Journal of Machine Learning Research 24 (2023) 1-53 Submitted 2/22; Revised 9/23; Published 10/23

Elastic Gradient Descent, an Iterative Optimization Method Approximating the Solution Paths of the Elastic Net

Oskar Allerbo ALLERBO@CHALMERS.SE
Mathematical Sciences
University of Gothenburg and Chalmers University of Technology
SE-412 96 Gothenburg, Sweden

Johan Jonasson JONASSON@CHALMERS.SE
Mathematical Sciences
University of Gothenburg and Chalmers University of Technology
SE-412 96 Gothenburg, Sweden

Rebecka Jörnsten JORNSTEN@CHALMERS.SE
Mathematical Sciences
University of Gothenburg and Chalmers University of Technology
SE-412 96 Gothenburg, Sweden

Editor: Ryan Tibshirani

Abstract

The elastic net combines lasso and ridge regression to fuse the sparsity property of lasso with the grouping property of ridge regression. The connections between ridge regression and gradient descent and between lasso and forward stagewise regression have previously been shown. Similar to how the elastic net generalizes lasso and ridge regression, we introduce elastic gradient descent, a generalization of gradient descent and forward stagewise regression. We theoretically analyze elastic gradient descent and compare it to the elastic net and forward stagewise regression. Parts of the analysis are based on elastic gradient flow, a piecewise analytical construction, obtained for elastic gradient descent with infinitesimal step size. We also compare elastic gradient descent to the elastic net on real and simulated data and show that it provides similar solution paths, but is several orders of magnitude faster. Compared to forward stagewise regression, elastic gradient descent selects a model that, although still sparse, provides considerably lower prediction and estimation errors.

Keywords: elastic net, gradient descent, gradient flow, forward stagewise regression

1. Introduction

Lasso (Tibshirani, 1996) is a popular method for combining regularization and model selection in linear regression. The objective is to minimize

$$\frac{1}{2n}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1 \tag{1}$$

with respect to the parameter vector $\boldsymbol{\beta} \in \mathbb{R}^p$, where $\boldsymbol{X} \in \mathbb{R}^{n\times p}$ is the design matrix, $\boldsymbol{y} \in \mathbb{R}^n$ is the response vector, and $\lambda > 0$ is the regularization strength. Provided that the regularization is large enough, the lasso estimates of some parameters in $\boldsymbol{\beta}$ become exactly zero, thus eliminating the corresponding variables from the model, which results in a simpler

representation. Since the introduction of lasso, many extensions have been proposed, such as the adaptive lasso (Zou, 2006), with individual regularization strengths to each β_i; the group lasso (Yuan and Lin, 2006), which regularizes predefined groups of parameters together; the fused lasso (Tibshirani et al., 2005), which accounts for spatial and/or temporal dependencies; the graphical lasso (Friedman et al., 2008), for sparse inverse covariance estimation; and the elastic net (Zou and Hastie, 2005), which is a convex combination of lasso and ridge regression, generalizing Equation 1 into

$$\frac{1}{2n}\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\beta}\|_2^2+\lambda(\alpha\|\boldsymbol{\beta}\|_1+(1-\alpha)\|\boldsymbol{\beta}\|_2^2),\ \alpha\in[0,1]. \tag{2}$$

The motivation behind adding the squared ℓ_2 penalty of ridge regression to the elastic net is two-fold. First, in the high-dimensional setting, when $p>n$, lasso can select at most n variables. Second, if two or more variables are highly correlated, lasso tends to include only one of these in the model, and to be quite indifferent as to which. Both of these shortcomings are alleviated by the elastic net.

As can be seen in Equation 1, a larger value of λ enforces a smaller value of $\|\boldsymbol{\beta}\|_1$. Thus, provided $n\geq p$, the lasso estimate, $\hat{\boldsymbol{\beta}}$, shrinks (in ℓ_1 norm) with increasing λ from the ordinary least squares solution, $\hat{\boldsymbol{\beta}}^{\text{OLS}}:=(\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top\boldsymbol{y}$ for $\lambda=0$, to $\mathbf{0}$ for $\lambda\geq\lambda_{\max}:=\frac{1}{n}\|\boldsymbol{X}^\top\boldsymbol{y}\|_\infty$. Due to Lagrangian duality, the solution path of $\hat{\boldsymbol{\beta}}$ as a function of λ from 0 to $\lambda_{\max}$ can equivalently be expressed in terms of $\|\hat{\boldsymbol{\beta}}\|_1$, where $\lambda=0$ corresponds to $\|\hat{\boldsymbol{\beta}}\|_1=\|\hat{\boldsymbol{\beta}}^{\text{OLS}}\|_1$ and $\lambda=\lambda_{\max}$ corresponds to $\|\hat{\boldsymbol{\beta}}\|_1=0$.

Several authors have addressed the striking similarities between the lasso solution path and the solution path of forward stagewise linear regression (see e.g. work by Rosset et al. 2004, Efron et al. 2004 and Hastie et al. 2007). Forward stagewise regression is an iterative method for solving linear regression. Starting at $\hat{\boldsymbol{\beta}}=\mathbf{0}$, the solution moves toward $\hat{\boldsymbol{\beta}}^{\text{OLS}}$, successively adding more variables to the model, resulting in a solution path very similar to that of lasso. Selecting a solution before convergence, something that is often referred to as early stopping, can thus be thought of as applying lasso with a regularization strength $\lambda\in(0,\lambda_{\max})$. Tibshirani (2015) proposed a generalization of forward stagewise regression to be used with any convex function as opposed to just the ℓ_1 norm, and used it to obtain solution paths for group lasso, nuclear norm regularized matrix completion (e.g. Candès and Recht (2009)) and ridge logistic regression. Vaughan et al. (2017) used the general stagewise procedure to obtain solution paths for sparse group lasso (Simon et al., 2013), while Zhang (2019) used it for clustering.

Just as forward stagewise regression and lasso provide similar solution paths, so do gradient descent and ridge regression. Ali et al. (2019) investigated these similarities for infinitesimal optimization step size. They argued that, just as for forward stagewise regression, optimization time can be thought of as an inverted penalty, and that early stopping at time t roughly corresponds to ridge regression with penalty $1/t$.

In this paper, we combine forward stagewise regression and gradient descent into elastic gradient descent, an iterative optimization method that produces a solution path similar to that of the elastic net. Analogously to how the elastic net is a combination of lasso and ridge regression, elastic gradient descent is a combination of forward stagewise regression and gradient descent.

In Section 2, we introduce the elastic gradient descent algorithm. In Section 3, we theoretically analyze the algorithm, and compare it to the elastic net and to forward

stagewise regression. In Section 4, we compare elastic gradient descent to the elastic net and forward stagewise regression on synthetic and real data sets.

Our main contributions are:

- We define elastic gradient descent, an iterative optimization algorithm, that generalizes forward stagewise regression (also known as coordinate descent) and gradient descent, with solution paths very similar to those of the elastic net.
- We theoretically analyze the convergence properties of elastic gradient descent, the similarities and differences between elastic gradient descent with and without momentum, and the similarities and differences between elastic gradient descent and the elastic net and forward stagewise regression.
- We show on real and synthetic data that
  - compared to the elastic net, elastic gradient descent selects similar models, but is orders of magnitude faster.
  - compared to forward stagewise regression, elastic gradient descent is able to select a sparse model with considerably lower prediction and estimation errors.

All proofs are deferred to Appendix D.

2. Elastic Gradient Descent

Gradient descent is an iterative optimization method, where, in each time step, the solution is updated in the direction of the negative gradient. For the related method coordinate descent, each optimization step is constrained to update only one coordinate, namely the one with the largest absolute gradient value. For linear regression, coordinate descent and forward stagewise regression coincide, and thus we will henceforth use the name coordinate descent. For both coordinate and gradient descent, one optimization step can be expressed as

$$\hat{\boldsymbol{\beta}}(t + \Delta t) = \hat{\boldsymbol{\beta}}(t) - \Delta t \cdot \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}(t), \tag{3}$$

where $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}$ differs between the two algorithms.

We let $\boldsymbol{g}$ denote the gradient of the loss function, i.e. $\boldsymbol{g}(t) := \nabla_{\boldsymbol{\beta}(t)} L\left(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\beta}(t)\right)$. (For least squares, with $L\left(\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\beta}(t)\right) = \frac{1}{2n}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}(t)\|_2^2$, $\boldsymbol{g}(t) = -\frac{1}{n}\boldsymbol{X}^\top(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}(t))$.) For coordinate descent, $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{cd}}$ is defined according to

$$\begin{aligned} m(t) &:= \underset{d}{\operatorname{argmax}} \, |g_d(t)|, \\ \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{cd}}(t) &= \operatorname{sign}(g_m(t)) \cdot \boldsymbol{e_m}(t) = \boldsymbol{I}_{\text{cd}}(t) \cdot \operatorname{sign}(\boldsymbol{g}(t)), \end{aligned}$$

where $\boldsymbol{e_m}$ is the m-th standard basis vector, $\boldsymbol{I}_{\text{cd}}$ is a matrix of only zeros, except $(\boldsymbol{I}_{\text{cd}})_{mm}$ which is 1, and the sign of vector $\boldsymbol{g}$ is taken element-wise. Multiplying the matrix $\boldsymbol{I}_{\text{cd}}$ with the vector $\operatorname{sign}(\boldsymbol{g})$ we obtain a vector where all elements are zero, except element m which is exactly $\operatorname{sign}(g_m)$. For gradient descent,

$$\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{gd}}(t) = \boldsymbol{g}(t) = \boldsymbol{I}_{\text{gd}} \cdot \boldsymbol{g}(t),$$

where $\boldsymbol{I}_{\text{gd}} = \boldsymbol{I}$ is the identity matrix, which is included to emphasize the similarities to coordinate descent.

Naively combining coordinate and gradient descent with inspiration from the elastic net, Equation 2, suggests that

$$\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}}(\alpha, t) = \alpha \cdot \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{cd}}(t) + (1-\alpha) \cdot \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{gd}}(t) = \alpha \cdot \boldsymbol{I}_{\text{cd}}(t) \cdot \text{sign}(\boldsymbol{g}(t)) + (1-\alpha) \cdot \boldsymbol{I}_{\text{gd}} \cdot \boldsymbol{g}(t), \ \alpha \in [0, 1],$$

where coordinate and gradient descent are recovered as special cases for $\alpha = 1$ and $\alpha = 0$. However, this proposal does not share the desirable model selection property of the elastic net since $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{gd}}$ updates all parameters at all time steps, thus making all parameters non-zero already in the first time step. Therefore, we need a combination with the ability to keep some parameters fixed. Hence, we define

$$\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}}(\alpha, t) := \boldsymbol{I}_{\text{egd}}(\alpha, t) \cdot (\alpha \cdot \text{sign}(\boldsymbol{g}(t)) + (1-\alpha) \cdot \boldsymbol{g}(t)), \tag{4}$$

where $\boldsymbol{I}_{\text{egd}}$ is a diagonal matrix with zeros and ones on the diagonal, such that $\boldsymbol{I}_{\text{egd}}(0, t) = \boldsymbol{I}_{\text{gd}} = \boldsymbol{I}$ and $\boldsymbol{I}_{\text{egd}}(1, t) = \boldsymbol{I}_{\text{cd}}(t)$. $\boldsymbol{I}_{\text{egd}}$ could be defined in multiple ways. We, however, choose the following simple definition:

Definition 1 ($\boldsymbol{I}_{\text{egd}}$)**.**

$$\begin{aligned} &\textit{For } m(t) = \underset{d}{\text{argmax}}\, |g_d(t)|, \\ &\boldsymbol{I}_{\text{egd}}(\alpha, t)_{d_1 d_2} := \begin{cases} 1 \ \textit{if } d_1 = d_2 = d \ \textit{and } |g_d(t)| \geq \alpha \cdot |g_m(t)| \\ 0 \ \textit{else.} \end{cases} \end{aligned}$$

That is, for large gradient components, where "large" means "larger than α times the maximum component", the corresponding value in $\boldsymbol{I}_{\text{egd}}$ is 1, while for small components it is 0. Note that if $\alpha = 0$, all components are considered large, while for $\alpha = 1$ only the maximum component is. Our definition of large gradient components coincides with that by Friedman and Popescu (2004), but the update directions differ since we include the signed gradient in $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}}$. The reason for including the sign gradient is for elastic gradient descent to generalize coordinate descent, and thus to obtain a distinct connection to the elastic net.

Elastic gradient descent is summarized in Algorithm 1.

