Training diagonal linear networks
Semester project report

Salim Najib
Supervised by Antoine Bodin and Nicolas Macris

June 2022

Contents

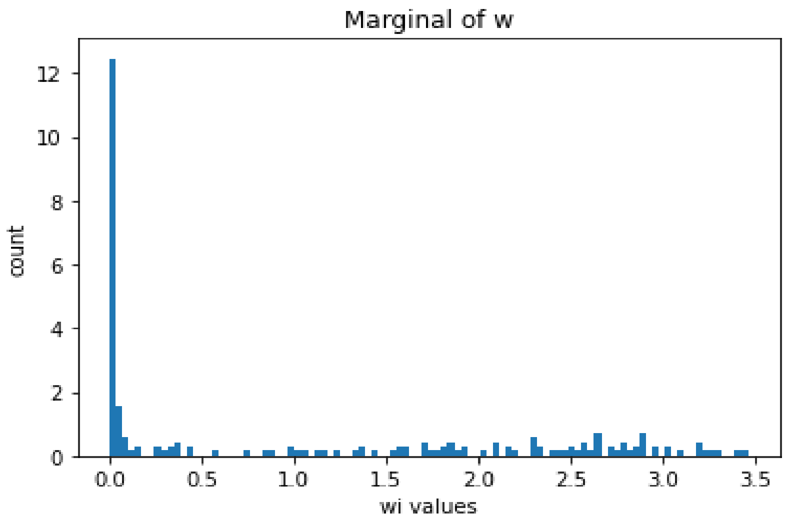
Introduction

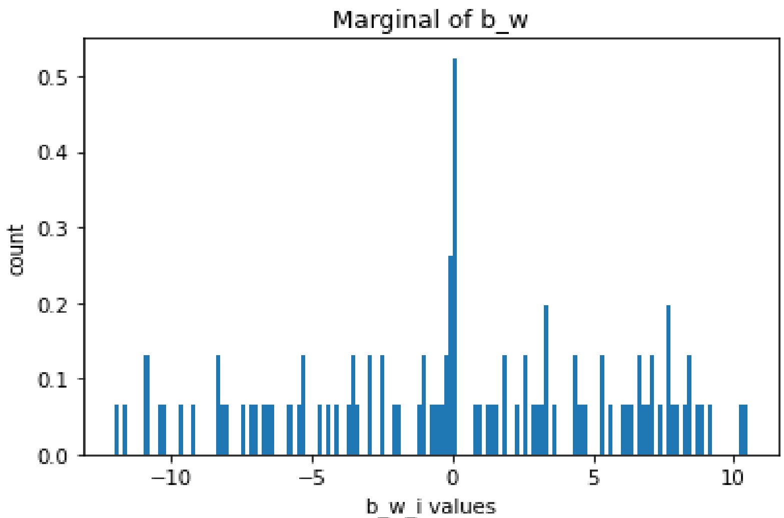
Diagonal linear networks are a toy model that has been studied to uncover phenomena taking place in larger neural networks. In this project, motivated by [1] we have sequentially studied - or tried to study - the sparsity of the parameters obtained through stochastic gradient descent training, the impact of the structure and norm of the initialization vector, and finally we have reached a related result in what we have called perturbed ridge regression.

Chapter 1

Sparsity of SGD solution

This chapter is based on numerical computation, thus we defer the reader's attention to the associated notebook `diagonal_networks.ipynb`, link: https://github.com/Dicedead/diagonalNetworksProject/blob/main/diagonal_networks.ipynb where the diagonal linear neural network model has been implemented - it is thus defined there.
Here, we will simply plot the obtained marginals of SGD solutions:





The sparsity discussed by [1] is numerically observed. However, getting a closed form for the marginal of β_w in full generality seems to be an intractable problem in the context of this project, even though f_{β_w} looks like a gaussian + a delta at 0. Therefore, in the next section, we will look at a more specific case.

Chapter 2

Characterizing initialization's impact on test error for the kernel regime

2.1 Introduction, setup and goals

We set out to study least squares interpolation, motivated by [2] and by findings in the kernel regime of [1]. Indeed, in the latter paper, when the initialization hyperparameter $\alpha \to \infty$ and we take $\beta_0 = (\alpha)_{i \in [\![1..d]\!]}$, $\beta_\infty^\alpha = \arg\min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = y}} \phi_{\alpha_\infty}(\beta) \to \arg\min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = y}} \frac{1}{16\alpha^2} ||\beta||_2^2$, because $\alpha_\infty \to \alpha$, which simplifies to minimum l_2 norm least squares:

$$\beta_\alpha^\infty = \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = y}} ||\beta||_2^2 = \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = y}} \beta^T I_d \beta$$

But what happens when β_0 is not a vector with constant coefficients? Do we improve the training and test errors by choosing β_0 not to be constant, but rather, for example, split into two constant halves?

The setup is the following. Assume we are given n samples $x_1, \ldots, x_n \overset{\text{i.i.d}}{\sim} P_x$ in $\mathbb{R}^d$ and $\beta^* \sim P_{\beta^*}$ in $\mathbb{R}^d$, such that the distribution P_{β^*} has mean $0 \in \mathbb{R}^d$ and covariance matrix $\Sigma = I_d \in \mathbb{R}^{d \times d}$, and P_x has mean $0 \in \mathbb{R}^d$ and covariance matrix I_d. β^* and x_i are independent $\forall i \in [\![1..n]\!]$.
Then, we define $y_i = x_i^T \beta^* + \epsilon_i$, $\forall i \in [\![1..n]\!]$ where $\epsilon_1, \ldots, \epsilon_n \overset{\text{i.i.d}}{\sim} P_\epsilon$ in $\mathbb{R}$ a distribution with mean $0 \in \mathbb{R}$ which are independent from β^* and all x_i.

