sum_{\substack{j,k\in S\\j\neq i}}(W_i^\top A_k^*)(W_j^\top A_k^*)W_j + (-2D)m_2\sum_{\substack{j\in S\\j\neq i}}(W_i^\top A_j^*)A_j^*\right\}\right] \\
+\mathbb{E}_{S\in\mathbb{S}}&\left[\mathbf{1}_{i\in S}\times\left\{m_2\sum_{\substack{j,k\in S\\k\neq i}}(W_i^\top W_j)(W_j^\top A_k^*)A_k^*\right\}\right]
\end{aligned}$$

Expanding further over the summation of the j and the k indices we have,

$$\tilde{e_i} = \mathbb{E}_{S\in\mathbb{S}}\Bigg[\mathbf{1}_{i\in S}\times m_2\Bigg\{\sum_{j(=k)\in S\setminus i}(W_i^\top A_j^*)(W_j^\top A_j^*)W_j + \sum_{\substack{j\in S\setminus i\\ k\in S\setminus i,j}}(W_i^\top A_k^*)(W_j^\top A_k^*)W_j$$
$$+\sum_{\substack{j\in S\setminus i\\ k=i}}(W_i^\top A_i^*)(W_j^\top A_i^*)W_j\Bigg\}\Bigg]$$
$$+\mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times(-2D)m_2\left\{\sum_{\substack{j\in S\\ j\neq i}}(W_i^\top A_j^*)A_j^*\right\}\right]$$
$$+\mathbb{E}_{S\in\mathbb{S}}\Bigg[\mathbf{1}_{i\in S}\times m_2\Bigg\{\sum_{k(=j)\in S\setminus i}(W_i^\top W_k)(W_k^\top A_k^*)A_k^* + \sum_{\substack{k\in S\setminus i\\ j\in S\setminus i,k}}(W_i^\top W_j)(W_j^\top A_k^*)A_k^*$$
$$+\sum_{\substack{k\in S\setminus i\\ j=i}}(W_i^\top W_i)(W_i^\top A_k^*)A_k^*\Bigg\}\Bigg]$$

Expanding the above in terms of q_S we have,

$$\tilde{e_i} = m_2\Bigg\{\sum_{j=1,j\neq i}^{h}(W_i^\top A_j^*)(W_j^\top A_j^*)W_j\sum_{\{S\in\mathbb{S}:i,j\in S,i\neq j\}}q_S + \sum_{\substack{j,k=1\\ j\neq k\neq i}}^{h}(W_i^\top A_k^*)(W_j^\top A_k^*)W_j\sum_{\{S\in\mathbb{S}:i,j,k\in S,i\neq j\neq k\}}q_S$$
$$+\sum_{\substack{j=1\\ j\neq i}}^{h}(W_i^\top A_i^*)(W_j^\top A_i^*)W_j\sum_{\{S\in\mathbb{S}:i,j\in S,i\neq j\}}q_S\Bigg\}$$
$$+(-2D)m_2\left\{\sum_{\substack{j=1\\ j\neq i}}^{h}(W_i^\top A_j^*)A_j^*\sum_{\{S\in\mathbb{S}:i,j\in S,i\neq j\}}q_S\right\}$$
$$+m_2\Bigg\{\sum_{\substack{k=1\\ k\neq i}}^{h}(W_i^\top W_k)(W_k^\top A_k^*)A_k^*\sum_{\{S\in\mathbb{S}:i,k\in S,i\neq k\}}q_S + \sum_{\substack{j,k=1\\ j\neq i\neq k}}^{h}(W_i^\top W_j)(W_j^\top A_k^*)A_k^*\sum_{\{S\in\mathbb{S}:i,j,k\in S,i\neq j\neq k\}}q_S$$
$$+\sum_{\substack{k=1\\ k\neq i}}^{h}(W_i^\top W_i)(W_i^\top A_k^*)A_k^*\sum_{\{S\in\mathbb{S}:i,k\in S,i\neq k\}}q_S\Bigg\}$$

Expanding the q_S dependency in terms of q_{ij} and q_{ijk} we have,

$$\tilde{e_i} = m_2 \Bigg\{ \sum_{j=1, j\neq i}^{h} q_{ij} (W_i^\top A_j^*)(W_j^\top A_j^*) W_j + \sum_{\substack{j,k=1 \\ j\neq k\neq i}}^{h} q_{ijk} (W_i^\top A_k^*)(W_j^\top A_k^*) W_j$$
$$+ \sum_{\substack{j=1 \\ j\neq i}}^{h} q_{ij} (W_i^\top A_i^*)(W_j^\top A_i^*) W_j \Bigg\} + (-2D) m_2 \left\{ \sum_{\substack{j=1 \\ j\neq i}}^{h} q_{ij} (W_i^\top A_j^*) A_j^* \right\}$$
$$+ m_2 \Bigg\{ \sum_{\substack{k=1 \\ k\neq i}}^{h} q_{ik} (W_i^\top W_k)(W_k^\top A_k^*) A_k^* + \sum_{\substack{j,k=1 \\ j\neq i\neq k}}^{h} q_{ijk} (W_i^\top W_j)(W_j^\top A_k^*) A_k^*$$
$$+ \sum_{\substack{k=1 \\ k\neq i}}^{h} q_{ik} (W_i^\top W_i)(W_i^\top A_k^*) A_k^* \Bigg\}$$