Algorithm 1 Elastic Gradient Descent

1: Initialize $\hat{\boldsymbol{\beta}} = \boldsymbol{0}$.
2: **repeat**
3: $\quad \boldsymbol{g} = \nabla_{\hat{\boldsymbol{\beta}}} L\left(\boldsymbol{X}, \boldsymbol{y}, \hat{\boldsymbol{\beta}}\right)$, where $L(\cdot)$ denotes the loss function.
4: $\quad \boldsymbol{I}_{\text{egd}} = \text{diag}\left(\mathbb{I}\left[\frac{|\boldsymbol{g}|}{\max_d |g_d|} \geq \alpha\right]\right)$, where $\mathbb{I}[\cdot]$ denotes the indicator function, which is taken element-wise, and where $\text{diag}(\cdot)$ creates a diagonal matrix from a vector.
5: $\quad \hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}} - \Delta t \cdot \boldsymbol{I}_{\text{egd}} \cdot (\alpha \cdot \text{sign}(\boldsymbol{g}) + (1-\alpha) \cdot \boldsymbol{g})$.
6: **until** convergence or other stopping criterion.

In Figure 1, we demonstrate the similarities between the solution paths of explicit regularization and iterative optimization. We compare the solution paths of ridge regression

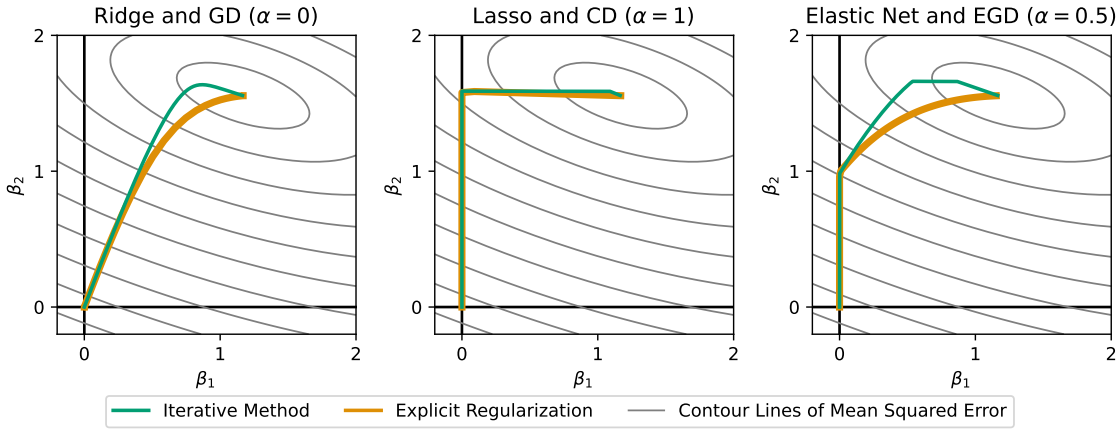

Figure 1: Solution paths of explicitly regularized and early stopping methods. In all three cases, the two methods follow similar, although not identical, solution paths.

and gradient descent (GD), lasso and coordinate descent (CD), and the elastic net and elastic gradient descent (EGD) for a simple linear model with two correlated parameters, β_1 and β_2. The solution paths of the corresponding algorithms are similar, although not identical.

Even though our definition of elastic gradient descent includes an element of arbitrariness, it proves to work well, as is shown in Section 4. In Appendix A, we investigate two slightly different definitions, based on the frameworks of steepest descent (Boyd and Vandenberghe, 2004) and the general stagewise procedure (Tibshirani, 2015). The three definitions provide virtually identical solutions.

2.1 Elastic Gradient Flow

Gradient descent with infinitesimal step size, Δt, is often referred to as gradient flow, which, since $\Delta t \to 0$, can be interpreted as a differential equation in training time, t. For some problems, including linear regression, this differential equation has a closed-form solution, which opens up for a better theoretical understanding of the algorithm. For elastic gradient descent, the corresponding differential equation becomes quite complicated. However, in Appendix C, we use it to construct something we refer to as elastic gradient flow, in analogy with gradient flow. Elastic gradient flow helps us to establish a theoretical connection between elastic gradient descent and the elastic net. To improve readability, in this section, we just state the equations of elastic gradient flow, and its special cases gradient flow and coordinate flow; for details, see Appendix C.

The elastic gradient flow estimate at time t is given by Equation 5,

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{\text{egf}}(t) =& \hat{\boldsymbol{\beta}}_{\text{egf}}(t_i) + \left((1-\alpha)\hat{\boldsymbol{\Sigma}}\right)^{-1}\left(\boldsymbol{I} - \exp\left(\boldsymbol{\Omega}^i(t_i,t)\right)\right) \\ & \cdot \left(\alpha \cdot \text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{egf}}(t_i))\right) + (1-\alpha)\cdot\hat{\boldsymbol{\Sigma}}\left(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{egf}}(t_i)\right)\right), \\ & t \in [t_i, t_{i+1}),\end{aligned} \tag{5}$$

where $\hat{\boldsymbol{\Sigma}} := \frac{1}{n}\boldsymbol{X}^\top\boldsymbol{X}$ is the empirical covariance matrix, $\boldsymbol{\Omega}^i(t_i,t)$ is the Magnus expansion (Magnus, 1954) of $-\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\boldsymbol{I}^i_{\text{egf}}(\alpha,t)$, $\{\boldsymbol{I}^i_{\text{egf}}\}_{i=0}^{i_{\max}}$ are the continuous-time versions of $\boldsymbol{I}_{\text{egd}}(\alpha,t)$ for $t\in[t_i,t_{i+1})$, and $\{t_i\}_{i=0}^{i_{\max}}$ are the times when parameters enter or leave the model.

The parameter $\gamma \in [0,1)$ is the strength of the momentum (Polyak, 1964), which is a generalization of gradient descent discussed in Section 3.6. For standard gradient descent without momentum, $\gamma = 0$.

For $\alpha = 0$, Equation 5 simplifies to gradient flow,

$$\hat{\boldsymbol{\beta}}_{\text{gf}}(t) = \left(\boldsymbol{I} - \exp\left(-\frac{t}{1-\gamma}\hat{\boldsymbol{\Sigma}}\right)\right)\hat{\boldsymbol{\beta}}^{\text{OLS}}, \tag{6}$$

and for $\alpha = 1$, it simplifies to what we refer to as coordinate flow,

$$\hat{\boldsymbol{\beta}}_{\text{cf}}(t) = \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i) + \frac{t-t_i}{1-\gamma}\boldsymbol{I}^i_{\text{cf}}\cdot\text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i))\right), \quad t\in[t_i,t_{i+1}), \tag{7}$$

where $\{\boldsymbol{I}^i_{\text{cf}}\}_{i=0}^{i_{\max}}$ are continuous-time versions of $\boldsymbol{I}_{\text{cd}}(t)$ for $t\in[t_i,t_{i+1})$.

3. Properties of Elastic Gradient Descent

In this section, we theoretically investigate elastic gradient descent and flow and make comparisons to the elastic net and coordinate descent, assessing the similarities and differences.

3.1 Convergence of Elastic Gradient Descent

For a small enough step size, Δt, elastic gradient descent always moves downhill in the optimization landscape. In Proposition 2, we present bounds for the step size that guarantee an improvement when applying elastic gradient descent to a strongly convex problem.

Proposition 2.
Assume that the loss function, $L(\boldsymbol{\beta}) = L(\boldsymbol{X},\boldsymbol{y},\boldsymbol{\beta})$, is strongly convex with Hessian bounded according to $\nabla^2 L(\boldsymbol{\beta}) \preceq M\boldsymbol{I}$ (i.e. $M\boldsymbol{I} - \nabla^2 L(\boldsymbol{\beta})$ is a positive semi-definite matrix), for some $M > 0$. Denote

$$\boldsymbol{g} := \nabla L(\boldsymbol{\beta}),\ g_{\max} = g_m := \max_d |g_d| \text{ and } g_{\min} := \min_{\substack{d:\ |g_d|\geq\alpha,\\ g_d\neq 0}} |g_d|.$$

Then,

$$\Delta t < \frac{2}{M}\cdot g_{\max}\cdot\frac{\alpha+(1-\alpha)g_{\max}}{(1_{\alpha>0}+(1-\alpha)g_{\max})^2} \tag{8a}$$

$$\implies \Delta t < \frac{2}{M}\cdot\frac{g^2_{\min}}{g_{\max}}\cdot\frac{\alpha+(1-\alpha)g_{\max}}{(\alpha+(1-\alpha)g_{\min})^2} \tag{8b}$$

$$\implies L(\hat{\boldsymbol{\beta}} - \Delta t\cdot\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}}) - L(\hat{\boldsymbol{\beta}}) \leq 0,$$

where

$$1_{\alpha>0} = \begin{cases} 1 \text{ if } \alpha > 0 \\ 0 \text{ if } \alpha = 0. \end{cases}$$

Remark 1: The bound in 8b allows for a greater value of Δt than that in Equation 8a, but requires knowledge of the minimum gradient value in addition to the maximum gradient value.
Remark 2: For $\alpha = 0$ and $\alpha = 1$, the bounds for Δt become $\frac{2}{M}$ and $\frac{2g_{\max}}{M}$ respectively. Note than for $\alpha = 1$, $g_{\min} = g_{\max}$.
Remark 3: If a fixed value is used for Δt, once $g_{\max}$ gets small enough, the loss function is not guaranteed to decrease, unless $\alpha = 0$. In this case, training should be interrupted when the solution starts to worsen.
Remark 4: For linear regression, M is the maximum eigenvalue of the empirical covariance matrix, $\hat{\boldsymbol{\Sigma}} = \frac{1}{n}\boldsymbol{X}^\top \boldsymbol{X}$.

3.2 Calculating Solution Paths

In the original implementation of the elastic net, ridge regression in the penalized version,

$$\min_{\boldsymbol{\beta}} \frac{1}{2n}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda_2\|\boldsymbol{\beta}\|_2^2,$$

is combined with the LARS algorithm (Efron et al., 2004), which solves the constrained version of the lasso problem,

$$\min_{\boldsymbol{\beta}} \frac{1}{2n}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2, \text{ s.t. } \|\boldsymbol{\beta}\|_1 \leq R_1,$$

returning the entire solution path as a function of R_1. This implies that the elastic net problem is formulated as

$$\min_{\boldsymbol{\beta}} \frac{1}{2n}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda_2\|\boldsymbol{\beta}\|_2^2, \text{ s.t. } \|\boldsymbol{\beta}\|_1 \leq R_1$$

and that call to the algorithm returns the solution path for $\hat{\boldsymbol{\beta}}$ for different values of R_1, with a fixed value of λ_2. Thus, each solution corresponds to a combination (R_1, λ_2), rather than the more intuitive combination (α, λ). Later versions, including those by Friedman et al. (2010), use iterative methods to obtain solutions expressed as combinations of (α, λ), where the solution for a given λ is calculated independently of the others by running an iterative algorithm to convergence.

Elastic gradient descent is also an iterative algorithm, but here the solution at each iteration is of interest by itself and corresponds to a combination (α, t). Running the algorithm to convergence once returns all values of t between 0 and $t_{\max}$. In contrast, the elastic net algorithm has to be run to convergence once for every value of λ.

Comparing elastic gradient descent to coordinate descent, while there is no restriction on the number of parameters elastic gradient descent can update in each iteration, coordinate descent always only updates one parameter per iteration. Thus, especially for problems with many dimensions, elastic gradient descent has a computational advantage compared to coordinate descent.

In Section 4, we verify the faster computational speed of elastic gradient descent compared to those of the elastic net and coordinate descent.

3.3 Differences in the Solution Paths

For $\alpha > 0$, both the elastic net and elastic gradient descent tend to set some parameters to 0, but this is done using two different techniques. The elastic net has no closed-form solution, unless for isotropic features, i.e. $\boldsymbol{\Sigma} = \boldsymbol{I}$, for which the solution is given by

$$\hat{\beta}_d^{\text{en}}(\lambda) = \frac{\text{sign}(\hat{\beta}_d^{\text{OLS}}) \cdot \max(0, |\hat{\beta}_d^{\text{OLS}}| - \alpha\lambda)}{1 + (1-\alpha)\lambda}. \tag{9}$$

Consider the numerator of Equation 9. Compared to the ordinary least squares solution, each $\hat{\beta}_d^{\text{en}}$ is translated toward zero, and once it changes sign it is set to exactly zero, i.e. the elastic net shifts all paths toward 0. Elastic gradient descent, in contrast, by Definition 1 stops updating a parameter when the corresponding gradient value is small. If this occurs when the parameter value is 0, the value will constantly remain so, but it might also stay constant at some other level. This is illustrated in Figures 1, 2 and 3.