Now, we estimate β^* using least-squares linear regression *with* ***weighted*** *minimum* l_2 *norm*, and our interest lies especially in the over-parameterized case where $d > n$. That is, denoting this estimate by $\beta \in \mathbb{R}^d$:

$$\beta = \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = y}} \beta^T \Lambda \beta$$

where the equation $X\beta = y$ is understood in the least squares sense, and:

$$X = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix} \in \mathbb{R}^{n \times d}, \quad \Lambda = \operatorname{diag}\left(\frac{1}{\alpha^2}\right) = \begin{bmatrix} \frac{1}{\alpha_1^2} & 0 & 0 & \ldots & 0 \\ 0 & \frac{1}{\alpha_2^2} & 0 & \ldots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & \ldots & 0 & \frac{1}{\alpha_d^2} \end{bmatrix} \in \mathbb{R}^{d \times d} \text{ with } \alpha \in (\mathbb{R}_+^*)^d.$$

Define the training and test errors:

$$\mathrm{R}^{\text{train}} = \mathbb{E}_{\beta^*}\left[||X\beta - y||_2^2\right]$$

$$\mathrm{R}^{\text{test}} = \mathbb{E}_{x,\beta^*}\left[(x^T(\beta - \beta^*))^2\right]$$

where $x \sim P_x$ is independent from β^*.
In the following, we also set α as the concatenation of potentially differently sized vectors with total

dimensionality d, with $q \in [\![0..d]\!]$ and $\gamma = \frac{q}{d}$:

$$\alpha = \left(\underbrace{\alpha_1}_{[1,d\gamma]} \mid \underbrace{\alpha_2}_{[d\gamma+1,d]} \right) \in \mathbb{R}^d$$

Observe that without loss of generality, by factoring by α_2^{-2} in $\beta\Lambda\beta^*$, it is sufficient to consider the case where $\alpha_2 = 1$ and $\alpha_1 \in \mathbb{R}_+^*$.

Now, the hope is to obtain an expression for R^{test} and show that they depend only on γ, α_1 and $\frac{n}{d}$.

2.2 Noiseless case

Assume P_ϵ has variance $0 \in \mathbb{R}$, that is: $\forall i \in [\![1..n]\!]$ $y_i = x_i^T \beta^*$, thus $y = X\beta^*$. Then the training error simplifies to:

$$\mathrm{R}^{\text{train}} = \mathbb{E}_{\beta^*} \left[||X(\beta - \beta^*)||_2^2 \right]$$

and the feasible set of the optimization also simplifies:

$$\beta = \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = X\beta^*}} \beta^T \Lambda \beta$$

since this time, the equation $X\beta = X\beta^*$ has at least one solution, β^*, and is thus no longer a linear system in the least squares sense.

A case studied with greater generality in [2] (Theorem 1, page 10) is when $\alpha_1 = 1 = \alpha_2$, and thus $\Lambda = I_d$. In that case, the optimization problem $\beta = \arg\min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = X\beta^*}} \beta^T \beta$ is simply the least squares problem with minimum l_2 norm, and setting $p = \frac{d}{n}$:

$$A = (X^T X)^+ X^T X$$

$$\beta = (X^T X)^+ X^T X \beta^*$$

$$\mathrm{R}^{\text{test}} = \begin{cases} \frac{p}{1-p} & \text{if } p < 1 \\ \frac{1}{p-1} & \text{if } p > 1 \end{cases}$$

We can now try to reduce the more general case where $\alpha_1 := \alpha \in \mathbb{R}_+^*$ to this special case, by considering block matrices. Recall that we had set $\alpha_2 = 1$ without loss of generality, since one can consider $\alpha = \frac{\alpha_1}{\alpha_2}$ equivalently. It may be useful to remember that α is equal to the ratio of the two chosen initialization values.

$$\begin{aligned}
\beta &= \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = X\beta^*}} \beta^T \Lambda \beta \\
&= \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = X\beta^*}} \beta^T \begin{bmatrix} \frac{1}{\alpha^2} & 0 & 0 & \dots & \dots & 0 \\ 0 & \frac{1}{\alpha^2} & 0 & \dots & \dots & 0 \\ \vdots & \vdots & & & \vdots & \vdots \\ 0 & \dots & \frac{1}{\alpha^2} & 0 & \dots & 0 \\ 0 & \dots & 0 & 1 & \dots & 0 \\ \vdots & \vdots & & & \vdots & \vdots \\ 0 & \dots & \dots & & 0 & 1 \end{bmatrix} \beta \\
&= \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = X\beta^*}} \sum_{i=1}^{d\gamma} \frac{1}{\alpha^2} \beta_i^2 + \sum_{i=d\gamma+1}^{d} \beta_i^2 \\
&= \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = X\beta^*}} \frac{1}{\alpha^2} ||\beta_{[1,d\gamma]}||_2^2 + ||\beta_{[d\gamma+1,d]}||_2^2
\end{aligned}$$

For concision, denote $s_d = [1, d\gamma]$ and $e_d = [d\gamma + 1, d]$, thus $\beta_{[1,d\gamma]}$ by β_{s_d} (s for start) and $\beta_{[d\gamma+1,d]}$ by β_{e_d} (e for end).

$$\beta = \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = X\beta^*}} \frac{1}{\alpha^2}||\beta_{s_d}||_2^2 + ||\beta_{e_d}||_2^2$$

Preparing for block matrix operations, we'll write $\Phi = X^T X$:

$$\Phi = X^T X = \begin{bmatrix} \Phi_{s_d,s_d} & \Phi_{s_d,e_d} \\ \Phi_{e_d,s_d} & \Phi_{e_d,e_d} \end{bmatrix} \in \mathbb{R}^{d \times d}$$