Upper bounding the norm of this vector $\tilde{e_i}$ we get,

$$\begin{aligned}
||\tilde{e_i}|| &\leq m_2 h^{2p-1} \left(\delta + \frac{\mu}{\sqrt{n}}\right)(1+\delta)^2 + m_2 h^{3p-1} \left(\delta + \frac{\mu}{\sqrt{n}}\right)^2 (1+\delta) \\
&+ m_2 h^{2p-1} \left(\delta + \frac{\mu}{\sqrt{n}}\right)(1+\delta)^2 + 2D m_2 h^{2p-1} \left(\delta + \frac{\mu}{\sqrt{n}}\right) \\
&+ m_2 h^{2p-1} \left(\delta^2 + 2\delta + \frac{\mu}{\sqrt{n}}\right)(1+\delta) + m_2 h^{3p-1} \left(\delta^2 + 2\delta + \frac{\mu}{\sqrt{n}}\right) \left(\delta + \frac{\mu}{\sqrt{n}}\right) \\
&+ m_2 h^{2p-1} \left(\delta + \frac{\mu}{\sqrt{n}}\right)(1+\delta)^2 \\
&\leq m_2 h^{2p-1} (h^{-p-\nu^2} + 2h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + 2h^{-p-\nu^2-\xi} + h^{-2p-2\nu^2-\xi} + h^{-\xi}) \\
&+ m_2 h^{3p-1} (h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + 2h^{-p-\nu^2-\xi} + 2h^{-2p-2\nu^2-\xi} + h^{-2\xi} + h^{-p-\nu^2-2\xi}) \\
&+ m_2 h^{2p-1} (h^{-p-\nu^2} + 2h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + 2h^{-p-\nu^2-\xi} + h^{-2p-2\nu^2-\xi} + h^{-\xi}) \\
&+ 2D m_2 h^{2p-1} (h^{-p-\nu^2} + h^{-\xi}) \\
&+ m_2 h^{2p-1} (2h^{-p-\nu^2} + 3h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + h^{-p-\nu^2-\xi} + h^{-\xi}) \\
&+ m_2 h^{3p-1} (2h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + 3h^{-p-\nu^2-\xi} + h^{-2p-2\nu^2-\xi} + h^{-2\xi}) \\
&+ m_2 h^{2p-1} (h^{-p-\nu^2} + 2h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + 2h^{-p-\nu^2-\xi} + h^{-2p-2\nu^2-\xi} + h^{-\xi}) \qquad (8)
\end{aligned}$$

If $D = 1$ and $0 < p < \xi$, we get $||\tilde{e_i}|| = o(m_2 h^{p-1})$

9.2 Estimating the m_1 dependent parts of the derivative

We continue working in the same regime for the W matrix as in the previous subsection. Hence the same inequalities as listed at the beginning of the previous subsection continue to hold and we use them to get the following bounds,

Bounding $\hat{\alpha_i}$

$$\begin{aligned}
\hat{\alpha_i} &= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{m_1^2\sum_{\substack{k,l\in S\\k\neq l}}(W_i^\top A_k^*)(W_i^\top A_l^*) - 2m_1\sum_{k\in S}\epsilon_i(W_i^\top A_k^*)+\epsilon_i^2\right\}\right]\\
&= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{m_1^2\sum_{\substack{k\in S\\k\neq i}}\langle W_i,A_k^*\rangle\langle W_i,A_i^*\rangle + m_1^2\sum_{\substack{l\in S\\l\neq i}}\langle W_i,A_i^*\rangle\langle W_i,A_l^*\rangle + m_1^2\sum_{\substack{k,l\in S\\k\neq l\\k\neq i\\l\neq i}}\langle W_i,A_k^*\rangle\langle W_i,A_l^*\rangle\right.\right.\\
&\quad\left.\left. - 2m_1\epsilon_i\langle W_i,A_i^*\rangle - 2m_1\sum_{\substack{k\in S\\k\neq i}}\epsilon_i\langle W_i,A_k^*\rangle+\epsilon_i^2\right\}\right]\\
&= 2m_1^2\sum_{\substack{k=1\\k\neq i}}^{h}\langle W_i,A_k^*\rangle\langle W_i,A_i^*\rangle\sum_{\{S\in\mathbb{S}:i,k\in S,k\neq i\}}q_S + m_1^2\sum_{\substack{k,l=1\\k\neq l\\k\neq i\\l\neq i}}^{h}\langle W_i,A_k^*\rangle\langle W_i,A_l^*\rangle\sum_{\{S\in\mathbb{S}:i,k,l\in S,k\neq i\neq l\}}q_S\\
&\quad - 2m_1\epsilon_i\langle W_i,A_i^*\rangle\sum_{\{S\in\mathbb{S}:i\in S\}}q_S - 2m_1\sum_{\substack{k=1\\k\neq i}}^{h}\epsilon_i\langle W_i,A_k^*\rangle\sum_{\{S\in\mathbb{S}:i,k\in S,k\neq i\}}q_S+\epsilon_i^2\sum_{\{S\in\mathbb{S}:i\in S\}}q_S\\
\implies \hat{\alpha_i} &= 2m_1^2\sum_{\substack{k=1\\k\neq i}}^{h}q_{ik}\langle W_i,A_k^*\rangle\langle W_i,A_i^*\rangle + m_1^2\sum_{\substack{k,l=1\\k\neq l\\k\neq i\\l\neq i}}^{h}q_{ikl}\langle W_i,A_k^*\rangle\langle W_i,A_l^*\rangle\\
&\quad - 2m_1q_i\epsilon_i\langle W_i,A_i^*\rangle - 2m_1\sum_{\substack{k=1\\k\neq i}}^{h}q_{ik}\epsilon_i\langle W_i,A_k^*\rangle+q_i\epsilon_i^2
\end{aligned}$$

We plugin $\epsilon_i = 2m_1h^p\left(\delta+\frac{\mu}{\sqrt{n}}\right)$ for $i=1,\ldots,h$

$$\begin{aligned}
|\hat{\alpha_i}| &\leq 2m_1^2h^{2p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta) + m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2 + 4m_1^2h^{2p-1}(1+\delta)\left(\delta+\frac{\mu}{\sqrt{n}}\right)\\
&\quad + 4m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2 + 4m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2\\
&= 2m_1^2h^{2p-1}(h^{-p-\nu^2}+h^{-2p-2\nu^2}+h^{-p-\nu^2-\xi}+h^{-\xi}) + m_1^2h^{3p-1}(h^{-2p-2\nu^2}+2h^{-p-\nu^2-\xi}+h^{-2\xi})\\
&\quad + 4m_1^2h^{2p-1}(h^{-p-\nu^2}+h^{-2p-2\nu^2}+h^{-\xi}+h^{-p-\nu^2-\xi}) + 4m_1^2h^{3p-1}(h^{-2p-2\nu^2}+2h^{-p-\nu^2-\xi}+h^{-2\xi})\\
&\quad + 4m_1^2h^{3p-1}(h^{-2p-2\nu^2}+2h^{-p-\nu^2-\xi}+h^{-2\xi})
\end{aligned}$$

This means that if $p<\xi$, $|\hat{\alpha_i}|= o(m_1^2h^{p-1})$. Putting this together with the bounds obtained below equation 7, we get that $\alpha_i = \Theta(m_2h^{p-1}) + o(m_1^2h^{p-1})$.