3.4 Susceptibility to Correlations

The ridge estimate is usually written as $\hat{\boldsymbol{\beta}}(\lambda) := (\boldsymbol{X}^\top \boldsymbol{X} + n\lambda \boldsymbol{I})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$, but according to Lemma 3 it can be reformulated in a way that resembles the gradient flow estimate.

Lemma 3.
With $\hat{\boldsymbol{\Sigma}} := \frac{1}{n} \boldsymbol{X}^\top \boldsymbol{X}$ and $\hat{\boldsymbol{\beta}}^{\text{OLS}} := (\boldsymbol{X}^\top \boldsymbol{X})^+ \boldsymbol{X}^\top \boldsymbol{y}$, where $(\cdot)^+$ denotes the Moore-Penrose pseudoinverse, the ridge estimate can be written as

$$\hat{\boldsymbol{\beta}}(\lambda) = \left(\boldsymbol{I} - \left(\boldsymbol{I} + \frac{1}{\lambda} \hat{\boldsymbol{\Sigma}} \right)^{-1} \right) \hat{\boldsymbol{\beta}}^{\text{OLS}}. \tag{10}$$

Comparing Equation 10 to Equation 6,

$$\hat{\boldsymbol{\beta}}_{\text{gf}}(t) = \left(\boldsymbol{I} - \exp\left(-\frac{t}{1-\gamma} \hat{\boldsymbol{\Sigma}} \right) \right) \hat{\boldsymbol{\beta}}^{\text{OLS}} = \left(\boldsymbol{I} - \exp\left(\frac{t}{1-\gamma} \hat{\boldsymbol{\Sigma}} \right)^{-1} \right) \hat{\boldsymbol{\beta}}^{\text{OLS}},$$

we see that if we define $\lambda := (1-\gamma)/t$, the ridge estimate can be thought of as a first-order Taylor approximation of the gradient flow estimate. The fact that the ridge estimate depends linearly on $\hat{\boldsymbol{\Sigma}}$, whereas the gradient flow estimate depends exponentially, suggests that elastic gradient descent takes correlations into larger consideration than the elastic net does, with an even stronger tendency to, for standardized data, assign similar parameter values to correlated variables. This is further illustrated in Section 4.

3.5 The Connection between λ and t

As stated above, lasso and the elastic net have no closed-form solutions, unless for isotropic features, where the elastic net solution is given by Equation 9, which for lasso ($\alpha = 1$) simplifies to

$$\hat{\beta}_d^{\text{lasso}}(\lambda) = \text{sign}(\hat{\beta}_d^{\text{OLS}}) \cdot \max(0, |\hat{\beta}_d^{\text{OLS}}| - \lambda). \tag{11}$$

In Proposition 4 we investigate the connection between Equations 9 (elastic net with isotropic features) and 5 (elastic gradient flow), and, as a special case, between Equations 11 (lasso with

isotropic features) and 7 (coordinate flow) when $\hat{\boldsymbol{\Sigma}} = \boldsymbol{I}$ by requiring $\hat{\beta}_d^{\text{en}}(\lambda) = (\hat{\boldsymbol{\beta}}_{\text{egf}}(t))_d =: \hat{\beta}_d^{\text{egf}}(t)$.

Proposition 4.
Solving $\hat{\beta}_d^{\text{en}}(\lambda) = \hat{\beta}_d^{\text{egf}}(t) := (\hat{\boldsymbol{\beta}}_{\text{egf}}(t))_d$ for $\hat{\boldsymbol{\Sigma}} = \boldsymbol{I}$, with $\hat{\boldsymbol{\beta}}_{\text{egf}}(t)$ according to Equation 5 and $\hat{\beta}_d^{\text{en}}(\lambda)$ according to Equation 9, we obtain

$$\lambda_d = \max\left(\frac{|\hat{\beta}_d^{\text{OLS}}| - |\hat{\beta}_d^{\text{egf}}(t_i)| - \boldsymbol{v}}{\alpha + (1-\alpha)\left(|\hat{\beta}_d^{\text{egf}}(t_i)| + \boldsymbol{v}\right)}, 0\right), \tag{12}$$

where

$$\boldsymbol{v} = \frac{1}{1-\alpha}\left(1 - \exp\left(-\frac{1-\alpha}{1-\gamma}\int_{t_i}^{t} (\boldsymbol{I}_{\text{egf}}^i)_{dd}(\alpha,\tau)d\tau\right)\right)\left(\alpha + (1-\alpha)\left(|\hat{\beta}_d^{\text{OLS}}| - |\hat{\beta}_d^{\text{egf}}(t_i)|\right)\right),$$

which implies $\frac{\partial \lambda_d(t)}{\partial t} \leq 0$.

For $\alpha = 1$, Equation 12 reduces to

$$\lambda_d = \max\left(|\hat{\beta}_d^{\text{OLS}}| - |\hat{\beta}_d^{\text{cf}}(t_i)| - \frac{t - t_i}{1-\gamma}(\boldsymbol{I}_{\text{cf}}^i)_{dd}, 0\right).$$

We note that while for gradient flow, the relationship between λ and $t/(1-\gamma)$ is approximately the multiplicative inverse, $\lambda \approx (1-\gamma)/t$, for coordinate flow it is approximately the additive inverse, $\lambda \approx -t/(1-\gamma)$. For the elastic net, it is something in between. Furthermore, for the elastic net, the relationship between λ and t depends on $\int_{t_i}^{t} (\boldsymbol{I}_{\text{egf}}^i)_{dd}(\alpha,\tau)d\tau$. When this integral is close to zero for a parameter, the relationship between λ_d and t becomes almost linear, while for a larger value, the relation becomes almost exponential. Since the value of the integral may vary with d, for a given value of α the relation between λ_d and t might be almost linear for some parameters and almost exponential for others. Furthermore, since $\boldsymbol{I}_{\text{egf}}^i$ is recalculated at times t_i, for some time t_i the relation might change between almost linear and almost exponential for a parameter.

In summary, Proposition 4 reveals that, while always decreasing with optimization time, the rate of the decrease of the regularization might vary substantially, both between parameters and during optimization.

3.6 The Effect of Momentum

Momentum (Polyak, 1964) is a way to introduce memory into gradient-based optimization methods. The idea is to increase the computational stability and speed by allowing not only for current, but also for past, gradient values to influence the update direction, analogous to how a ball rolls down a slope: with increased momentum (and speed), it does not respond immediately to changes in the slope. Introducing momentum, Equation 3 generalizes into

$$\hat{\boldsymbol{\beta}}(t + \Delta t) = \hat{\boldsymbol{\beta}}(t) + \gamma\left(\hat{\boldsymbol{\beta}}(t) - \hat{\boldsymbol{\beta}}(t - \Delta t)\right) - \Delta t \cdot \boldsymbol{g}(t),$$

where $\gamma \in [0, 1)$ is the strength of the momentum.

For elastic gradient flow, and its special cases gradient and coordinate flow, $\gamma > 0$ has the effect of rescaling the gradient selection matrix. $\boldsymbol{I}_{\text{gf}}$ is replaced by $\frac{1}{1-\gamma}\boldsymbol{I}_{\text{gf}}$ (where $\boldsymbol{I}_{\text{gf}} = \boldsymbol{I}$), $\boldsymbol{I}_{\text{cf}}$ by $\frac{1}{1-\gamma}\boldsymbol{I}_{\text{cf}}$, and $\boldsymbol{I}_{\text{egf}}$ by $\frac{1}{1-\gamma}\boldsymbol{I}_{\text{egf}}$. Thus, for (very) small step sizes, momentum does not affect the solution path, it just increases the speed.

For larger step sizes, the addition of momentum may, in addition to increasing the computational speed, change the solution path. With momentum, the gradient values at the beginning of the training contribute more to the solution, than without it. For elastic gradient descent, in the early stages of training many parameters have small gradient values and are not yet included in the model. This suggests that elastic gradient descent with momentum would promote sparser models, compared to elastic gradient descent without momentum, something that is supported by the experiments in Section 4 and Appendix B.

4. Experiments

In this section, we compare elastic gradient descent with and without momentum to the elastic net and coordinate descent on twelve different data sets. In order to illustrate the path differences between elastic gradient descent and the elastic net as discussed in Section 3.3, we use a very simple data set with only three variables, and the diabetes data set used by Efron et al. (2004).[1] We then compare model selection accuracy and performance on a synthetic data set consisting of two blocks of parameters, where one block is included in the true model, and the other is not, for different correlations. Finally, we compare the performance of the algorithms on nine relatively large real data sets.

For elastic gradient and coordinate descent, a step size of 0.01 was used in all experiments except for the first, simple experiment, where 0.001 was used. We stopped the training when the training error no longer decreased, which, for $\alpha > 0$, eventually happens according to Proposition 2. For elastic gradient descent with momentum, we consistently used $\gamma = 0.5$. For the elastic net the `enet_path` method in the Scikit-learn library (Pedregosa et al., 2011) was used. All experiments, except those in Section 4.3 (and the corresponding additional experiments in Appendix B), were run in Python on a Dell Latitude 7480 laptop, with an Intel Core i7, 2.80 GHz processor with four kernels. The experiments in Section 4.3, and the corresponding experiments in the appendix, were run on a cluster with Intel Xeon Gold 6130, 2.10 GHz processors.

4.1 Solution Paths for Simple Synthetic Data

To illustrate the different path properties of elastic gradient descent and the elastic net, 1000 observations were generated according to

$$\boldsymbol{X} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} 1 & 0.7 & 0.7 \\ 0.7 & 1 & 0.7 \\ 0.7 & 0.7 & 1 \end{bmatrix}\right), \; \boldsymbol{y} = \boldsymbol{X}\begin{bmatrix} 1 \\ 0.1 \\ 0 \end{bmatrix}.$$

The solution paths for four different values of α are shown in Figure 2. For $\alpha = 0$, due to the correlations in the data, initially, all parameter estimates aim toward values somewhere between 0 and 1. As t increases (λ decreases), the estimates start approaching their true

1. The data set is available at `https://web.stanford.edu/~hastie/Papers/LARS/diabetes.data`.

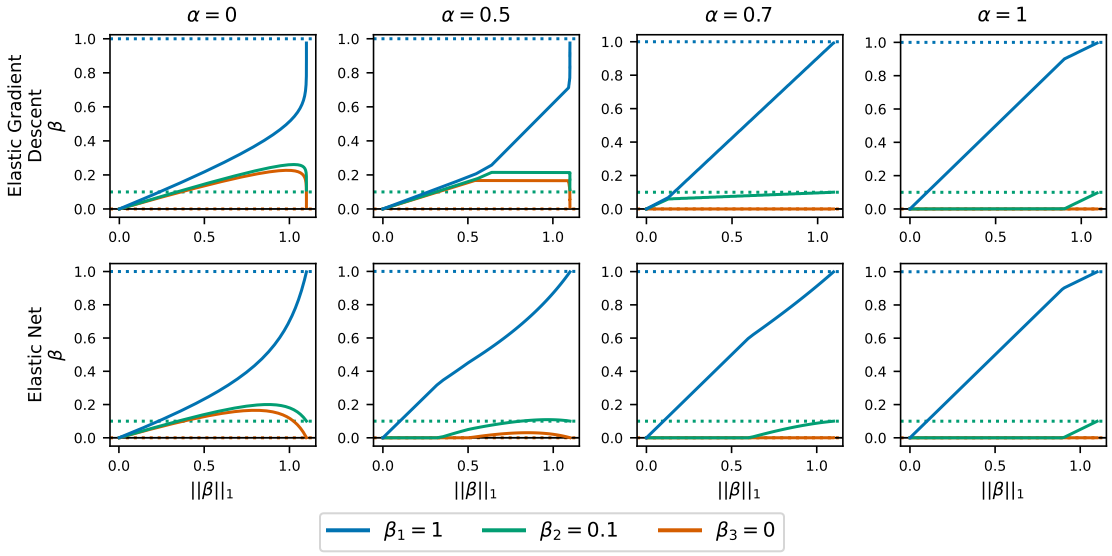

Figure 2: Comparison between elastic gradient descent (without momentum) and the elastic net on highly correlated data. In both cases, the drift toward 1 of the second parameter reduces with increasing α, but while elastic gradient descent cuts the peak from above, the elastic net moves the entire path downward. In contrast to the elastic net, elastic gradient descent correctly includes β_2 for the whole solution paths when $\alpha = 0.5$ and $\alpha = 0.7$, but erroneously includes β_3 for a larger fraction of the solution path for $\alpha = 0.5$.

values. Elastic gradient descent is more affected by the correlations, i.e. the parameter estimates move together for a larger fraction of the solution path, than the elastic net is, which is in line with the observations in Section 3.4, and also tends to affect the model selection properties. While elastic gradient descent includes the true positive β_2 for the entire solution paths for $\alpha = 0.5$ and $\alpha = 0.7$, this is not the case for the elastic net. On the other hand, for $\alpha = 0.5$, elastic gradient descent erroneously includes true negative β_3 for a larger fraction of the solution path than the elastic net does. As α increases, in both cases, the maximum values of the paths β_2 and β_3 are reduced, but while elastic gradient descent "cuts the peak" from above, the elastic net translates the entire path downward. The "peak cutting" behavior of elastic gradient descent comes from the fact that the gradient is the smallest just before changing sign, at the top of the peak.