Since the rows of $X \in \mathbb{R}^{n \times d}$ are continuous random vectors and (statistically) independent, it is almost sure that the d columns and n rows of X are linearly independent, thus, almost surely: $\text{rank}(X) = \min(d, n)$. Expanding the interpolation requirement:

$$\begin{aligned} & X\beta = X\beta^* \\ \implies & X^T X\beta = X^T X\beta^* \\ \iff & \Phi\beta = \Phi\beta^* \\ \iff & \begin{bmatrix} \Phi_{s_d,s_d} & \Phi_{s_d,e_d} \\ \Phi_{e_d,s_d} & \Phi_{e_d,e_d} \end{bmatrix} \begin{bmatrix} \beta_{s_d} \\ \beta_{e_d} \end{bmatrix} = \begin{bmatrix} \Phi_{s_d,s_d} & \Phi_{s_d,e_d} \\ \Phi_{e_d,s_d} & \Phi_{e_d,e_d} \end{bmatrix} \begin{bmatrix} \beta^*_{s_d} \\ \beta^*_{e_d} \end{bmatrix} \\ \iff & \begin{bmatrix} \beta_{s_d} \\ \beta_{e_d} \end{bmatrix} = \begin{cases} \Phi^{-1}\Phi\beta^* = \beta^* & \text{if } d \leq n \\ \Phi^+\Phi\beta^* & \text{if } d > n \end{cases} \end{aligned}$$

Notice that the noiseless case is not interesting when $d \leq n$. In the rest of this section, we take $d > n$. Instead of considering pseudoinverses, we can consider the equivalence between the following problems as introduced in [2]:

$$\beta = \arg \min_{\substack{\beta \in \mathbb{R}^d \\ X\beta = X\beta^*}} \beta^T \Lambda \beta = \lim_{\lambda \to 0} \beta_\lambda = \lim_{\lambda \to 0} \overbrace{\arg \min_{\beta \in \mathbb{R}^d} ||X(\beta - \beta^*)||_2^2 + \lambda \beta^T \Lambda \beta}^{:= f_\lambda(\beta)}$$

f_λ is differentiable, thus we can compute its gradient:

$$\nabla f_\lambda(\beta) = 2X^T X\beta + 2\lambda\Lambda\beta - 2X^T y$$

And setting the gradient to 0, since this function is ($2\lambda-$strongly) convex, we yield:

$$\beta = (X^T X + \lambda\Lambda)^{-1} X^T y$$

Note that this expression is also valid in the noisy case, so we will add noise to the problem before continuing.

2.3 Noisy case

Let's go back to $d \leq n$ in the noisy case, where the variance of P_ϵ is σ^2. Here $y = X\beta^* + \epsilon$ with $\epsilon = (\epsilon_i)_{i \in [\![1..n]\!]}$, thus $\epsilon \sim (\mu_\epsilon = 0 \in \mathbb{R}^n, \Sigma_\epsilon = \sigma^2 I_n)$.

$$\begin{aligned} & X\beta = y \\ \implies & X^T X\beta = X^T y \\ \iff & \Phi\beta = X^T(X\beta^* + \epsilon) \\ \iff & \Phi\beta = \Phi\beta^* + X^T\epsilon \\ \iff & \beta = \beta^* + \Phi^{-1}X^T\epsilon \\ \iff & \beta = \beta^* + \begin{bmatrix} \Phi_{s_d,s_d} & \Phi_{s_d,e_d} \\ \Phi_{e_d,s_d} & \Phi_{e_d,e_d} \end{bmatrix}^{-1} X^T\epsilon \end{aligned}$$

Computing $\Phi^{-1}X^T$ using standard block matrix inversion and multiplication formulae (see [3]):

$$\Phi^{-1}X^T = \begin{bmatrix} \Phi_{s_d,s_d} & \Phi_{s_d,e_d} \\ \Phi_{e_d,s_d} & \Phi_{e_d,e_d} \end{bmatrix}^{-1} X^T$$

$$\Phi^{-1} = \begin{bmatrix} (\Phi_{s_d,s_d} - \Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1}\Phi_{e_d,s_d})^{-1} & 0 \\ 0 & (\Phi_{e_d,e_d} - \Phi_{e_d,s_d}(\Phi_{s_d,s_d})^{-1}\Phi_{s_d,e_d})^{-1} \end{bmatrix} \cdot \underbrace{\begin{bmatrix} I_{d\gamma} & -\Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1} \\ -\Phi_{e_d,s_d}(\Phi_{s_d,s_d})^{-1} & I_{d-d\gamma} \end{bmatrix}}_{:=\kappa}$$

The matrix on top is scary. Let's focus on κX^T. First, notice that κ, in the expression above, is decomposed into blocks of the following sizes:

$$\kappa = \begin{bmatrix} I_{d\gamma} & -\Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1} \\ -\Phi_{e_d,s_d}(\Phi_{s_d,s_d})^{-1} & I_{d-d\gamma} \end{bmatrix} \in \begin{bmatrix} \mathbb{R}^{d\gamma\times d\gamma} & \mathbb{R}^{d\gamma\times d-d\gamma} \\ \mathbb{R}^{d-d\gamma\times d\gamma} & \mathbb{R}^{d-d\gamma\times d-d\gamma} \end{bmatrix}$$

We'll compute κX^T by first decomposing $X^T \in \mathbb{R}^{d\times n}$ into 4 blocks of product compatible sizes, as follows:

$$X^T \in \begin{bmatrix} \mathbb{R}^{d\gamma\times a} & \mathbb{R}^{d\gamma\times b} \\ \mathbb{R}^{d-d\gamma\times a} & \mathbb{R}^{d-d\gamma\times b} \end{bmatrix}$$

Here, a should be chosen such that $a = n$ when $\gamma = 1$, and $b = n$ when $\gamma = 0$, then $a + b = n$. Thus: $a = n\gamma$ and $b = n - n\gamma$, and:

$$X^T \in \begin{bmatrix} \mathbb{R}^{d\gamma\times n\gamma} & \mathbb{R}^{d\gamma\times n-n\gamma} \\ \mathbb{R}^{d-d\gamma\times n\gamma} & \mathbb{R}^{d-d\gamma\times n-n\gamma} \end{bmatrix}$$