Bounding $\hat{\beta}_i$

$$\hat{\beta}_i = \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{2Dm_1^2\sum_{\substack{k\in S\\k\neq i}}(W_i^\top A_k^*) - Dm_1\epsilon_i + m_1\sum_{j\in S}\epsilon_j(W_i^\top W_j) - m_1^2\sum_{\substack{j,l\in S\\l\neq i}}(W_i^\top W_j)(W_j^\top A_l^*)\right\}\right]$$

$$= 2Dm_1^2\sum_{\substack{k=1\\k\neq i}}^{h}\langle W_i, A_k^*\rangle\sum_{\{S\in\mathbb{S}:i,k\in S,k\neq i\}}q_S - Dm_1\epsilon_i\sum_{\{S\in\mathbb{S}:i\in S\}}q_S + m_1\epsilon_i||W_i||^2\sum_{\{S\in\mathbb{S}:i\in S\}}q_S$$

$$+ m_1\sum_{j=1,j\neq i}^{h}\epsilon_j\langle W_i, W_j\rangle\sum_{\{S\in\mathbb{S}:i,j\in S,j\neq i\}}q_S - m_1^2\sum_{\substack{l=1\\l\neq i}}^{h}||W_i||^2\langle W_i, A_l^*\rangle\sum_{\{S\in\mathbb{S}:i,l\in S,l\neq i\}}q_S$$

$$- m_1^2\sum_{\substack{l=1\\l\neq i}}^{h}\langle W_i, W_l\rangle\langle W_l, A_l^*\rangle\sum_{\{S\in\mathbb{S}:i,l\in S,l\neq i\}}q_S - m_1^2\sum_{\substack{j,l=1\\l\neq i\\j\neq l,i}}^{h}\langle W_i, W_j\rangle\langle W_j, A_l^*\rangle\sum_{\{S\in\mathbb{S}:i,j,l\in S,l\neq i\neq i\}}q_S$$

$$= 2Dm_1^2\sum_{\substack{k=1\\k\neq i}}^{h}q_{ik}\langle W_i, A_k^*\rangle - Dm_1\epsilon_i q_i + m_1\epsilon_i||W_i||^2 q_i + m_1\sum_{j=1,j\neq i}^{h}\epsilon_j q_{ij}\langle W_i, W_j\rangle$$

$$- m_1^2\sum_{\substack{l=1\\l\neq i}}^{h}||W_i||^2\langle W_i, A_l^*\rangle q_{il} - m_1^2\sum_{\substack{l=1\\l\neq i}}^{h}\langle W_i, W_l\rangle\langle W_l, A_l^*\rangle q_{il} - m_1^2\sum_{\substack{j,l=1\\l\neq i\\j\neq l,i}}^{h}\langle W_i, W_j\rangle\langle W_j, A_l^*\rangle q_{ijl}$$

We plugin $\epsilon_i = 2m_1h^p\left(\delta + \frac{\mu}{\sqrt{n}}\right)$ for $i = 1,\ldots,h$

$$|\hat{\beta}_i| \leq 4Dm_1^2h^{2p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right) + 2m_1^2h^{2p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)^2 + 2m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)$$
$$+ m_1^2h^{2p-1}(1+\delta)^2\left(\delta+\frac{\mu}{\sqrt{n}}\right) + m_1^2h^{2p-1}\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)$$
$$+ m_1^2h^{3p-1}\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta+\frac{\mu}{\sqrt{n}}\right)$$
$$= 4Dm_1^2h^{2p-1}(h^{-p-\nu^2}+h^{-\xi})$$
$$+ 2m_1^2h^{2p-1}(h^{-p-\nu^2}+2h^{-2p-2\nu^2}+h^{-3p-3\nu^2}+h^{-\xi}+2h^{-p-\nu^2-\xi}+h^{-2p-2\nu^2-\xi})$$
$$+ 2m_1^2h^{3p-1}(2h^{-2p-2\nu^2}+h^{-3p-3\nu^2}+3h^{-p-\nu^2-\xi}+h^{-2p-2\nu^2-\xi}+h^{-2\xi})$$
$$+ m_1^2h^{2p-1}(h^{-p-\nu^2}+2h^{-2p-2\nu^2}+h^{-3p-3\nu^2}+h^{-\xi}+2h^{-p-\nu^2-\xi}+h^{-2p-2\nu^2-\xi})$$
$$+ m_1^2h^{2p-1}(3h^{-2p-2\nu^2}+h^{-3p-3\nu^2}+h^{-p-\nu^2-\xi}+2h^{-p-\nu^2}+h^{-\xi})$$
$$+ m_1^2h^{3p-1}(2h^{-2p-2\nu^2}+h^{-3p-3\nu^2}+3h^{-p-\nu^2-\xi}+h^{-2p-2\nu^2-\xi}+h^{-2\xi})$$

This means that if $p < \xi$, $|\hat{\beta}_i| = o(m_1^2h^{p-1})$. Putting this together with the bounds obtained below 5, we get that $\beta_i = \Theta(m_2h^{p-1}) + o(m_1^2h^{p-1})$.