4.2 Solution Paths for the Diabetes Data

The diabetes data set contains 442 observations, each consisting of 1 target value, which measures disease progression, and the 10 covariates `age`, `sex`, `bmi` (body mass index), `bp` (average blood pressure), `tc` (t-cells), `ld` (low-density lipoproteins), `hdl` (high-density lipoproteins), `tch` (thyroid stimulating hormone), `ltg` (lamotrigine) and `glu` (blood sugar level).

In Figure 3, we show the solution paths of elastic gradient descent, without momentum, and the elastic net for two different values of α. Similar to in Figure 2, elastic gradient descent cuts peaks from above, while the elastic net translates them toward zero. It can be seen how this difference makes the algorithms behave differently for small values of $\|\boldsymbol{\beta}\|_1$. While elastic gradient descent tends to include a subset of the parameters in the model immediately, the inclusion of the same set is more spread out for the elastic net. This contributes to elastic gradient descent proposing fewer models along the solution path than the elastic net does. Excluding the empty model, elastic gradient descent proposes 3 different models for $\alpha = 0.3$ and 7 models for $\alpha = 0.7$. The corresponding numbers for the elastic net are 10 and 11. If it were to be taken into account that the elastic net proposes the same model at different, non-adjacent sections along the path, its numbers would be even higher. This suggests that in terms of model selection, elastic gradient descent is more robust with respect to the degree of penalization than the elastic net is.

In Figure 4, we compare the solution paths and the normalized gradients for elastic gradient descent with $\alpha = 0.5$, coordinate descent, and the elastic net. Note that the bottom right panel does not show the gradients of the elastic net, since there are none, but instead the gradients of the elastic gradient flow solution. Compared to coordinate descent, elastic gradient descent includes more parameters earlier, which is in line with the motivation behind the elastic net to include correlated covariates together. Studying the gradients, it can be seen how the parameters are split into three sets, which we refer to as the free, coupled, and inactive sets, see Appendix C for details. The free parameters all have normalized gradient values, $|g_d|/\|\boldsymbol{g}\|_\infty$, larger than α, and are updated freely. This group includes the maximum gradient parameter with $|g_m|/\|\boldsymbol{g}\|_\infty = 1$. For the inactive parameters, $|g_d|/\|\boldsymbol{g}\|_\infty < \alpha$ and these parameters are not updated as can be seen in the first column. For the coupled parameters, $|g_d|/\|\boldsymbol{g}\|_\infty$ oscillates around (for the descent algorithms) or equals (for the flow algorithm) α. The coupled parameters are still updated but at a slower pace than the free ones. For coordinate descent, there are no free parameters, only coupled and inactive. Toward the end of the training, when $\|\boldsymbol{g}\|_\infty$ is small, we see oscillations in the gradients for coordinate descent and elastic gradient descent, which is in line with the conclusions from Proposition 2.

4.3 Synthetic Data for Model Selection

To compare model selection and performance, the following synthetic data set was created: The variables were split into two blocks of equal length, where the first block was included in the true model, and the second was not. The parameter values of the true positive variables were normally distributed with mean 2 and variance 1, the correlations within the two blocks were set to ρ_1, and between the two blocks to ρ_2:

$$
\begin{aligned}
\boldsymbol{\beta^*} &= \begin{bmatrix} \mathcal{N}(2,1)_{p/2}^\top & \mathbf{0}_{p/2}^\top \end{bmatrix}^\top \\
\mathbf{\Sigma_{11}} = \mathbf{\Sigma_{22}} &= \rho_1 \cdot (\mathbf{1}\mathbf{1}^\top)_{p/2\times p/2} + (1-\rho_1)\cdot \boldsymbol{I}_{p/2\times p/2} \\
\mathbf{\Sigma_{12}} = \mathbf{\Sigma_{12}^\top} &= \rho_2 \cdot (\mathbf{1}\mathbf{1}^\top)_{p/2\times p/2} \\
\mathbf{\Sigma} &= \begin{bmatrix} \mathbf{\Sigma_{11}} & \mathbf{\Sigma_{12}} \\ \mathbf{\Sigma_{12}^\top} & \mathbf{\Sigma_{11}} \end{bmatrix},
\end{aligned}
$$

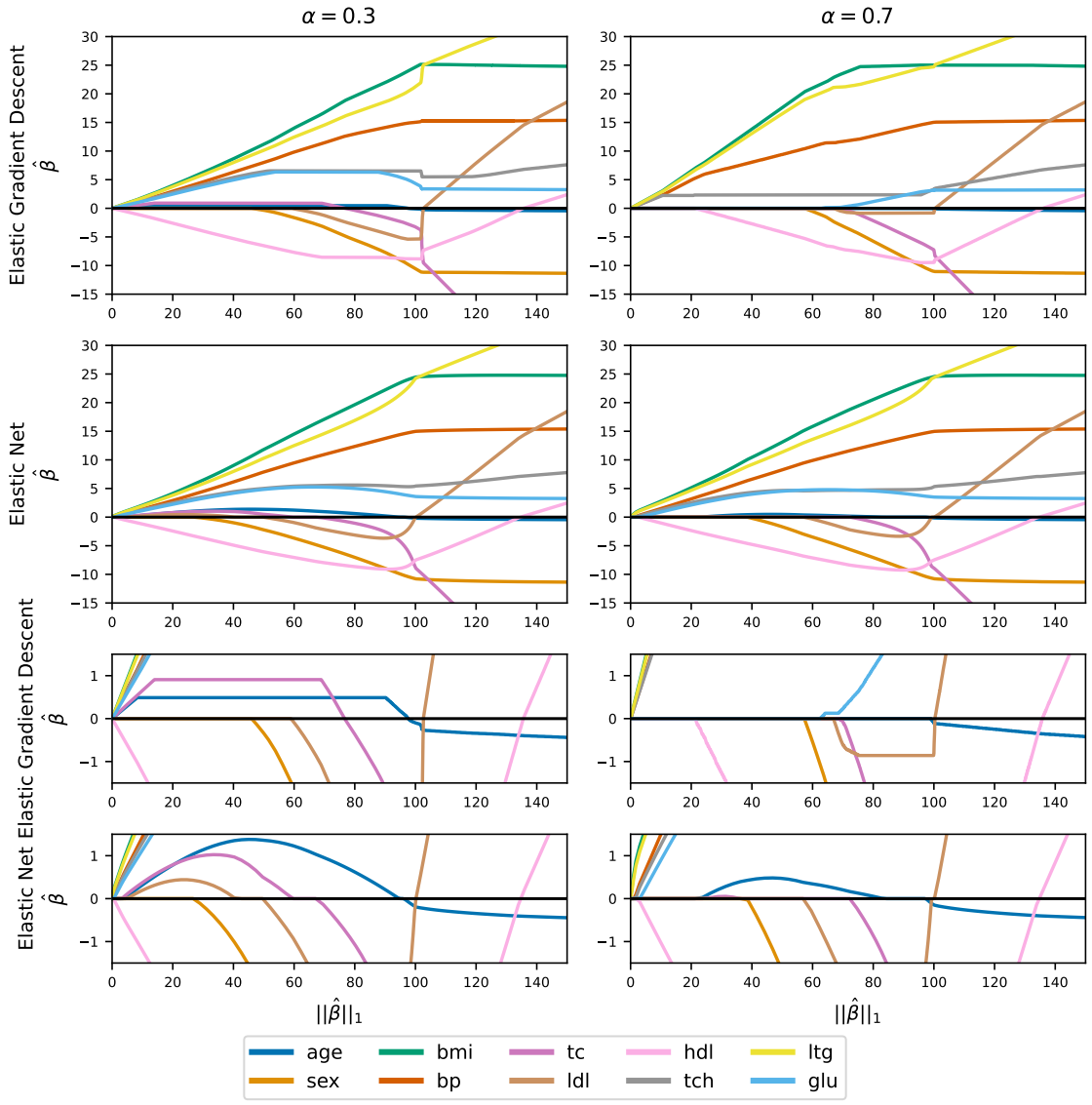

Figure 3: Solution paths for elastic gradient descent (without momentum) and the elastic net on the diabetes data. Rows three and four show the same things as rows one and two but on different y-scales. While elastic gradient descent cuts peaks from above, the elastic net translates them toward zero. Elastic gradient descent is more robust in terms of model selection with respect to the degree of penalization, proposing fewer different models along the solution path than the elastic net does.

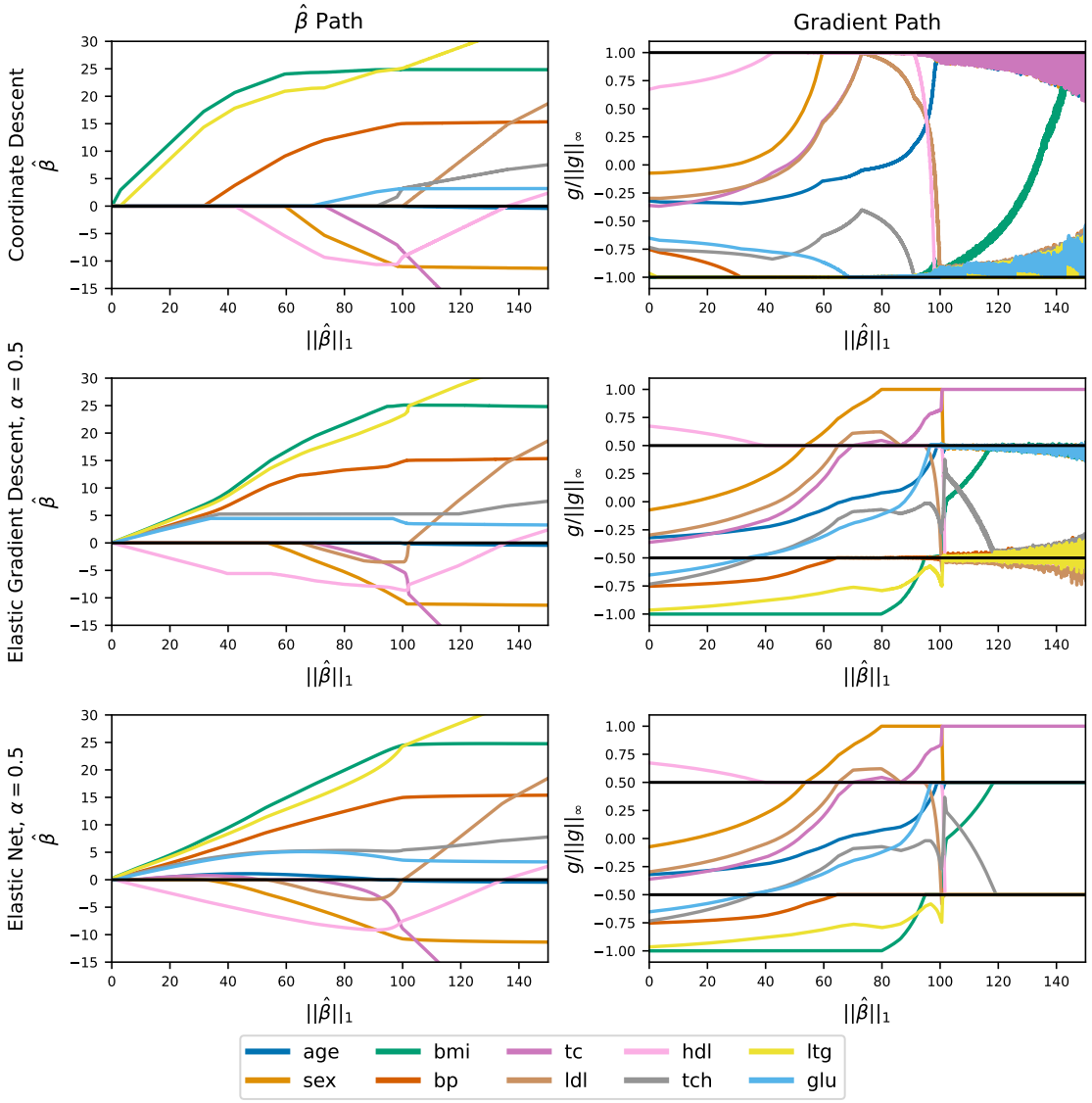

Figure 4: Solution paths and normalized gradients for coordinate descent, elastic gradient descent (without momentum), and the elastic net on the diabetes data. Note that the bottom right pane shows the gradients of elastic gradient flow. We see how the parameters are split into the free, coupled, and inactive sets. Depending on whether $|g_d|$ is greater than, equal to, or smaller than $\alpha \cdot \|\boldsymbol{g}\|_\infty$, the parameters update either freely, in a coupled fashion, or not at all, respectively. For instance, the `tch` parameter initially has an absolute normalized gradient value larger than $\alpha = 0.5$ and updates freely. At $\|\hat{\boldsymbol{\beta}}\|_1 \approx 40$, the absolute normalized gradient becomes less than α and the parameter is not updated at all until $\|\hat{\boldsymbol{\beta}}\|_1 \approx 120$. Then it starts updating in a coupled fashion.