Denoting $s_n = [1, n\gamma]$ and $e_n = [n\gamma + 1, n]$:

$$X^T = \begin{bmatrix} X^T_{s_d,s_n} & X^T_{s_d,e_n} \\ X^T_{e_d,s_n} & X^T_{e_d,e_n} \end{bmatrix}$$

This also gives us a suitable block decomposition of X:

$$X = (X^T)^T = \begin{bmatrix} X^T_{s_d,s_n} & X^T_{s_d,e_n} \\ X^T_{e_d,s_n} & X^T_{e_d,e_n} \end{bmatrix}^T = \begin{bmatrix} X_{s_n,s_d} & X_{s_n,e_d} \\ X_{e_n,s_d} & X_{e_n,e_d} \end{bmatrix}$$

$$X^T = \begin{bmatrix} (X_{s_n,s_d})^T & (X_{e_n,s_d})^T \\ (X_{s_n,e_d})^T & (X_{e_n,e_d})^T \end{bmatrix}$$

And now, the blocks of $\Phi = X^T X$ can be made more explicit:

$$\Phi = \begin{bmatrix} \Phi_{s_d,s_d} & \Phi_{s_d,e_d} \\ \Phi_{e_d,s_d} & \Phi_{e_d,e_d} \end{bmatrix} = \begin{bmatrix} (X_{s_n,s_d})^T X_{s_n,s_d} + (X_{e_n,s_d})^T X_{e_n,s_d} & (X_{s_n,s_d})^T X_{s_n,e_d} + (X_{e_n,s_d})^T X_{e_n,e_d} \\ (X_{s_n,e_d})^T X_{s_n,s_d} + (X_{e_n,e_d})^T X_{e_n,s_d} & (X_{s_n,e_d})^T X_{s_n,e_d} + (X_{e_n,e_d})^T X_{e_n,e_d} \end{bmatrix}$$

Some more preparatory computations before computing κX^T; computing the blocks of B:

$$\begin{aligned} & -\Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1} = -\left[(X_{s_n,s_d})^T X_{s_n,e_d} + (X_{e_n,s_d})^T X_{e_n,e_d}\right]\left[(X_{s_n,e_d})^T X_{s_n,e_d} + (X_{e_n,e_d})^T X_{e_n,e_d}\right]^{-1} \\ \iff & \Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1}\left[(X_{s_n,e_d})^T X_{s_n,e_d} + (X_{e_n,e_d})^T X_{e_n,e_d}\right] = (X_{s_n,s_d})^T X_{s_n,e_d} + (X_{e_n,s_d})^T X_{e_n,e_d} \\ \iff & \left[(X_{s_n,e_d})^T X_{s_n,e_d} + (X_{e_n,e_d})^T X_{e_n,e_d}\right]^T (\Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1})^T = \left[(X_{s_n,s_d})^T X_{s_n,e_d} + (X_{e_n,s_d})^T X_{e_n,e_d}\right]^T \end{aligned}$$

Set $W^T = \Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1}$. Thus, we seek the matrix $W \in \mathbb{R}^{d-d\gamma\times d\gamma}$ such that:

$$\left[(X_{s_n,e_d})^T X_{s_n,e_d} + (X_{e_n,e_d})^T X_{e_n,e_d}\right] W = (X_{s_n,e_d})^T X_{s_n,s_d} + (X_{e_n,e_d})^T X_{e_n,s_d}$$

This has the structure:

$$(A^T A + B^T B)W = A^T C + B^T D$$

with $A = X_{s_n,e_d} \in \mathbb{R}^{n\gamma\times d-d\gamma}$ and $B = X_{e_n,e_d} \in \mathbb{R}^{n-n\gamma\times d-d\gamma}$ thus it suffices that $AW = C$ and $BW = D$. To get some intuition on whether this can work or not, assume $n = d$, then B is a square matrix, and $W = B^{-1}D = (X_{e_n,e_d})^{-1}X_{e_n,s_d}$. Is it the case that $AW = C$?

$$AW = X_{s_n,e_d}(X_{e_n,e_d})^{-1}X_{e_n,s_d} \stackrel{?}{=} X_{s_n,s_d} = C$$

Which is not the case...

Similarly, setting $Z^T = \Phi_{e_d,s_d}(\Phi_{s_d,s_d})^{-1}$ with $Z \in \mathbb{R}^{d\gamma \times d-d\gamma}$, we seek for Z such that:

$$\begin{aligned}
&Z^T = \Phi_{e_d,s_d}(\Phi_{s_d,s_d})^{-1} = \left[(X_{s_n,e_d})^T X_{s_n,s_d} + (X_{e_n,e_d})^T X_{e_n,s_d}\right]\left[(X_{s_n,s_d})^T X_{s_n,s_d} + (X_{e_n,s_d})^T X_{e_n,s_d}\right]^{-1} \\
\iff &Z^T \left[(X_{s_n,s_d})^T X_{s_n,s_d} + (X_{e_n,s_d})^T X_{e_n,s_d}\right] = (X_{s_n,e_d})^T X_{s_n,s_d} + (X_{e_n,e_d})^T X_{e_n,s_d} \\
\iff &\left[(X_{s_n,s_d})^T X_{s_n,s_d} + (X_{e_n,s_d})^T X_{e_n,s_d}\right] Z = (X_{s_n,s_d})^T X_{s_n,e_d} + (X_{e_n,s_d})^T X_{e_n,e_d}
\end{aligned}$$

This equation follows the same structure as the one for W mentioned above.