Bounding $||\hat{e_i}||_2$

$$\hat{e_i} = \underbrace{\mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{\sum_{\substack{j\in S\\ j\neq i}}\epsilon_i\epsilon_j W_j - m_1\sum_{\substack{j,k\in S\\ j\neq i}}(W_j^\top A_k^*)W_j\epsilon_i - m_1\sum_{\substack{j,k\in S\\ j\neq i}}\epsilon_j(W_i^\top A_k^*)W_j\right\}\right]}_{\hat{e_{i1}}}$$

$$+\underbrace{\mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{m_1^2\sum_{\substack{j,k,l\in S\\ j\neq i\\ k\neq l}}(W_i^\top A_k^*)(W_j^\top A_l^*)W_j\right\}\right]}_{\hat{e_{i2}}}$$

$$+\underbrace{\mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{-2Dm_1^2\sum_{\substack{j,k\in S\\ j\neq i\\ k\neq i}}(W_i^\top A_k^*)A_j^* + Dm_1\sum_{\substack{j\in S\\ j\neq i}}\epsilon_i A_j^*\right\}\right]}_{\hat{e_{i3}}}$$

$$+\underbrace{\mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{-m_1\sum_{\substack{j,k\in S\\ k\neq i}}\epsilon_j(W_i^\top W_j)A_k^* + m_1^2\sum_{\substack{j,k,l\in S\\ k\neq i,l}}(W_i^\top W_j)(W_j^\top A_l^*)A_k^*\right\}\right]}_{\hat{e_{i4}}}$$

We estimate the different summands separately.

$$\hat{e_{i1}} = \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{\sum_{\substack{j\in S\\ j\neq i}}\epsilon_i\epsilon_j W_j\right\}\right]$$

$$+\mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times(-m_1)\left\{\sum_{j(=k)\in S\setminus i}(W_j^\top A_j^*)W_j\epsilon_i + \sum_{\substack{j\in S\setminus i\\ k\in S\setminus i,j}}(W_j^\top A_k^*)W_j\epsilon_i + \sum_{\substack{j\in S\setminus i\\ k=i}}(W_j^\top A_i^*)W_j\epsilon_i\right\}\right]$$

$$+\mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times(-m_1)\left\{\sum_{j(=k)\in S\setminus i}\epsilon_j(W_i^\top A_j^*)W_j + \sum_{\substack{j\in S\setminus i\\ k\in S\setminus i,j}}\epsilon_j(W_i^\top A_k^*)W_j + \sum_{\substack{j\in S\setminus i\\ k=i}}\epsilon_j(W_i^\top A_i^*)W_j\right\}\right]$$

We substitute, $\epsilon = 2m_1h^p(h^{-p-\nu^2} + h^{-\xi})$ and for any two vectors $\mathbf{x}$ and $\mathbf{y}$ and any two scalars a and b we use the inequality, $||a\mathbf{x} + b\mathbf{y}||_2 \leq |a|_{max}||\mathbf{x}||_{2,max} + |b|_{max}||\mathbf{y}||_{2,max}$ to get,

$$
\begin{aligned}
||\hat{e_{i1}}||_2 &\leq 4m_1^2h^{2p}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2\sum_{j=1,j\neq i}^{h}q_{ij}||W_j|| \\
&\quad+2m_1^2h^p\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\sum_{j=1,j\neq i}^{h}q_{ij}\langle W_j,A_j^*\rangle W_j+\sum_{j,k=1,j\neq i,k\neq i,j}q_{ijk}\langle W_j,A_k^*\rangle W_j\right. \\
&\quad\left.+\sum_{j=1,j\neq i}^{h}q_{ij}\langle W_j,A_i^*\rangle W_j\right) \\
&\quad+2m_1^2h^p\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\sum_{j=1,j\neq i}^{h}q_{ij}\langle W_i,A_j^*\rangle W_j+\sum_{j,k=1,j\neq i,k\neq i,j}q_{ijk}\langle W_i,A_k^*\rangle W_j\right. \\
&\quad\left.+\sum_{j=1,j\neq i}^{h}q_{ij}\langle W_i,A_i^*\rangle W_j\right) \\
\implies ||\hat{e_{i1}}||_2 &\leq 4m_1^2h^{2p}h^{2p-1}(1+\delta)\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2 \\
&\quad+2m_1^2h^p\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(h^{2p-1}(1+\delta)^2+h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)+h^{2p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)\right) \\
&\quad+2m_1^2h^p\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(h^{2p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)+h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)+h^{2p-1}(1+\delta)^2\right) \\
\implies ||\hat{e_{i1}}||_2 &\leq 4m_1^2h^{4p-1}(1+\delta)\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2 \\
&\quad+2m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)^2+2m_1^2h^{4p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(1+\delta) \\
&\quad+2m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(1+\delta) \\
&\quad+2m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(1+\delta)+2m_1^2h^{4p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(1+\delta) \\
&\quad+2m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)^2 \\
\implies ||\hat{e_{i1}}||_2 &\leq 8m_1^2h^{4p-1}(1+\delta)\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2+4m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)^2 \\
&\quad+4m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(1+\delta)
\end{aligned}
$$

$$
\begin{aligned}
\implies ||\hat{e_{i1}}||_2 &\leq 8m_1^2h^{4p-1}(h^{-2p-2\nu^2}+h^{-3p-3\nu^2}+2h^{-p-\nu^2-\xi}+2h^{-2p-2\nu^2-\xi}+h^{-p-\nu^2-2\xi}+h^{-2\xi}) \\
&+ 4m_1^2h^{3p-1}(h^{-p-\nu^2}+h^{-3p-3\nu^2}+2h^{-2p-2\nu^2}+h^{-\xi}+h^{-2p-2\nu^2-\xi}+2h^{-p-\nu^2-\xi}) \\
&+ 4m_1^2h^{3p-1}(h^{-2p-2\nu^2}+h^{-3p-3\nu^2}+2h^{-p-\nu^2-\xi}+2h^{-2p-2\nu^2-\xi}+h^{-p-\nu^2-2\xi}+h^{-2\xi}) \\
&= 8m_1^2h^{p-1}(h^{p-2\nu^2}+h^{-3\nu^2}+2h^{p-\nu^2+p-\xi}+2h^{-2\nu^2+p-\xi}+h^{-\nu^2+2p-2\xi}+h^{3p-2\xi}) \\
&+ 4m_1^2h^{p-1}(h^{p-\nu^2}+h^{-p-3\nu^2}+2h^{-2\nu^2}+h^{2p-\xi}+h^{-2\nu^2-\xi}+2h^{-\nu^2+p-\xi}) \\
&+ 4m_1^2h^{p-1}(h^{-2\nu^2}+h^{-p-3\nu^2}+2h^{-\nu^2+p-\xi}+2h^{-2\nu^2-\xi}+h^{-\nu^2+p-2\xi}+h^{2p-2\xi})
\end{aligned}
$$