where $\boldsymbol{I}_{p/2\times p/2}$ denotes the $(p/2)\times(p/2)$ identity matrix, $(\mathbf{1}\mathbf{1}^\top)_{p/2\times p/2}$ denotes a $(p/2)\times(p/2)$ matrix of only ones, $\mathcal{N}(\cdot,\cdot)_{p/2}$ denotes an i.i.d. vector of length $p/2$ and $\mathbf{0}_{p/2}$ a vector of length $p/2$ with all zeros. For $n = 100/30/30$, where the three values of n denote training, validation and testing sets, n observations were sampled according to

$$\boldsymbol{X} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}),\ \boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta}^* + \mathcal{N}(\mathbf{0}, \sigma^2\boldsymbol{I}),$$

for $\sigma = 10$, $\rho_1 = 0.7$, $\rho_2 = 0.3$ and $p \in [50, 60, \ldots, 200]$. For each value of p, the experiment was repeated 5001 times for different data realizations. For elastic gradient descent and the elastic net, nine different values of α were considered, $\alpha \in \{0.1,\ 0.2, \ldots\ 0.9\}$, and the combination of $(\alpha, t/\lambda)$ with the lowest mean squared error, MSE, on validation data was selected. For coordinate descent, where always $\alpha = 1$, t was selected by validation MSE.

The following test statistics were computed and compared between the three models:

- Sensitivity (true positive rate).
- Specificity (true negative rate).
- Estimation error, $\frac{1}{n^*}\|\boldsymbol{X}^*\hat{\boldsymbol{\beta}} - \boldsymbol{X}^*\boldsymbol{\beta}^*\|_2$, where $\boldsymbol{X}^* \in \mathbb{R}^{n^*\times p}$ is previously unseen data.
- Prediction error, $\frac{1}{p}\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|_2$.
- Execution time in seconds.

Figure 5 shows the median values together with the first and third quartiles across the 5001 realizations, for the different test statistics.

Elastic gradient descent and the elastic net perform similarly in all aspects except computational time, where elastic gradient descent performs significantly faster. The computational performance of elastic gradient descent improves with momentum. For high-dimensional data ($p > n = 100$), where no unique solution exists, momentum also greatly improves the model specificity. These two results are in line with the discussion in Section 3.6, according to which momentum increases the computational speed and promotes a sparser solution. The elastic net is more stable than elastic gradient descent in terms of specificity, at least in the absence of momentum, where elastic gradient descent, although performing well in general, sometimes includes all true negatives. In Appendix B, we further examine the specificity properties by varying the experiment so that the number of non-zero parameters is constant when p increases.

Compared to coordinate descent, elastic gradient descent performs better in all aspects except for specificity. The higher specificity of coordinate descent, however, comes at the cost of much worse sensitivity, and prediction and estimation errors. The execution times of elastic gradient descent and the elastic net include testing for nine different values of α, while for coordinate descent only one value of α is considered. Still, coordinated descent requires more computational time than elastic gradient descent. This can be attributed to the fact that coordinate descent updates only one parameter per iteration, something that becomes more apparent when p is large.

The computational time of elastic gradient descent is less affected by the dimensionality than those of the elastic net and coordinate descent. Since the elastic net and coordinate descent algorithms only update one parameter per iteration, the dimensionality has quite a

Data set	Size, $n \times p$
Quality of aspen tree fibres[2]	25165×5
House values in California (Pace and Barry, 1997)[3]	20640×8
Daily concentration of black smoke particles in the U.K. in the year 2000 (Wood et al., 2017)[4]	45568×10
Results of the 2019 Portuguese Parliamentary Elections[5]	21643×18
Appliances energy use in a low energy building in Stambruges, Belgium (Candanedo et al., 2017)[6]	19735×27
Protein structure as root-mean-square deviation of atomic positions, taken from CASP[7]	45730×9
Critical temperature of superconductors[8]	21263×81
Readability of texts used in English education[9]	2834×768
Topic popularity on Twitter (Kawala et al., 2016)[10]	291624×77

Table 1: Real data sets used for comparing elastic gradient descent to the elastic net and coordinate descent.

large impact on the execution time of these algorithms. On the other hand, elastic gradient descent may update multiple parameters per iteration, and the execution time is thus less affected by the dimensionality.

In Appendix B, we extend the simulation, presenting results for all combinations of $\rho_1 \in [0.5,\ 0.6,\ 0.7,\ 0.8,\ 0.9,]$, $\rho_2 \in [0.0,\ 0.1,\ 0.2,\ 0.3, 0.4,\ 0.5]$ and $p \in [50, 100, 200]$. The conclusions are consistent with the ones presented here.

4.4 Computational Efficiently on Real Data sets

In this section, we compare elastic gradient descent with and without momentum to the elastic net and coordinate descent on the nine real data sets described in Table 1. The data sets were selected to compare the algorithms on a diverse set of applications, although they all have in common that they are relatively large in terms of number of observations and/or dimensions. The data was split 80%/10%/10% into training, validation, and testing data for 5001 random splits. For elastic gradient descent and the elastic net, nine different values of α were considered, $\alpha \in \{0.1,\ 0.2, \dots\ 0.9\}$, and the combination of $(\alpha, t/\lambda)$ with the lowest mean squared error, MSE, on validation data was selected. For coordinate descent, where always $\alpha = 1$, t was selected by validation MSE.

2. The data set is available at `https://openmv.net/info/wood-fibres`.
3. The data set is available at `https://www.dcc.fc.up.pt/~ltorgo/Regression/cal_housing.html`.
4. The data set is available at `https://www.maths.ed.ac.uk/~swood34`.
5. The data set is available at `https://archive.ics.uci.edu/dataset/513/real+time+election+results+portugal+2019`.
6. The data set is available at `https://github.com/LuisM78/Appliances-energy-prediction-data`.
7. `https://predictioncenter.org/`, The data set is available at `https://archive.ics.uci.edu/ml/datasets/Physicochemical+Properties+of+Protein+Tertiary+Structure`.
8. The data set is available at `https://archive.ics.uci.edu/dataset/464/superconductivty+data`.
9. The data set is available at `https://www.kaggle.com/code/uocoeeds/building-a-regression-model-with-elastic-net/input`.
10. The data set is available at `http://archive.ics.uci.edu/dataset/248/buzz+in+social+media`.

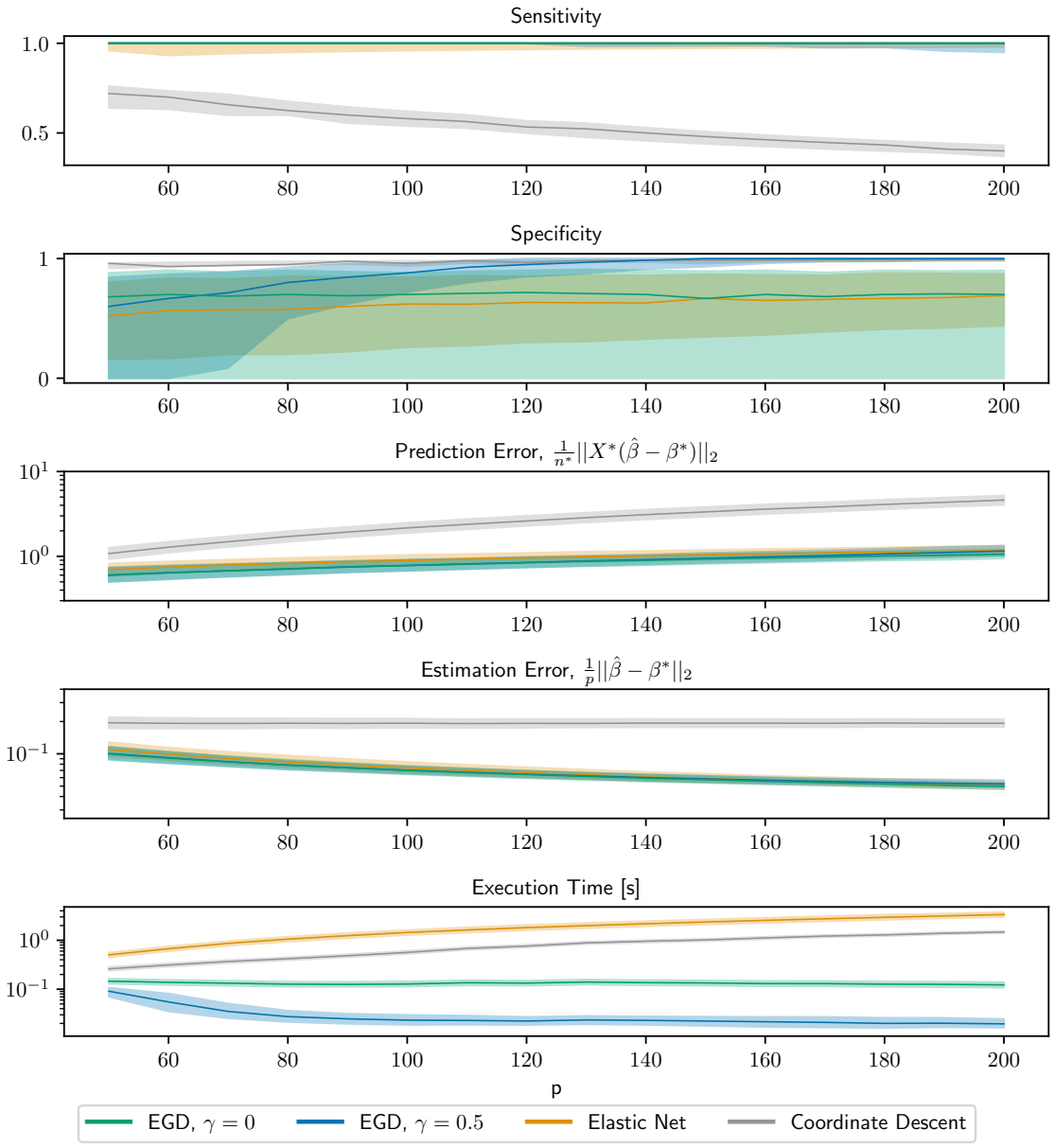

Figure 5: Median and first and third quartiles for the sensitivity, specificity, prediction and estimation errors, and execution time in seconds. Compared to the elastic net, elastic gradient descent performs similarly, except for execution time, where it is much faster. Compared to coordinate descent, elastic gradient descent performs better in all aspects except specificity. Elastic gradient descent performs faster, and has better specificity (especially when $p > n$), with momentum than without. The signal-to-noise ratio increases with the dimensionality and is, for some different values of p, $p = 50$: 18.5, $p = 100$: 72.5, $p = 150$: 162, $p = 200$: 287.

The results are presented in Table 2, where we compare execution time in seconds, R^2 (proportion of explained variation) on test data, and model size (number of non-zeros in $\hat{\boldsymbol{\beta}}$), and are in line with those of Section 4.3.

Elastic gradient descent and the elastic net perform similarly, apart from elastic gradient descent being up to three orders of magnitude faster. In contrast to coordinate descent, which is executed once, elastic gradient descent is executed nine times, for nine different values of α. Still, for all data sets except the California housing and protein structure data sets it performs faster than nine times the speed coordinate descent (with momentum, it is faster for all data sets). When p is large, elastic gradient descent tends to be faster than coordinate descent even in absolute numbers, with the exception of the English readability data.

Coordinate descent tends to select a sparser model, at the expense of a lower R^2. This is in line with the higher specificity of coordinate descent observed in Section 4.3. In addition to increasing the computational performance of elastic gradient descent, there is also a tendency for momentum to promote a sparser model, especially when p is large (again with the exception of the English readability data).