Moving forward on κX^T:

$$\begin{aligned}
\kappa X^T &= \begin{bmatrix} I_{d\gamma} & -\Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1} \\ -\Phi_{e_d,s_d}(\Phi_{s_d,s_d})^{-1} & I_{d-d\gamma} \end{bmatrix} \begin{bmatrix} (X_{s_n,s_d})^T & (X_{e_n,s_d})^T \\ (X_{s_n,e_d})^T & (X_{e_n,e_d})^T \end{bmatrix} \\
&= \begin{bmatrix} (X_{s_n,s_d})^T - \Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1}(X_{s_n,e_d})^T & (X_{e_n,s_d})^T - \Phi_{s_d,e_d}(\Phi_{e_d,e_d})^{-1}(X_{e_n,e_d})^T \\ (X_{s_n,e_d})^T - \Phi_{e_d,s_d}(\Phi_{s_d,s_d})^{-1}(X_{s_n,s_d})^T & (X_{e_n,e_d})^T - \Phi_{e_d,s_d}(\Phi_{s_d,s_d})^{-1}(X_{e_n,s_d})^T \end{bmatrix}
\end{aligned}$$

This is proving to be rather intractable, we should rethink our original problem and see how we can tone it down without losing too much generality.

2.4 The perturbation model

Recall:

$$\beta = KX^T y$$

where $K = (X^T X + \lambda\Lambda)^{-1} \in \mathbb{R}^{d\times d}$ obtained from the noiseless case. We explore a modeling technique for Λ. Namely, we set:

$$\tilde{\Lambda} = I_d + uu^T \in \mathbb{R}^{d\times d}$$

where u is a random vector on $\mathbb{R}^d$ such that it's components are i.i.d, each following a distribution P_u and $\tilde{K} = (X^T X + \lambda\tilde{\Lambda})^{-1} = (\underbrace{X^T X + \lambda I_d}_{:=K_I^{-1}} + \lambda uu^T)^{-1} = (K_I^{-1} + \lambda uu^T)^{-1}$.

By Sherman-Morrison [4]:

$$\tilde{K} = K_I - \lambda \frac{K_I uu^T K_I}{1 + \lambda u^T K_I u}$$

Realizing something quite general, for $K = (X^T X + \lambda\Lambda)^{-1}$ for any matrix Λ that keeps K well defined:

$$\begin{aligned}
\beta - \beta^* &= KX^T(X\beta^* + \epsilon) - \beta^* \\
&= (KX^T X - I)\beta^* + KX^T\epsilon
\end{aligned}$$

Focus on the first term, for some unknown matrix A:

$$\begin{aligned}
&KX^T X = I + A \\
&\iff X^T X = K^{-1} + K^{-1}A = X^T X + \lambda\Lambda + K^{-1}A \\
&\iff A = -\lambda K\Lambda
\end{aligned}$$

Then we can apply this result to $\tilde{K}$: if $\tilde{\beta} = \tilde{K}X^T y$,

$$\begin{aligned}
\tilde{\beta} - \beta^* &= (\tilde{K}X^T X - I)\beta^* + \tilde{K}X^T\epsilon \\
&= -\lambda\tilde{K}(I + uu^T)\beta^* + \tilde{K}X^T\epsilon
\end{aligned}$$

This gives, generalizing slightly with $\beta^* \sim P_{\beta^*}$ with mean 0 and covariance matrix $r^2 I_d$:

$$\begin{aligned}
\mathbb{E}_{\beta^*,\epsilon}\left[||\tilde{\beta} - \beta^*||_2^2\right] &= \mathbb{E}_{\beta^*,\epsilon}\left[||\tilde{K}X^T\epsilon - \lambda\tilde{K}(I + uu^T)\beta^*||_2^2\right] \\
&= \mathbb{E}_\epsilon\left[||\tilde{K}X^T\epsilon||_2^2\right] + \lambda^2\mathbb{E}_{\beta^*}\left[||\tilde{K}(I + uu^T)\beta^*||_2^2\right] - 2\lambda\mathbb{E}_{\beta^*,\epsilon}\left[(\tilde{K}(I + uu^T)\beta^*)^T\tilde{K}X^T\epsilon\right] \\
&= \mathbb{E}_\epsilon\left[||\tilde{K}X^T\epsilon||_2^2\right] + \lambda^2\mathbb{E}_{\beta^*}\left[||\tilde{K}(I + uu^T)\beta^*||_2^2\right] - 2\lambda(\tilde{K}(I + uu^T)\mathbb{E}_{\beta^*}[\beta^*])^T\tilde{K}X^T\mathbb{E}_\epsilon[\epsilon] \\
&= \mathbb{E}_\epsilon\left[\operatorname{Tr}\left(\epsilon^T X\tilde{K}^2 X^T\epsilon\right)\right] + \lambda^2\mathbb{E}_{\beta^*}\left[\operatorname{Tr}\left(\beta^{*T}(I + uu^T)\tilde{K}^2(I + uu^T)\beta^*\right)\right]
\end{aligned}$$

$$
\begin{aligned}
&= \mathrm{Tr}\left(\mathbb{E}_\epsilon\left[\epsilon\epsilon^T\right] X\tilde{K}^2X^T\right) + \lambda^2\mathrm{Tr}\left(\mathbb{E}_{\beta^*}\left[\beta^*\beta^{*T}\right]\tilde{K}^2(I+uu^T)^2\right)\\
&= \sigma^2\mathrm{Tr}(\tilde{K}X^TX\tilde{K}) + r^2\lambda^2\mathrm{Tr}(\tilde{K}^2(I+uu^T)^2)\\
&= \sigma^2\mathrm{Tr}((I-\lambda\tilde{K}(I+uu^T))\tilde{K}) + r^2\lambda^2\mathrm{Tr}(\tilde{K}^2(I+(2+||u||_2^2)uu^T))\\
&= \sigma^2\mathrm{Tr}(\tilde{K}-\lambda(\tilde{K}+\tilde{K}uu^T)\tilde{K}) + r^2\lambda^2\mathrm{Tr}(\tilde{K}^2+(2+||u||_2^2)\tilde{K}uu^T)\\
&= \sigma^2\mathrm{Tr}(\tilde{K}) - \sigma^2\lambda(\mathrm{Tr}(\tilde{K}^2)+\mathrm{Tr}(uu^T\tilde{K}^2)) + r^2\lambda^2\mathrm{Tr}(\tilde{K}^2) + r^2\lambda^2(2+||u||_2^2)\mathrm{Tr}(uu^T\tilde{K})\\
&= \sigma^2\mathrm{Tr}(\tilde{K}) + (r^2\lambda^2-\sigma^2\lambda)\mathrm{Tr}(\tilde{K}^2) + r^2\lambda^2(2+||u||_2^2)\mathrm{Tr}(uu^T\tilde{K}) - \sigma^2\lambda\mathrm{Tr}(uu^T\tilde{K}^2)
\end{aligned}
$$