From the above it follows that, $||\hat{e_{i1}}||_2 = o(m_1^2h^{p-1})$ for $p < \nu^2$ and $2p < \xi$
And now we start to estimate $\hat{e_{i2}}$

$$
\begin{aligned}
\hat{e_{i2}} &= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times m_1^2\left\{\sum_{\substack{j,k,l\in S\\ j\neq i\\ k\neq l}}(W_i^\top A_k^*)(W_j^\top A_l^*)W_j\right\}\right] \\
&= \mathbb{E}_{S\in\mathbb{S}}\Bigg[\mathbf{1}_{i\in S}\times m_1^2\Bigg\{\sum_{\substack{j\in S\\ j\neq i}}(W_i^\top A_j^*)(W_j^\top A_i^*)W_j + \sum_{\substack{j,k\in S\\ k\neq j\neq i}}(W_i^\top A_k^*)(W_j^\top A_i^*)W_j \\
&+ \sum_{\substack{j\in S\\ j\neq i}}(W_i^\top A_i^*)(W_j^\top A_j^*)W_j \\
&+ \sum_{\substack{j,l\in S\\ l\neq j\neq i}}(W_i^\top A_i^*)(W_j^\top A_l^*)W_j + \sum_{\substack{j,l\in S\\ l\neq j\neq i}}(W_i^\top A_j^*)(W_j^\top A_l^*)W_j + \sum_{\substack{j,k\in S\\ k\neq j\neq i}}(W_i^\top A_k^*)(W_j^\top A_j^*)W_j \\
&+ \sum_{\substack{j,k,l\in S\\ l\neq k\neq j\neq i}}(W_i^\top A_k^*)(W_j^\top A_l^*)W_j\Bigg\}\Bigg] \\
\implies \hat{e_{i2}} &= m_1^2\Bigg\{\sum_{\substack{j=1\\ j\neq i}}^{h} q_{ij}(W_i^\top A_j^*)(W_j^\top A_i^*)W_j + \sum_{\substack{j,k=1\\ k\neq j\neq i}}^{h} q_{ijk}(W_i^\top A_k^*)(W_j^\top A_i^*)W_j \\
&+ \underbrace{\sum_{\substack{j=1\\ j\neq i}}^{h} q_{ij}(W_i^\top A_i^*)(W_j^\top A_j^*)W_j}_{\mathbf{a}} \\
&+ \sum_{\substack{j,l=1\\ l\neq j\neq i}}^{h} q_{ijl}(W_i^\top A_i^*)(W_j^\top A_l^*)W_j + \sum_{\substack{j,l=1\\ l\neq j\neq i}}^{h} q_{ijl}(W_i^\top A_j^*)(W_j^\top A_l^*)W_j + \sum_{\substack{j,k=1\\ k\neq j\neq i}}^{h} q_{ijk}(W_i^\top A_k^*)(W_j^\top A_j^*)W_j \\
&+ \sum_{\substack{j,k,l\in S\\ l\neq k\neq j\neq i}} q_{ijkl}(W_i^\top A_k^*)(W_j^\top A_l^*)W_j\Bigg\} \\
\implies ||\hat{e_{i2}}|| &\leq m_1^2\Bigg\{h^{2p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(1+\delta) + h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(1+\delta) + ||\mathbf{a}|| \\
&+ h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)^2 + h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(1+\delta) + h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)^2 \\
&+ h^{4p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2(1+\delta)\Bigg\}
\end{aligned}
$$

$$
\begin{aligned}
\implies ||\hat{e_{i2}}|| \leq m_1^2 \Big\{ & h^{2p-1}(h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + 2h^{-p-\nu^2-\xi} + 2h^{-2p-2\nu^2-\xi} + h^{-p-\nu^2-2\xi} + h^{-2\xi}) \\
& + h^{3p-1}(h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + 2h^{-p-\nu^2-\xi} + 2h^{-2p-2\nu^2-\xi} + h^{-p-\nu^2-2\xi} + h^{-2\xi}) \\
& + ||\mathbf{a}|| \\
& + h^{3p-1}(h^{-p-\nu^2} + h^{-3p-3\nu^2} + 2h^{-2p-2\nu^2} + h^{-2p-2\nu^2-\xi} + 2h^{-p-\nu^2-\xi} + h^{-\xi}) \\
& + h^{3p-1}(h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + 2h^{-p-\nu^2-\xi} + 2h^{-2p-2\nu^2-\xi} + h^{-p-\nu^2-2\xi} + h^{-2\xi}) \\
& + h^{3p-1}(h^{-p-\nu^2} + h^{-3p-3\nu^2} + 2h^{-2p-2\nu^2} + h^{-2p-2\nu^2-\xi} + 2h^{-p-\nu^2-\xi} + h^{-\xi}) \\
& + h^{4p-1}(h^{-2p-2\nu^2} + h^{-3p-3\nu^2} + 2h^{-p-\nu^2-\xi} + 2h^{-2p-2\nu^2-\xi} + h^{-p-\nu^2-2\xi} + h^{-2\xi}) \Big\} \\
\implies ||\hat{e_{i2}}|| \leq m_1^2 \Big\{ & h^{p-1}(h^{-p-2\nu^2} + h^{-2p-3\nu^2} + 2h^{-\nu^2-\xi} + 2h^{-p-2\nu^2-\xi} + h^{-\nu^2-2\xi} + h^{p-2\xi}) \\
& + h^{p-1}(h^{-2\nu^2} + h^{-p-3\nu^2} + 2h^{-\nu^2+p-\xi} + 2h^{-2\nu^2-\xi} + h^{-\nu^2+p-2\xi} + h^{2p-2\xi}) \\
& + ||\mathbf{a}|| \\
& + h^{p-1}(h^{p-\nu^2} + h^{-p-3\nu^2} + 2h^{-2\nu^2} + h^{-2\nu^2-\xi} + 2h^{-\nu^2+p-\xi} + h^{2p-\xi}) \\
& + h^{p-1}(h^{-2\nu^2} + h^{-p-3\nu^2} + 2h^{-\nu^2+p-\xi} + 2h^{-2\nu^2-\xi} + h^{-\nu^2+p-2\xi} + h^{2p-2\xi}) \\
& + h^{p-1}(h^{p-\nu^2} + h^{-2p-3\nu^2} + 2h^{-2\nu^2} + h^{-2\nu^2-\xi} + 2h^{-\nu^2+p-\xi} + h^{2p-\xi}) \\
& + h^{p-1}(h^{p-2\nu^2} + h^{-3\nu^2} + 2h^{p-\nu^2+p-\xi} + 2h^{-2\nu^2+p-\xi} + h^{-\nu^2+2p-2\xi} + h^{3p-2\xi}) \Big\}
\end{aligned}
$$