5. Conclusions

We proposed elastic gradient descent, a simple-to-implement, iterative optimization method, which generalizes gradient descent and coordinate descent (forward stagewise regression). We also investigated the case of infinitesimal optimization step size, presenting a piecewise analytical solution for solving linear regression with elastic gradient flow.

We compared elastic gradient descent with and without momentum to the elastic net and coordinate descent, both theoretically and on simulated and real data. Elastic gradient descent and the elastic net provided similar solutions, but with elastic gradient descent being up to three orders of magnitude faster on the investigated data. Compared to coordinate descent, elastic gradient descent selected a model with better performance, although still sparse. In addition to faster performance, adding momentum to elastic gradient descent promotes a sparser model for high dimensional data.

We used elastic gradient descent for standard linear regression. However, it would also be interesting to apply it for classification by extending it to logistic and multinomial regression. Furthermore, the optimization algorithm can be used instead of e.g. gradient descent on any optimization problem. For instance, it would be interesting to train a neural network with elastic gradient descent, obtaining a model that grows in complexity with optimization time.

Code is available at `https://github.com/allerbo/elastic_gradient_descent`.

Acknowledgments

This research was supported by funding from the Swedish Research Council (VR), the Swedish Foundation for Strategic Research, the Wallenberg AI, Autonomous Systems and Software Program (WASP), and the Chalmers AI Research Center (CHAIR).
OA would like to thank the editor and an anonymous reviewer, whose suggestions helped to improve the manuscript substantially.

Data set ($n \times p$)	Algorithm	Execution Time in Seconds	R^2	Model Size
Aspen Fibres (25165×5)	EGD, $\gamma = 0$	0.16, (0.13, 0.18)	0.49, (0.47, 0.51)	5, (5, 5)
	EGD, $\gamma = 0.5$	0.11, (0.082, 0.13)	0.49, (0.47, 0.51)	5, (5, 5)
	Elastic Net	6.2, (5.7, 6.4)	0.49, (0.47, 0.51)	5, (5, 5)
	CD	0.023, (0.013, 0.031)	0.48, (0.46, 0.50)	3, (3, 3)
California Housing (20640×8)	EGD, $\gamma = 0$	1.1, (1.1, 1.2)	0.60, (0.59, 0.61)	8, (8, 8)
	EGD, $\gamma = 0.5$	0.62, (0.57, 0.69)	0.60, (0.59, 0.62)	8, (7, 8)
	Elastic Net	11, (9.7, 11)	0.60, (0.59, 0.62)	8, (7, 8)
	CD	0.083, (0.053, 0.11)	0.57, (0.55, 0.59)	5, (4, 6)
U.K. Black Smoke (45568×10)	EGD, $\gamma = 0$	0.26, (0.22, 0.30)	0.14, (0.13, 0.14)	10, (10, 10)
	EGD, $\gamma = 0.5$	0.20, (0.17, 0.23)	0.14, (0.13, 0.14)	10, (10, 10)
	Elastic Net	27, (27, 28)	0.14, (0.13, 0.15)	10, (10, 10)
	CD	0.039, (0.022, 0.056)	0.13, (0.13, 0.14)	7, (7, 7)
Portugese Elections (21643×18)	EGD, $\gamma = 0$	0.61, (0.55, 0.67)	0.11, (0.087, 0.12)	14, (12, 15)
	EGD, $\gamma = 0.5$	0.35, (0.30, 0.40)	0.11, (0.087, 0.12)	12, (12, 15)
	Elastic Net	74, (66, 83)	0.10, (0.087, 0.12)	12, (9, 14)
	CD	0.11, (0.083, 0.15)	0.10, (0.086, 0.12)	7, (5, 7)
Appliances Energy Use (19735×27)	EGD, $\gamma = 0$	4.2, (3.9, 4.6)	0.16, (0.15, 0.17)	27, (27, 27)
	EGD, $\gamma = 0.5$	2.9, (2.6, 3.1)	0.16, (0.15, 0.17)	27, (27, 27)
	Elastic Net	83, (80, 85)	0.16, (0.15, 0.18)	27, (27, 27)
	CD	1.2, (1.1, 1.4)	0.078, (0.073, 0.083)	5, (4, 5)
Protein Structure (45730×9)	EGD, $\gamma = 0$	1.3, (1.2, 1.4)	0.24, (0.24, 0.25)	7, (7, 8)
	EGD, $\gamma = 0.5$	0.86, (0.73, 0.97)	0.26, (0.26, 0.27)	9, (9, 9)
	Elastic Net	120, (120, 120)	0.28, (0.27, 0.29)	9, (9, 9)
	CD	0.11, (0.077, 0.14)	0.15, (0.15, 0.16)	2, (2, 2)
Super-conductors (21263×81)	EGD, $\gamma = 0$	2.6, (2.2, 2.8)	0.70, (0.70, 0.71)	81, (81, 81)
	EGD, $\gamma = 0.5$	1.1, (0.93, 1.2)	0.66, (0.66, 0.67)	77, (76, 77)
	Elastic Net	340, (330, 360)	0.72, (0.71, 0.73)	64, (64, 65)
	CD	1.5, (1.2, 1.7)	0.62, (0.62, 0.63)	15, (14, 15)
English Readability (2834×768)	EGD, $\gamma = 0$	1.7, (1.5, 1.7)	0.69, (0.67, 0.71)	105, (89, 123)
	EGD, $\gamma = 0.5$	1.2, (1.0, 1.3)	0.70, (0.68, 0.72)	115, (97, 144)
	Elastic Net	1400, (1300, 1500)	0.73, (0.71, 0.74)	304, (265, 374)
	CD	0.26, (0.22, 0.32)	0.58, (0.56, 0.60)	22, (21, 24)
Twitter Popularity (291624×77)	EGD, $\gamma = 0$	48, (41, 55)	0.94, (0.89, 0.95)	60, (57, 61)
	EGD, $\gamma = 0.5$	6.0, (5.3, 7.0)	0.93, (0.89, 0.94)	27, (24, 29)
	Elastic Net	5100, (4800, 5300)	0.94, (0.90, 0.94)	49, (48, 50)
	CD	360, (350, 370)	0.93, (0.92, 0.94)	8, (7, 8)

Table 2: Median and first and third quartiles (within parenthesis) of execution time (in seconds), R^2, and model size when applying the three algorithms on the nine real data sets. Elastic gradient descent performs significantly faster than the elastic net. Adding momentum further increases the computational speed. Coordinate descent tends to select a sparser model. While coordinate descent is evaluated only once (for $\alpha = 1$), elastic gradient descent and the elastic net are evaluated nine times (for $\alpha \in \{0.1, \dots\ 0.9\}$).

Appendix A. Connection to Steepest Descent and the General Stagewise Procedure

In this section, we redefine elastic gradient descent within the frameworks of steepest descent (Boyd and Vandenberghe, 2004) and the general stagewise procedure (Tibshirani, 2015), obtaining two related, but slightly different, flavors of Equation 4.

A.1 Steepest Descent

Steepest descent (Boyd and Vandenberghe, 2004), generalizes coordinate and gradient descent. For a given norm $\|\cdot\|$, $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}$ is given by

$$\begin{aligned} \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{sd}}(t) &= \underset{\boldsymbol{v}:\ \|\boldsymbol{v}\|=1}{\operatorname{argmax}}\, \boldsymbol{g}(t)^\top \boldsymbol{v} \\ \hat{\boldsymbol{\beta}}(t+\Delta t) &= \hat{\boldsymbol{\beta}}(t) - \Delta t \cdot \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{sd}}(t). \end{aligned} \tag{13}$$

For the ℓ_2 norm, steepest descent becomes normalized gradient descent,

$$\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{gd,sd}}(t) = \frac{\boldsymbol{g}(t)}{\|\boldsymbol{g}(t)\|_2} = \frac{\boldsymbol{I}_{\text{gd}} \cdot \boldsymbol{g}(t)}{\|\boldsymbol{I}_{\text{gd}} \cdot \boldsymbol{g}(t)\|_2},$$

while the ℓ_1 norm corresponds to coordinate descent,

$$\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{cd,sd}}(t) = \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{cd}}(t) = \boldsymbol{I}_{\text{cd}}(t) \cdot \operatorname{sign}(\boldsymbol{g}(t)) = \frac{\boldsymbol{I}_{\text{cd}}(t) \cdot \boldsymbol{g}(t)}{\|\boldsymbol{I}_{\text{cd}}(t) \cdot \boldsymbol{g}(t)\|_1}.$$

When formulating elastic gradient descent, inspired by Equation 2, we would like to use $\alpha\|\boldsymbol{v}\|_1 + (1-\alpha)\|\boldsymbol{v}\|_2^2 = 1$ in Equation 13, however then there is no analytical solution to the equation. Instead, we use the following strategy to obtain an approximate solution:

1. Define $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}$ as a generalization of both $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{cd,sd}}$ and $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{gd,sd}}$, such that
   (a) the model selection property of the elastic net is obtained,
   (b) $\alpha\|\boldsymbol{v}\|_1 + (1-\alpha)\|\boldsymbol{v}\|_2^2 = 1$.
2. Within the freedom remaining after step 1, tune $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}$ to, approximately, maximize $\boldsymbol{g}^\top \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}$.

Combining $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{gd,sd}}$ and $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{cd,sd}}$ in the same way as was done in Equation 4, we define

$$\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}(t) := \boldsymbol{I}_{\text{egd,sd}}(\alpha, t) \cdot \boldsymbol{g}(t) \left(\frac{\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}(\alpha, t) \cdot \boldsymbol{g}(t)\|_1} + \frac{1-\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}(\alpha, t) \cdot \boldsymbol{g}(t)\|_2} \right), \tag{14}$$

where $\boldsymbol{I}_{\text{egd,sd}}$ is a diagonal matrix with zeros and ones on the diagonal, such that $\boldsymbol{I}_{\text{egd,sd}}(0,t) = \boldsymbol{I}_{\text{gd}} = \boldsymbol{I}$ and $\boldsymbol{I}_{\text{egd,sd}}(1,t) = \boldsymbol{I}_{\text{cd}}(t)$. However, $\boldsymbol{I}_{\text{egd,sd}}$ is not necessarily identical to $\boldsymbol{I}_{\text{egd}}$.

We define $p_1(t) \in [1, p]$ to be the number of ones in $\boldsymbol{I}_{\text{egd,sd}}$, i.e. the number of parameters that are updated at time t:

Definition 5 (p_1)**.**

$$p_1(\alpha, t) := \sum_{d=1}^{p} \left(\boldsymbol{I}_{\text{egd,sd}}(\alpha, t)\right)_{dd}.$$

As optimization proceeds toward convergence, all gradient components approach zero, and thus each other. This means that p_1 increases (i.e. more parameters are updated), but not necessarily monotonically, toward p. However, for $\alpha > 0$, some absolute gradient components may oscillate around $\alpha \cdot |g_m|$, being updated in one time step, but not in the next. In that case, we may have $p_1 < p$ during the entire training. Also, note that p_1 is not explicitly defined; its value is a consequence of Definition 5.

Now, to obtain $\alpha \left\| \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}} \right\|_1 + (1-\alpha) \left\| \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}} \right\|_2^2 = 1$, $\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}$ needs to be scaled, as specified in Proposition 6.