Let inputs $x_i \sim P_x$ with mean 0 and covariance matrix I_d, we define the test error and recap our findings so far:

$$
\begin{aligned}
\tilde{\mathrm{R}}_d^{\text{test}} &= \mathbb{E}_{x,\epsilon,\beta^*}\left[|x^T\tilde{\beta} - x^T\beta^* + \epsilon|^2\right]\\
&= \sigma^2 + \frac{1}{d}\mathbb{E}_{\beta^*,\epsilon}\left[||\tilde{\beta}-\beta^*||_2^2\right]\\
&= \sigma^2 + \sigma^2\mathrm{Tr}_d(\tilde{K}) + (r^2\lambda^2-\sigma^2\lambda)\mathrm{Tr}_d(\tilde{K}^2) + r^2\lambda^2(2+||u||_2^2)\mathrm{Tr}_d(uu^T\tilde{K}) - \sigma^2\lambda\mathrm{Tr}_d(uu^T\tilde{K}^2)
\end{aligned}
$$

where $\mathrm{Tr}_d(M) = \frac{1}{d}\mathrm{Tr}(M)$.
Say P_u has mean $\mu = 0$ and variance ν^2. $\frac{1}{d}||u||_2^2 \underset{d\to\infty}{\overset{\text{a.s}}{\to}} \mathbb{E}_u[||u||_2^2] = \frac{1}{d}\sum_{i=1}^d \mathbb{E}_{u_i}[u_i^2] = \frac{\nu^2}{d}$, by the law of large numbers. We also have:

$$
\mathrm{Tr}_d(uu^TM) \overset{\text{a.s}}{\to} \mathbb{E}_u\left[\mathrm{Tr}_d(uu^TM)\right] = \nu^2\mathrm{Tr}_d(M)
$$

This derives from:

$$
\begin{aligned}
\mathbb{E}_u\left[\mathrm{Tr}(uu^TM)\right] &= \mathbb{E}_u\left[\mathrm{Tr}(u^TMu)\right]\\
&= \mathrm{Tr}(M\mathbb{E}_u\left[uu^T\right])\\
&= \mathrm{Tr}(M\Sigma_u)\\
&= \nu^2\mathrm{Tr}(M) \text{ as } \Sigma_u = \nu^2 I_d
\end{aligned}
$$

So we can simplify $\tilde{\mathrm{R}}_d^{\text{test}}$:

$$
\begin{aligned}
\tilde{\mathrm{R}}_d^{\text{test}} &= \sigma^2 + \sigma^2\mathrm{Tr}_d(\tilde{K}) + (r^2\lambda^2-\sigma^2\lambda)\mathrm{Tr}_d(\tilde{K}^2) + r^2\lambda^2\nu^2(2+d\nu^2)\mathrm{Tr}_d(\tilde{K}) - \sigma^2\lambda\nu^2\mathrm{Tr}_d(\tilde{K}^2)\\
&= \sigma^2 + \left[\sigma^2 + r^2\lambda^2\nu^2(2+d\nu^2)\right]\mathrm{Tr}_d(\tilde{K}) + \left[r^2\lambda^2 - \sigma^2\lambda(1+\nu^2)\right]\mathrm{Tr}_d(\tilde{K}^2)
\end{aligned}
$$

Next, we compute $\tilde{K}^2$ using the formula obtained through Sherman-Morrison above:

$$
\begin{aligned}
\tilde{K}^2 &= \left(K_I - \lambda\frac{K_Iuu^TK_I}{1+\lambda u^TK_Iu}\right)^2\\
&= K_I^2 - \frac{\lambda}{1+\lambda u^TK_Iu}K_Iuu^TK_I^2 - \frac{\lambda}{1+\lambda u^TK_Iu}K_I^2uu^TK_I + \frac{\lambda^2}{(1+\lambda u^TK_Iu)^2}(K_Iuu^TK_I)^2\\
&= K_I^2 - \frac{\lambda}{1+\lambda u^TK_Iu}\left(K_Iuu^TK_I^2 + K_I^2uu^TK_I\right) + \frac{\lambda^2}{(1+\lambda u^TK_Iu)^2}K_Iuu^TK_I^2uu^TK_I
\end{aligned}
$$

Trace-wise, let's detail a tricky step first:

$$
\mathrm{Tr}(K_Iuu^TK_I^2uu^TK_I) = \mathrm{Tr}(u^TK_I^2uu^TK_I^2u) = u^TK_I^2uu^TK_I^2u = \mathrm{Tr}(u^TK_I^2u)^2 = \mathrm{Tr}(uu^TK_I^2)^2
$$

Then:

$$
\begin{aligned}
\mathrm{Tr}_d(\tilde{K}^2) &= \mathrm{Tr}_d(K_I^2) - \frac{2\lambda}{1+\lambda u^TK_Iu}\mathrm{Tr}_d(uu^TK_I^3) + \frac{\lambda^2}{(1+\lambda u^TK_Iu)^2}\mathrm{Tr}_d(uu^TK_I^2)^2\\
&= \mathrm{Tr}_d(K_I^2) - \frac{2\lambda}{1+\lambda\mathrm{Tr}(uu^TK_I)}\mathrm{Tr}_d(uu^TK_I^3) + \frac{\lambda^2}{(1+\lambda\mathrm{Tr}(uu^TK_I))^2}\mathrm{Tr}_d(uu^TK_I^2)^2
\end{aligned}
$$