Now let us find a bound for $||\mathbf{a}||$.

$$
\begin{aligned}
\mathbf{a} &= \sum_{\substack{j=1 \\ j \neq i}}^{h} q_{ij}(W_i^\top A_i^*)(W_j^\top A_j^*)W_j \\
&= \langle W_i, A_i^* \rangle q_{ij} W_{-j}^\top \text{diag}(W_{-j} A_{-j}^*)
\end{aligned}
$$

Where A_{-j}^* is the dictionary A^* with the jth column set to zero, W_{-j} is the dictionary W with the jth row set to zero, and $\text{diag}(W_{-j}A_{-j}^*)$ is the h-dimensional vector containing the diagonal elements of the matrix $W_{-j}A_{-j}^*$. We also make use of the distributional assumption that q_{ij} is the same for all i, j in order to pull

q_{ij} out of the sum.

$$
\begin{aligned}
||\mathbf{a}||_2 &= h^{2p-2}\langle W_i, A_i^*\rangle ||W_{-j}^\top \text{diag}(W_{-j}A_{-j}^*)||_2 \\
&\leq h^{2p-2}(1+\delta)||W_{-j}^\top||_2 ||\text{diag}(W_{-j}A_{-j}^*)||_2 \\
&\leq h^{2p-2}(1+\delta)^2 h^{1/2}\sqrt{\lambda_{\max}(W_{-j}^\top W_{-j})} \\
&\leq h^{2p-2}(1+\delta)^2 h^{1/2}\sqrt{h\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)+(1+\delta)^2} \\
&= h^{p-1}\sqrt{h^{2p-2}\times h\times(1+\delta)^4\times\left(h\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)+(1+\delta)^2\right)} \\
&= h^{p-1}\sqrt{h^{2p-1}\times(1+h^{-p-\nu^2})^4\times\left(h(h^{-2p-2\nu^2}+2h^{-p-\nu^2}+h^{-\xi})+(1+h^{-p-\nu^2})^2\right)} \\
&= h^{p-1}\sqrt{(1+h^{-p-\nu^2})^4\times(h^{-2\nu^2}+2h^{p-\nu^2}+h^{2p-\xi}+h^{2p-1}(1+h^{-p-\nu^2})^2)}
\end{aligned}
$$

Here $||W_{-j}^\top||_2$ is the spectral norm of $W_{-j}^\top$, and is the top singular value of the matrix. We use Gershgorin's Circle theorem to bound the top eigenvalue of $W_{-j}^\top W_{-j}$ by its maximum row sum.

If $p<\frac{\xi}{2}$, $p<\frac{1}{2}$, and $p<\nu^2$, then $||\hat{e_{i2}}||= o(m_1^2 h^{p-1})$
And now we start to estimate $\hat{e_{i3}}$ as follows.

$$
\begin{aligned}
\hat{e_{i3}} &= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{Dm_1\sum_{\substack{j\in S\\ j\neq i}}\epsilon_i A_j^* - 2Dm_1^2\sum_{\substack{j,k\in S\\ j\neq i\\ k\neq i}}(W_i^\top A_k^*)A_j^*\right\}\right] \\
&= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{Dm_1\sum_{\substack{j\in S\\ j\neq i}}\epsilon_i A_j^* - 2Dm_1^2\sum_{\substack{j\in S\\ j\neq i}}(W_i^\top A_j^*)A_j^* - 2Dm_1^2\sum_{\substack{j,k\in S\\ k\neq j\neq i}}(W_i^\top A_k^*)A_j^*\right\}\right] \\
&= Dm_1\sum_{\substack{j=1\\ j\neq i}}^{h}\epsilon_i A_j^*\sum_{\{S\in\mathbb{S}:i,j\in S,i\neq j\}}q_S - 2Dm_1^2\sum_{\substack{j=1\\ j\neq i}}^{h}(W_i^\top A_j^*)A_j^*\sum_{\{S\in\mathbb{S}:i,j\in S,i\neq j\}}q_S \\
&- 2Dm_1^2\sum_{\substack{j,k=1\\ k\neq j\neq i}}^{h}(W_i^\top A_k^*)A_j^*\sum_{\{S\in\mathbb{S}:i,j,k\in S,i\neq j\neq k\}}q_S \\
&= Dm_1\sum_{\substack{j=1\\ j\neq i}}^{h}q_{ij}\epsilon_i A_j^* - 2Dm_1^2\sum_{\substack{j=1\\ j\neq i}}^{h}q_{ij}(W_i^\top A_j^*)A_j^* - 2Dm_1^2\sum_{\substack{j,k=1\\ k\neq j\neq i}}^{h}q_{ijk}(W_i^\top A_k^*)A_j^*
\end{aligned}
$$

We plugin $\epsilon_i = 2m_1h^p\left(\delta+\frac{\mu}{\sqrt{n}}\right)$ for $i=1,\ldots,h$

$$
\begin{aligned}
||\hat{e_{i3}}|| &\leq 2Dm_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)+2Dm_1^2h^{2p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)+2Dm_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right) \\
&= 4Dm_1^2h^{3p-1}(h^{-p-\nu^2}+h^{-\xi})+2Dm_1^2h^{2p-1}(h^{-p-\nu^2}+h^{-\xi}) \\
&= 4Dm_1^2h^{p-1}(h^{p-\nu^2}+h^{2p-\xi})+2Dm_1^2h^{p-1}(h^{-\nu^2}+h^{p-\xi})
\end{aligned}
$$