Proposition 6.

$$\begin{aligned}
&\textit{Let } q_1(t) := \left(\frac{\|\boldsymbol{I}_{\text{egd,sd}}(\alpha, t) \cdot \boldsymbol{g}(t)\|_1}{\|\boldsymbol{I}_{\text{egd,sd}}(\alpha, t) \cdot \boldsymbol{g}(t)\|_2} \right)^2 \textit{ and let} \\
&c_\alpha(t) := \frac{\sqrt{q_1(t) \cdot (\alpha^2 q_1(t) + 4(1-\alpha))} - \alpha \cdot q_1(t)}{2(1-\alpha)\left(\sqrt{q_1(t)} \cdot (1-\alpha) + \alpha\right)}. \\
&\textit{Then } \alpha \cdot \left\| c_\alpha(t) \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}(t) \right\|_1 + (1-\alpha) \cdot \left\| c_\alpha(t) \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}(t) \right\|_2^2 = 1.
\end{aligned}$$

c_α depends both on α and the quotient between the ℓ_1 and ℓ_2 norms in a quite complicated form. However, according to Proposition 7, in the absence of c_α the distance from 1 is still bounded:

Proposition 7.
For $\alpha \in [0, 1]$, $1 \leq p_1 \leq p$

$$\begin{aligned}
0.61 < 1 - \alpha(1-\alpha)(2-\alpha) \cdot \left(1 - \frac{1}{p_1(t)}\right) \\
\leq \alpha \left\| \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}(t) \right\|_1 + (1-\alpha) \left\| \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}(t) \right\|_2^2 \\
\leq 1 + \alpha(1-\alpha) \cdot \left(\sqrt{p_1(t)} - 1\right) \leq 1 + \frac{\sqrt{p_1(t)} - 1}{4}.
\end{aligned}$$

What remains to do, is to select $\boldsymbol{I}_{\text{egd,sd}}$ to maximize $c_\alpha \boldsymbol{g}^\top \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}$. Since $\boldsymbol{g}^\top \boldsymbol{I}_{\text{egd,sd}} \boldsymbol{g} = \boldsymbol{g}^\top \boldsymbol{I}_{\text{egd,sd}} \boldsymbol{I}_{\text{egd,sd}} \boldsymbol{g} = \|\boldsymbol{I}_{\text{egd,sd}} \boldsymbol{g}\|_2^2$, and since $c_\alpha \geq 0$ for $q_1 \geq 0$ and $\alpha \in [0, 1]$, maximizing $c_\alpha \boldsymbol{g}^\top \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}$ amounts to maximizing

$$\boldsymbol{g}(t)^\top \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}(t) = \alpha \frac{\|\boldsymbol{I}_{\text{egd,sd}}(\alpha, t) \cdot \boldsymbol{g}(t)\|_2^2}{\|\boldsymbol{I}_{\text{egd,sd}}(\alpha, t) \cdot \boldsymbol{g}(t)\|_1} + (1-\alpha) \|\boldsymbol{I}_{\text{egd,sd}}(\alpha, t) \cdot \boldsymbol{g}(t)\|_2 .$$

The second term trivially increases with p_1 while, according to Lemma 8, the first term decreases with p_1.

Lemma 8.

$$\frac{\|\boldsymbol{I}_{\text{egd,sd}} \cdot \boldsymbol{g}\|_2^2}{\|\boldsymbol{I}_{\text{egd,sd}} \cdot \boldsymbol{g}\|_1} \textit{ is a decreasing function in } p_1.$$

The exact trade-off between the two terms depends on the gradient at the specific time step and has no general solution. However, when α is large we want $\|\boldsymbol{I}_{\text{egd,sd}}\|_2^2 / \|\boldsymbol{I}_{\text{egd,sd}}\|_1$ to be large, i.e. we want p_1 to be small. When α is small, we want $\|\boldsymbol{I}_{\text{egd,sd}}\|_2$ to be large, i.e. we want p_1 to be large. This desire is consistent with how we defined $\boldsymbol{I}_{\text{egd}}$ in Definition 1, and we thus define $\boldsymbol{I}_{\text{egd,sd}} := \boldsymbol{I}_{\text{egd}}$. This means that we use the same gradient selection matrix as in the original formulation of elastic gradient descent.

A.2 The General Stagewise Procedure

The general stagewise procedure (Tibshirani, 2015) is formulated similarly to steepest descent, but while the purpose of steepest descent just is to find the optimal solution, in the general stagewise procedure, the entire solution path is of interest. Here, the norm in the constraint is replaced by any convex function, h, and the optimization step size, Δt, is incorporated into $\boldsymbol{\Delta\hat{\beta}}$:

$$\begin{aligned}\boldsymbol{\Delta\hat{\beta}}_{\text{gs}}(t) &= \underset{h(\boldsymbol{v})\leq\Delta t}{\operatorname{argmax}}\, \boldsymbol{g}(t)^\top \boldsymbol{v}\\ \hat{\boldsymbol{\beta}}(t+\Delta t) &= \hat{\boldsymbol{\beta}}(t) - \boldsymbol{\Delta\hat{\beta}}_{\text{gs}}(t).\end{aligned}$$

In this framework we obtain

$$\begin{aligned}\boldsymbol{\Delta\hat{\beta}}_{\text{cd,gs}}(t) &= \Delta t \cdot \frac{\boldsymbol{I}_{\text{cd}}(t)\cdot\boldsymbol{g}(t)}{\|\boldsymbol{I}_{\text{cd}}(t)\cdot\boldsymbol{g}(t)\|_1} = \Delta t\cdot\boldsymbol{\Delta\hat{\beta}}_{\text{cd}}(t)\\ \boldsymbol{\Delta\hat{\beta}}_{\text{gd,gs}}(t) &= \sqrt{\Delta t}\cdot\frac{\boldsymbol{g}(t)}{\|\boldsymbol{g}(t)\|_2} = \sqrt{\Delta t}\cdot\frac{\boldsymbol{I}_{\text{gd}}\cdot\boldsymbol{g}(t)}{\|\boldsymbol{I}_{\text{gd}}\cdot\boldsymbol{g}(t)\|_2} = \sqrt{\Delta t}\cdot\boldsymbol{\Delta\hat{\beta}}_{\text{gd,sd}}(t),\end{aligned}$$

which suggests

$$\boldsymbol{\Delta\hat{\beta}}_{\text{egd,gs}}(t) := \boldsymbol{I}_{\text{egd}}(\alpha,t)\cdot\boldsymbol{g}(t)\left(\frac{\alpha\Delta t}{\|\boldsymbol{I}_{\text{egd}}(\alpha,t)\cdot\boldsymbol{g}(t)\|_1} + \frac{(1-\alpha)\sqrt{\Delta t}}{\|\boldsymbol{I}_{\text{egd}}(\alpha,t)\cdot\boldsymbol{g}(t)\|_2}\right). \tag{15}$$

The analogs of Propositions 6 and 7 in this framework are presented in Propositions 9 and 10.

Proposition 9.

Let $\boldsymbol{\Delta\hat{\beta}}_{\text{egd,gs,c}}(t) := \boldsymbol{I}_{\text{egd}}(\alpha,t)\cdot\boldsymbol{g}(t)\left(\frac{\alpha\cdot c_{\alpha,\Delta t}(t)\cdot\Delta t}{\|\boldsymbol{I}_{\text{egd}}(\alpha,t)\cdot\boldsymbol{g}(t)\|_1} + \frac{(1-\alpha)\sqrt{c_{\alpha,\Delta t}(t)\cdot\Delta t}}{\|\boldsymbol{I}_{\text{egd}}(\alpha)\cdot\boldsymbol{g}(t)\|_2}\right)$

for $c_{\alpha,\Delta t}(t) :=$

$$\left(\frac{\sqrt{2\alpha\sqrt{q_1\cdot(\alpha^2 q_1 + 4\Delta t(1-\alpha))} + q_1\cdot((1-\alpha)^3 - 2\alpha^2)} - (1-\alpha)\sqrt{q_1\cdot(1-\alpha)}}{\alpha\sqrt{4\Delta t(1-\alpha)}}\right)^2,$$

where $q_1 = q_1(t) := \left(\frac{\|\boldsymbol{I}_{\text{egd}}(\alpha,t)\cdot\boldsymbol{g}(t)\|_1}{\|\boldsymbol{I}_{\text{egd}}(\alpha,t)\cdot\boldsymbol{g}(t)\|_2}\right)^2.$

Then $\alpha\cdot\left\|\boldsymbol{\Delta\hat{\beta}}_{\text{egd,gs,c}}(t)\right\|_1 + (1-\alpha)\cdot\left\|\boldsymbol{\Delta\hat{\beta}}_{\text{egd,gs,c}}(t)\right\|_2^2 = \Delta t.$

Proposition 10.
For $\alpha \in [0,1]$, $1 \leq p_1 \leq p$

$$\begin{aligned} 0.61 \cdot \Delta t &< \Delta t \left(1 - \alpha(1-\alpha)(2-\alpha) \cdot \left(1 - \frac{\Delta t}{p_1(t)}\right)\right) \\ &\leq \alpha \left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs}}(t)\right\|_1 + (1-\alpha)\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs}}(t)\right\|_2^2 \\ &\leq \Delta t \left(1 + \alpha(1-\alpha) \cdot \left(\sqrt{\frac{p_1(t)}{\Delta t}} - 1\right)\right) \leq \Delta t \left(1 + \frac{\sqrt{p_1(t)/\Delta t} - 1}{4}\right). \end{aligned}$$

A.3 Comparing the Formulations

Compared to the original formulation of elastic gradient descent in Equation 4, both the steepest descent and general stagewise formulations differ in the normalization of the second term. Apart from that, the general stagewise formulation uses different step sizes for the coordinate and gradient descent contributions, where the difference grows with smaller Δt (assuming $\Delta t < 1$).

We also note that compared to Proposition 7, in Proposition 10, p_1 is replaced by $p_1/\Delta t$. This means that if Δt is small, $\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs}}\right\|_1 + (1-\alpha)\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs}}\right\|_2^2$ might deviate quite much from Δt.

According to our empirical experience, however, all flavors of elastic gradient descent, i.e. Equation 4, and Equations 14 and 15 with and without scaling, provide virtually identical solution paths. In Figure 6, we compare the solution paths for the diabetes data for the different flavors of elastic gradient descent with $\alpha = 0.5$. The paths, displayed in the first column, are hardly, if at all, distinguishable. The second column shows the normalized gradients. Just as for the solution paths, the gradients evolve very similarly between the five versions, even though some differences are visible.

The third column shows how $\alpha \left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}\right\|_1 + (1-\alpha)\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}\right\|_2^2 =: h_\alpha(\boldsymbol{\Delta}\hat{\boldsymbol{\beta}})$ deviates from 1 (or Δt), together with the two bounds provided by Propositions 7 and 10 and with p_1, which is displayed on a different y-scale. It can be seen how h_α is exactly 1 (Δt) in the scaled case and how it stays within the bounds in the unscaled case. As expected, h_α deviates more from Δt in the general stagewise framework than it does from 1 in the steepest descent framework. For standard gradient descent, h_α deviates a lot from 1. It can also be noted how the effective value of p_1 is always strictly lower than $p = 10$, which can be attributed to the oscillations around $\pm\alpha \cdot |g_m|$, i.e. the coupled parameters are included at some time steps and excluded at other.

Appendix B. Additional Experiments

In this section, we provide additional experiments.

B.1 Illustration of Momentum Induced Sparsity

In Figure 7, we compare the solution paths of elastic gradient descent with and without momentum, for a step size of 0.01. The inertia introduced by the momentum causes β_2 to be