For completeness:

$$
\mathrm{Tr}_d(\tilde{K}) \overset{\text{a.s}}{=} \mathrm{Tr}_d(K_I) - \frac{\lambda\nu^2}{1+\lambda\mathrm{Tr}(uu^TK_I)}\mathrm{Tr}_d(K_I^2)
$$

Notice that the denominators have $\mathrm{Tr}(K_I)$ and not $\mathrm{Tr}_d(K_I)$, thus they grow arbitrarily when $d \to \infty$. We can thus substitute in $\tilde{\mathrm{R}}_d^{\text{test}}$:

$$\begin{aligned}\tilde{\mathrm{R}}_d^{\text{test}} &\overset{\text{a.s}}{=} \sigma^2 + \left[\sigma^2 + r^2\lambda^2\nu^2(2 + d\nu^2)\right] \mathrm{Tr}_d(K_I) + \left[r^2\lambda^2 - \sigma^2\lambda(1+\nu^2)\right] \mathrm{Tr}_d(K_I^2) \\ &= \sigma^2 + \left[\sigma^2 + 2r^2\lambda^2\nu^2 + dr^2\lambda^2\nu^4\right] \mathrm{Tr}_d(K_I) + \left[r^2\lambda^2 - \sigma^2\lambda(1+\nu^2)\right] \mathrm{Tr}_d(K_I^2)\end{aligned}$$

We can ask the question: does the test error improve with this added perturbation $u \sim P_u$? In other words, is the test error minimised for $\nu = 0$, and if not, what is the optimal value of ν with respect to the other parameters? We can differentiate with respect to ν to find out, recalling that $\nu \in [0, +\infty[$.

$$\begin{aligned}\frac{\partial \tilde{\mathrm{R}}_d^{\text{test}}}{\partial \nu}(\nu) &= 4r^2\lambda^2\mathrm{Tr}_d(K_I)\left(\nu + d\nu^3\right) - 2\sigma^2\lambda\mathrm{Tr}_d(K_I^2)\nu \\ &= 2\lambda\left(2dr^2\lambda\mathrm{Tr}_d(K_I)\nu^3 + \left[2r^2\lambda\mathrm{Tr}_d(K_I) - \sigma^2\mathrm{Tr}_d(K_I^2)\right]\nu\right) \\ &= 2\lambda\nu\left(2dr^2\lambda\mathrm{Tr}_d(K_I)\nu^2 + 2r^2\lambda\mathrm{Tr}_d(K_I) - \sigma^2\mathrm{Tr}_d(K_I^2)\right) \\ \frac{1}{2\lambda}\frac{\partial^2 \tilde{\mathrm{R}}_d^{\text{test}}}{\partial \nu^2}(\nu) &= 6dr^2\lambda\mathrm{Tr}_d(K_I)\nu^2 + 2r^2\lambda\mathrm{Tr}_d(K_I) - \sigma^2\mathrm{Tr}_d(K_I^2) \\ \frac{1}{2\lambda}\frac{\partial^3 \tilde{\mathrm{R}}_d^{\text{test}}}{\partial \nu^3}(\nu) &= 12dr^2\lambda\mathrm{Tr}_d(K_I)\nu \\ \frac{1}{2\lambda}\frac{\partial^4 \tilde{\mathrm{R}}_d^{\text{test}}}{\partial \nu^4}(\nu) &= 12dr^2\lambda\mathrm{Tr}_d(K_I) > 0 \text{ when } r > 0\end{aligned}$$

Setting the derivative to 0: when $r = 0$, $\frac{\partial \tilde{\mathrm{R}}_d^{\text{test}}}{\partial \nu}(\nu) = -2\sigma^2\lambda\mathrm{Tr}_d(K_I^2)\nu \overset{\nu\to\infty}{\to} -\infty$ because $\mathrm{Tr}_d(K_I^2) > 0$ (by symmetry of K_I, this is the Frobenius norm squared of K_I which is nonzero), and $\nu = 0$ is actually a local (and global) maximum - so this is a case where arbitrarily growing ν is beneficial. This is no surprise since in that case:

$$\tilde{\mathrm{R}}_d^{\text{test}} \approx \sigma^2 + \sigma^2\mathrm{Tr}_d(K_I) - \sigma^2\lambda(1+\nu^2)\mathrm{Tr}_d(K_I^2)$$

For $r > 0$:

$$\begin{aligned}\frac{\partial \tilde{\mathrm{R}}_d^{\text{test}}}{\partial \nu}(\nu) = 0 &\iff \nu\left(2dr^2\lambda\mathrm{Tr}_d(K_I)\nu^2 + 2r^2\lambda\mathrm{Tr}_d(K_I) - \sigma^2\mathrm{Tr}_d(K_I^2)\right) = 0 \\ &\iff \nu = 0 \vee \nu^2 = \frac{\sigma^2\mathrm{Tr}_d(K_I^2)}{2dr^2\lambda\mathrm{Tr}_d(K_I)} - 1 \\ &\overset{\nu\geq 0}{\iff} \nu = 0 \vee \left(\nu = \sqrt{\frac{\sigma^2\mathrm{Tr}_d(K_I^2)}{2dr^2\lambda\mathrm{Tr}_d(K_I)} - 1} \wedge 2dr^2\lambda\mathrm{Tr}_d(K_I) < \sigma^2\mathrm{Tr}_d(K_I^2)\right)\end{aligned}$$