This means for $D = 1$, $p < \nu^2$ and $p < \frac{\xi}{2}$, we have $||\hat{e_{i3}}|| = o(m_1^2 h^{p-1})$
And now we start to estimate $\hat{e_{i4}}$ as follows.

$$
\begin{aligned}
\hat{e_{i4}} &= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times\left\{-m_1\sum_{\substack{j,k\in S\\ k\neq i}}\epsilon_j(W_i^\top W_j)A_k^* + m_1^2\sum_{\substack{j,k,l\in S\\ k\neq i,l}}(W_i^\top W_j)(W_j^\top A_l^*)A_k^*\right\}\right]\\
&= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times(-m_1)\left\{\sum_{k(=j)\in S\setminus i}\epsilon_k(W_i^\top W_k)A_k^* + \sum_{\substack{j\in S\setminus i\\ k\in S\setminus i,j}}\epsilon_j(W_i^\top W_j)A_k^* + \sum_{\substack{k\in S\setminus i\\ j=i}}\epsilon_j(W_i^\top W_i)A_k^*\right\}\right]\\
&+ \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times m_1^2\left\{\sum_{\substack{j,k,l\in S\\ k\neq i,l}}(W_i^\top W_j)(W_j^\top A_l^*)A_k^*\right\}\right]\\
&= \mathbb{E}_{S\in\mathbb{S}}\left[\mathbf{1}_{i\in S}\times(-m_1)\left\{\sum_{k(=j)\in S\setminus i}\epsilon_k(W_i^\top W_k)A_k^* + \sum_{\substack{j\in S\setminus i\\ k\in S\setminus i,j}}\epsilon_j(W_i^\top W_j)A_k^* + \sum_{\substack{k\in S\setminus i\\ j=i}}\epsilon_j(W_i^\top W_i)A_k^*\right\}\right]\\
&+ \mathbb{E}_{S\in\mathbb{S}}\Bigg[\mathbf{1}_{i\in S}\times m_1^2\Bigg\{\sum_{\substack{k\in S\\ k\neq i}}(W_i^\top W_i)(W_i^\top A_i^*)A_k^* + \sum_{\substack{k\in S\\ k\neq i}}(W_i^\top W_k)(W_k^\top A_i^*)A_k^* + \sum_{\substack{j,k\in S\\ j\neq k\neq i}}(W_i^\top W_j)(W_j^\top A_i^*)A_k^*\\
&+ \sum_{\substack{k,l\in S\\ \neq k\neq i}}(W_i^\top W_i)(W_i^\top A_l^*)A_k^* + \sum_{\substack{k,l\in S\\ l\neq k\neq i}}(W_i^\top W_k)(W_k^\top A_l^*)A_k^* + \sum_{\substack{k,l\in S\\ l\neq k\neq i}}(W_i^\top W_l)(W_l^\top A_l^*)A_k^*\\
&+ \sum_{\substack{j,k,l\in S\\ j\neq k\neq l\neq i}}(W_i^\top W_j)(W_j^\top A_l^*)A_k^*\Bigg\}\Bigg]\\
\hat{e_{i4}} &= (-m_1)\left\{\sum_{k=1,k\neq i}^{h}q_{ik}\epsilon_k(W_i^\top W_k)A_k^* + \sum_{\substack{j,k=1\\ j\neq k\neq i}}^{h}q_{ijk}\epsilon_j(W_i^\top W_j)A_k^* + \sum_{\substack{k=1\\ k\neq i}}^{h}q_{ik}\epsilon_i(W_i^\top W_i)A_k^*\right\}\\
&+ m_1^2\Bigg\{\underbrace{\sum_{\substack{k=1\\ k\neq i}}^{h}q_{ik}(W_i^\top W_i)(W_i^\top A_i^*)A_k^*}_{\mathbf{b}} + \sum_{\substack{k=1\\ k\neq i}}^{h}q_{ik}(W_i^\top W_k)(W_k^\top A_i^*)A_k^* + \sum_{\substack{j,k=1\\ j\neq k\neq i}}^{h}q_{ijk}(W_i^\top W_j)(W_j^\top A_i^*)A_k^*\\
&+ \sum_{\substack{k,l=1\\ l\neq k\neq i}}^{h}q_{ikl}(W_i^\top W_i)(W_i^\top A_l^*)A_k^* + \sum_{\substack{k,l=1\\ l\neq k\neq i}}^{h}q_{ikl}(W_i^\top W_k)(W_k^\top A_l^*)A_k^* + \sum_{\substack{k,l=1\\ l\neq k\neq i}}^{h}q_{ikl}(W_i^\top W_l)(W_l^\top A_l^*)A_k^*\\
&+ \sum_{\substack{j,k,l=1\\ j\neq k\neq l\neq i}}^{h}q_{ijkl}(W_i^\top W_j)(W_j^\top A_l^*)A_k^*\Bigg\}
\end{aligned}
$$

We plugin $\epsilon_i = 2m_1h^p\left(\delta + \frac{\mu}{\sqrt{n}}\right)$ for $i = 1, \ldots, h$ in the above to get,

$$
\begin{aligned}
||\hat{e_{i4}}|| &\leq 2m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2+2m_1^2h^{4p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)\\
&+2m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)^2\\
&+m_1^2||\mathbf{b}||+m_1^2h^{2p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)+m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)\\
&+m_1^2h^{3p-1}(1+\delta)^2\left(\delta+\frac{\mu}{\sqrt{n}}\right)+m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)\\
&+m_1^2h^{3p-1}(1+\delta)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)\\
&+m_1^2h^{4p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)\\
\implies ||\hat{e_{i4}}|| &\leq 2m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)^2+3m_1^2h^{4p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)\\
&+3m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)(1+\delta)^2\\
&+m_1^2||\mathbf{b}||+m_1^2h^{2p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)+2m_1^2h^{3p-1}\left(\delta+\frac{\mu}{\sqrt{n}}\right)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)\\
&+m_1^2h^{3p-1}(1+\delta)\left(\delta^2+2\delta+\frac{\mu}{\sqrt{n}}\right)\\
\implies ||\hat{e_{i4}}|| &\leq 2m_1^2h^{3p-1}(h^{-2p-2\nu^2}+2h^{-p-\nu^2-\xi}+h^{-2\xi})\\
&+3m_1^2h^{4p-1}(h^{-3p-3\nu^2}+2h^{-2p-2\nu^2}+3h^{-p-\nu^2-\xi}+h^{-2p-2\nu^2-\xi}+h^{-2\xi})\\
&+3m_1^2h^{3p-1}(h^{-3p-3\nu^2}+2h^{-2p-2\nu^2}+2h^{-p-\nu^2-\xi}+h^{-2p-2\nu^2-\xi}+h^{-\xi}+h^{-p-\nu^2})\\
&+m_1^2||\mathbf{b}||\\
&+m_1^2h^{2p-1}(h^{-3p-3\nu^2}+2h^{-2p-2\nu^2}+3h^{-p-\nu^2-\xi}+h^{-2p-2\nu^2-\xi}+h^{-2\xi})\\
&+2m_1^2h^{3p-1}(h^{-3p-3\nu^2}+2h^{-2p-2\nu^2}+3h^{-p-\nu^2-\xi}+h^{-2p-2\nu^2-\xi}+h^{-2\xi})\\
&+m_1^2h^{3p-1}(h^{-3p-3\nu^2}+3h^{-2p-2\nu^2}+h^{-p-\nu^2-\xi}+h^{-\xi}+2h^{-p-\nu^2})
\end{aligned}
$$

$$
\begin{aligned}
\implies ||\hat{e_{i4}}|| &\leq 2m_1^2h^{p-1}(h^{-2\nu^2}+2h^{-\nu^2+p-\xi}+h^{2p-2\xi})\\
&+3m_1^2h^{p-1}(h^{-3\nu^2}+2h^{-p-2\nu^2}+3h^{p-\nu^2+p-\xi}+h^{-2\nu^2+p-\xi}+h^{3p-2\xi})\\
&+3m_1^2h^{p-1}(h^{-p-3\nu^2}+2h^{-2\nu^2}+2h^{-\nu^2+p-\xi}+h^{-2\nu^2-\xi}+h^{2p-\xi}+h^{p-\nu^2})\\
&+m_1^2||\mathbf{b}||\\
&+m_1^2h^{p-1}(h^{-2p-3\nu^2}+2h^{-p-2\nu^2}+3h^{-\nu^2-\xi}+h^{-p-2\nu^2-\xi}+h^{p-2\xi})\\
&+2m_1^2h^{p-1}(h^{-p-3\nu^2}+2h^{-2\nu^2}+3h^{-\nu^2+p-\xi}+h^{-2\nu^2-\xi}+h^{2p-2\xi})\\
&+m_1^2h^{p-1}(h^{-p-3\nu^2}+3h^{-2\nu^2}+h^{-\nu^2+p-\xi}+h^{2p-\xi}+2h^{p-\nu^2})
\end{aligned}
$$

Now let us find a bound for $||\mathbf{b}||$.

$$\begin{aligned}\mathbf{b} &= \sum_{\substack{k=1\\k\neq i}}^{h} q_{ik}(W_i^\top W_i)(W_i^\top A_i^*)A_k^* \\ &= \langle W_i, W_i\rangle\langle W_i, A_i^*\rangle q_{ik} A_{-i}^* \mathbf{1}_h\end{aligned}$$

Where A_{-i}^* is the dictionary A^* with the ith column set to zero, and $\mathbf{1}_h \in \mathbb{R}^h$ is the h-dimensional vector of all ones. Here we make use of the distributional assumption that q_{ik} is the same for all i, k in order to pull q_{ik} out of the sum.

$$\begin{aligned}||\mathbf{b}||_2 &= h^{2p-2}\langle W_i, W_i\rangle\langle W_i, A_i^*\rangle||A_{-i}^*\mathbf{1}_h||_2 \\ &\leq h^{2p-2}(1+\delta)^3||A_{-i}^*||_2||\mathbf{1}_h||_2 \\ &= h^{2p-2}(1+\delta)^3 h^{1/2}\sqrt{\lambda_{\max}(A_{-i}^{*\top}A_{-i}^*)} \\ &= h^{2p-2}(1+\delta)^3 h^{1/2}\sqrt{h\frac{\mu}{\sqrt{n}}+1} \\ &= h^{p-1}\sqrt{h^{2p-2}\times h\times(1+\delta)^6\times\left(h\frac{\mu}{\sqrt{n}}+1\right)} \\ &= h^{p-1}\sqrt{h^{2p-1}\times(1+h^{-p-\nu^2})^6\times(h^{1-\xi}+1)} \\ &= h^{p-1}\sqrt{(1+h^{-p-\nu^2})^6\times(h^{2p-\xi}+h^{2p-1})}\end{aligned}$$

Here $||A_{-i}^*||_2$ is the spectral norm of A_{-i}^*, and is the top singular value of the matrix. We use Gershgorin's Circle theorem to bound the top eigenvalue of $A_{-i}^{*\top}A_{-i}^*$ by its maximum row sum.
If $p < \frac{\xi}{2}$, $p < \frac{1}{2}$, and $p < \nu^2$, then $||\hat{e_{i4}}|| = o(m_1^2 h^{p-1})$. Now we combine the above obtained bounds for $\|\hat{e_{it}}\|$ (for $t \in \{1,2,3,4\}$) with the bound obtained below equation 8 to say that, $\|e_i\| = o(\max\{m_1^2, m_2\}h^{p-1})$

9.3 About $\alpha_i - \beta_i$

Remembering that $D = 1$ and doing a close scrutiny of the terms in 7 and 5 will indicate that the coefficients are the *same* for the $m_2 h^{p-1}$ term in each of them. (which is the term with the highest h scaling in the m_2 dependent parts of α_i and β_i). So this largest term cancels off in the difference and we are left with the sub-leading order terms coming from both their m_1^2 as well as the m_2 parts and this gives us,

$$\alpha_i - \beta_i = o(\max\{m_1^2, m_2\}h^{p-1})$$