Three cases arise here. Precompute:

$$\tilde{\mathrm{R}}_d^{\text{test}}(\nu = 0) = \sigma^2 + \sigma^2\mathrm{Tr}_d(K_I) + \left[r^2\lambda^2 - \sigma^2\lambda\right]\mathrm{Tr}_d(K_I^2)$$

- $2dr^2\lambda\mathrm{Tr}_d(K_I) = \sigma^2\mathrm{Tr}_d(K_I^2) \implies \nu = 0$ is a local minimum because the fourth derivative is positive and all previous ones are zero. It is also the only stationary point, implying that in this case $\nu = 0$ is optimal globally.
- $2dr^2\lambda\mathrm{Tr}_d(K_I) > \sigma^2\mathrm{Tr}_d(K_I^2) \implies$
  - $\nu_1 = 0$ is a local minimum because the second derivative is positive.
  - $\nu_2 = \sqrt{\frac{\sigma^2\mathrm{Tr}_d(K_I^2)}{2dr^2\lambda\mathrm{Tr}_d(K_I)} - 1}$ is not well defined (and is not a stationary point).

  We collect that $2dr^2\lambda\mathrm{Tr}_d(K_I) \geq \sigma^2\mathrm{Tr}_d(K_I^2) \implies \nu = 0$ is optimal and no other value is.
- $2dr^2\lambda\mathrm{Tr}_d(K_I) < \sigma^2\mathrm{Tr}_d(K_I^2) \implies$
  - $\nu_1 = 0$ is actually a local maximum this time.

– $\nu_2 = \sqrt{\frac{\sigma^2 \mathrm{Tr}_d(K_I^2)}{2dr^2\lambda \mathrm{Tr}_d(K_I)} - 1}$ is stationary, and:

$$\begin{aligned}\frac{1}{2\lambda}\frac{\partial^2 \tilde{\mathrm{R}}_d^{\text{test}}}{\partial \nu^2}(\nu_2) &= 3 \times 2dr^2\lambda \mathrm{Tr}_d(K_I)\left(\frac{\sigma^2 \mathrm{Tr}_d(K_I^2)}{2dr^2\lambda \mathrm{Tr}_d(K_I)} - 1\right) - \sigma^2 \mathrm{Tr}_d(K_I^2) \\ &= 2\sigma^2 \mathrm{Tr}_d(K_I^2) - 3 \times 2dr^2\lambda \mathrm{Tr}_d(K_I)\end{aligned}$$

Once again, three cases arise.

* $2dr^2\lambda \mathrm{Tr}_d(K_I) < \frac{2}{3}\sigma^2 \mathrm{Tr}_d(K_I^2) \implies \nu_2$ is a local minimum as the second derivative is positive - the only one actually, thus ν_2 is globally optimal.
* $2dr^2\lambda \mathrm{Tr}_d(K_I) = \frac{2}{3}\sigma^2 \mathrm{Tr}_d(K_I^2) \implies \nu_2$ is a saddle point, and since 0 is a local maximum, $\nu_2 \to \infty$ is optimal.
* $2dr^2\lambda \mathrm{Tr}_d(K_I) > \frac{2}{3}\sigma^2 \mathrm{Tr}_d(K_I^2) \implies \nu_2$ is a local maximum as the second derivative is negative.

Summing up our findings in this section:

The test error can be approximated when d grows large almost surely as:

$$\tilde{\mathrm{R}}_d^{\text{test}} \overset{\text{a.s}}{=} \sigma^2 + \left[\sigma^2 + 2r^2\lambda^2\nu^2 + dr^2\lambda^2\nu^4\right] \mathrm{Tr}_d(K_I) + \left[r^2\lambda^2 - \sigma^2\lambda(1+\nu^2)\right] \mathrm{Tr}_d(K_I^2)$$

Also, when $0 \leq 2dr^2\lambda \mathrm{Tr}_d(K_I) < \frac{2}{3}\sigma^2 \mathrm{Tr}_d(K_I^2)$, picking $\nu^2 = \frac{\sigma^2 \mathrm{Tr}_d(K_I^2)}{2dr^2\lambda \mathrm{Tr}_d(K_I)} - 1$ minimizes the test error.

In short, there exists easily verifiable conditions where a nonzero variance perturbation achieves a better test error. Observe too that $0 \leq 2dr^2\lambda \mathrm{Tr}_d(K_I) < \frac{2}{3}\sigma^2 \mathrm{Tr}_d(K_I^2)$ holds when λ is small enough, independently of other problem parameters.

Further step?

It can be interesting to study how the findings of the previous section generalize, for $1 \leq k \leq d$:

$$\tilde{K} = I_d + \sum_{i=1}^{k} u_i u_i^T$$

Conclusion and outlook

This was a lovely project. I wanted a primer on research work, and I was very much served. The subject itself was wide enough to venture into many related subjects, going from stochastic gradient descent into linear regression somehow.

I cannot thank Antoine enough for his guidance and motivation throughout the semester, and Professor Macris for his trust and detailed explanations during the subject selection phase.

Bibliography

[1] Scott Pesme, Loucas Pillaud-Vivien, Nicolas Flammarion: Implicit Bias of SGD for Diagonal Linear Networks: a Provable Benefit of Stochasticity. https://arxiv.org/pdf/2106.09524

[2] Trevor Hastie, Andrea Montanari, Saharon Rosset, Ryan J. Tibshirani: Surprises in High-Dimensional Ridgeless Least Squares Interpolation https://arxiv.org/pdf/1903.08560.pdf

[3] Block matrix - Wikipedia article https://en.wikipedia.org/wiki/Block_matrix#:~:text=Block - the particular inversion formula comes from Bernstein, Dennis (2005): *Matrix Mathematics*, Princeton University Press, page 44

[4] Sherman, Jack; Morrison, Winifred J. (1949): *Adjustment of an Inverse Matrix Corresponding to a Change in One Element of a Given Matrix*, Annals of Mathematical Statistics, page 124