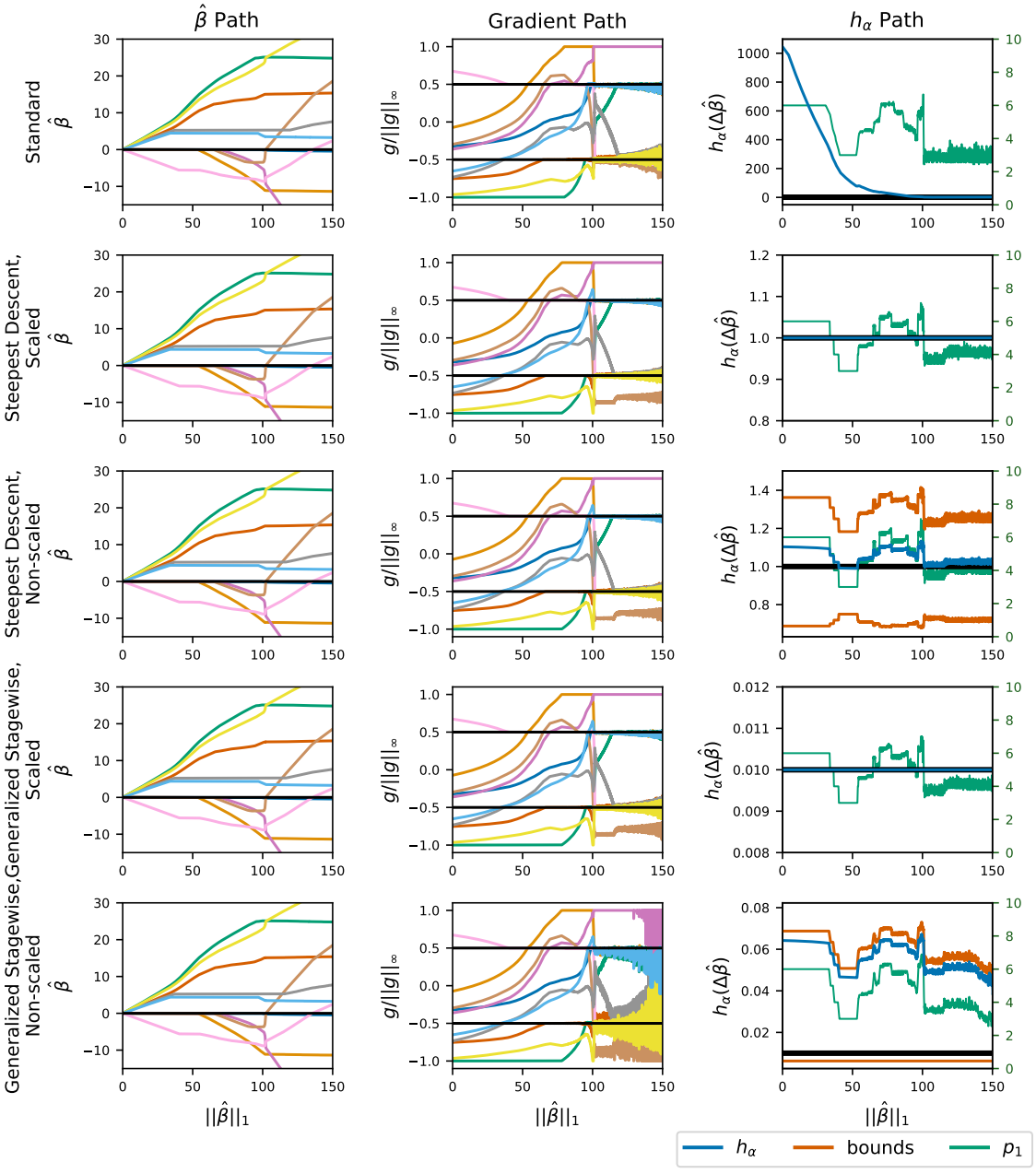

Figure 6: Solution paths for elastic gradient flow and the different flavors of elastic gradient descent with step size 0.01, without momentum, at $\alpha = 0.5$ on the diabetes data. The second column shows the normed gradients, with black lines at $\pm\alpha$. The third column shows how $h_\alpha(\boldsymbol{\Delta}\hat{\boldsymbol{\beta}})$ deviates from 1 (or Δt), together with bounds from Propositions 7 and 10. On the right y-axis, p_1 is plotted. To increase readability a moving average with width 9 was applied to the graphs in the third column.

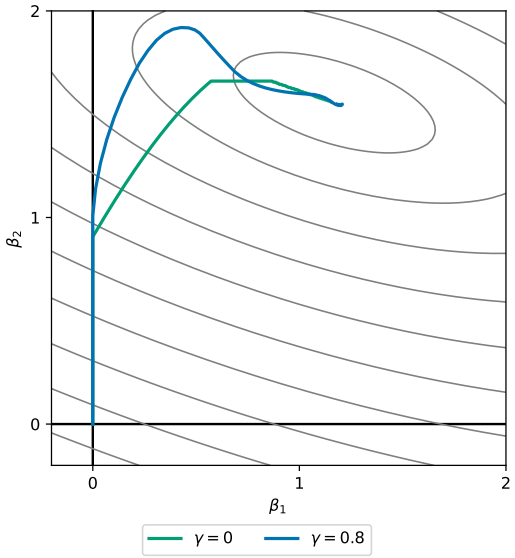

Figure 7: Solution paths of elastic gradient descent with and without momentum. With momentum, β_2 is included into the model later, i.e. for a smaller reconstruction error, than without momentum.

zero for a longer time, (it is included into the model at a smaller reconstruction error) than when no momentum is used.

B.2 Synthetic Data for Model Selection

In this section, we extend the simulation of Section 4.3.

To further investigate the specificity of the algorithms, we changed the experiment setup, so that we always used 40 non-zero parameters, with the remaining $p - 40$ parameters being zero, keeping all other aspects the same. The results are presented in Figure 8. This time, momentum no longer provides the same advantage in specificity as before. This can probably be attributed to the fact that in the first case, when the number of non-zero parameters grows, so does the value of the maximum initial gradient, which makes it harder for a zero parameter to be erroneously included, especially with momentum when early stages of training matter more, and thus leads to a better specificity. On the other hand, in this second case, when the number of non-zero parameters, and thus the maximum initial gradient, is constant while the number of parameters to possibly erroneously include increases, this advantage of momentum is less prominent.

We also extended the experiments of Section 4.3 by using the same setup as in the original experiments, except using $\rho_1 \in [0.5,\ 0.6,\ 0.7,\ 0.8,\ 0.9,]$, $\rho_2 \in [0.0,\ 0.1,\ 0.2,\ 0.3,\ 0.4,\ 0.5]$ and $p \in [50,\ 100,\ 200]$.

The results are presented in Figures 9, 10 and 11. The signal-to-noise ratios of the problems, which were essentially constant with respect to ρ_2, are stated for the different values of ρ_1.

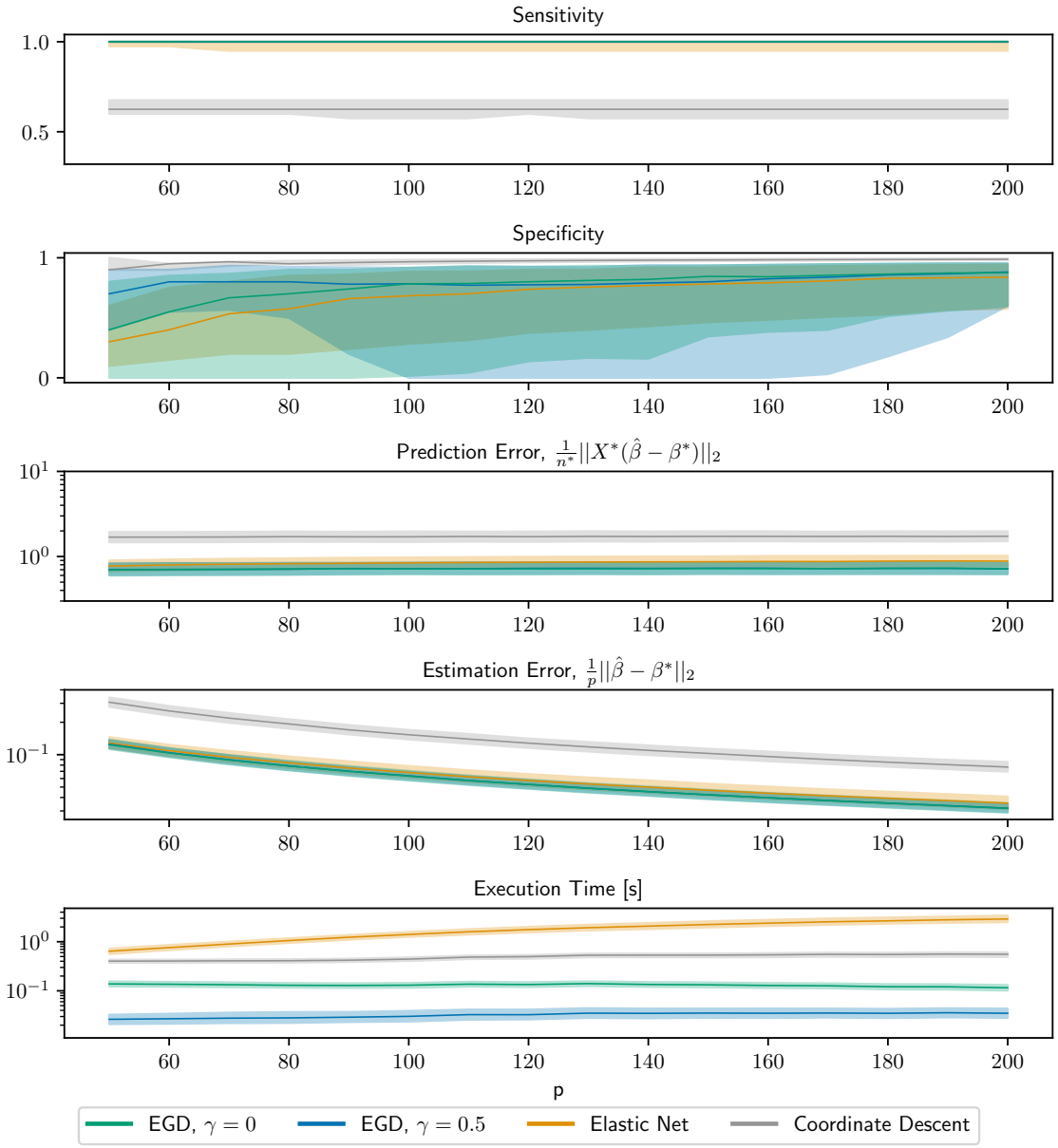

Figure 8: Median and first and third quartiles for the sensitivity, specificity, prediction and estimation errors, and execution time in seconds. Compared to in Figure 5, momentum no longer provides the same advantage in specificity as before. The signal-to-noise ratio is always between 46.4 and 46.9.

The conclusions are consistent with those from Section 4.3: Elastic gradient descent and the elastic net perform similarly in all aspects except computational time, where elastic gradient descent performs significantly faster. The computational performance of elastic gradient descent improves with momentum. For high-dimensional data ($p > n = 100$), where no unique solution exists, momentum also greatly improves the model specificity. Compared to coordinate descent, elastic gradient descent performs better in all aspects except for specificity. The higher specificity of coordinate descent, however, comes at the cost of much worse sensitivity, and prediction and estimation errors.

Appendix C. Elastic Gradient Flow

In this section, we investigate the limits as the step size, Δt, goes to zero when solving linear least squares with gradient, coordinate, and elastic gradient descent with momentum. Gradient descent with infinitesimal step size is known as gradient flow, and analogously we use the terms coordinate flow and elastic gradient flow. We start by reviewing gradient flow and then consider coordinate flow and elastic gradient flow, where the latter generalizes both gradient and coordinate flow. Since it is not obvious that the limits of $\boldsymbol{I}_{\text{cd}}(t)$ and $\boldsymbol{I}_{\text{egd}}(t)$ exist as $\Delta t \to 0$, coordinate and elastic gradient flow are presented as well-motivated definitions rather than as theorems, with motivations in Section C.4.

In the following, subscript with respect to a set denotes the sub-matrix (or sub-vector) specified by the indices in the set, $(\cdot)^{(k)}$ denotes the time derivative of order k, and $\odot$ denotes element-wise multiplication.

For linear least squares, the gradient at time t is

$$\boldsymbol{g}(t) := \nabla_{\hat{\boldsymbol{\beta}}(t)} \left(\frac{1}{2n} \|\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}(t)\|_2^2 \right) = \frac{1}{n} \boldsymbol{X}^\top (\boldsymbol{X}\hat{\boldsymbol{\beta}}(t) - \boldsymbol{y}) = -\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t)), \tag{16}$$

where $\hat{\boldsymbol{\Sigma}} := \frac{1}{n}\boldsymbol{X}^\top \boldsymbol{X}$ is the empirical covariance matrix and $\hat{\boldsymbol{\beta}}^{\text{OLS}} := (\boldsymbol{X}^\top \boldsymbol{X})^+ \boldsymbol{X}^\top \boldsymbol{y}$ (where $(\cdot)^+$ denotes the Moore-Penrose pseudoinverse) is the minimum norm ordinary least squares solution for $t = \infty$. The last equality in Equation 16 follows from $\boldsymbol{X}^\top = (\boldsymbol{X}^\top \boldsymbol{X})(\boldsymbol{X}^\top \boldsymbol{X})^+ \boldsymbol{X}^\top$.

C.1 Gradient Flow

When linear regression is solved using gradient descent with momentum, $\hat{\boldsymbol{\beta}}$ is updated iteratively according to

$$\begin{aligned} \hat{\boldsymbol{\beta}}(t + \Delta t) =& \hat{\boldsymbol{\beta}}(t) + \gamma \left(\hat{\boldsymbol{\beta}}(t) - \hat{\boldsymbol{\beta}}(t - \Delta t) \right) - \Delta t \cdot \boldsymbol{g}(t) \\ =& \hat{\boldsymbol{\beta}}(t) + \gamma \left(\hat{\boldsymbol{\beta}}(t) - \hat{\boldsymbol{\beta}}(t - \Delta t) \right) + \Delta t \cdot \hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t)). \end{aligned} \tag{17}$$

Moving all but the last term to the left-hand side, dividing by Δt and then letting $\Delta t \to 0$ results in the differential equation

$$(1 - \gamma) \cdot \frac{\partial \hat{\boldsymbol{\beta}}(t)}{\partial t} = \hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t)),$$

which has the solution

$$\hat{\boldsymbol{\beta}}(t) = \hat{\boldsymbol{\beta}}^{\text{OLS}} - \exp\left(-\frac{t}{1 - \gamma} \hat{\boldsymbol{\Sigma}} \right) (\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\boldsymbol{0}}),$$

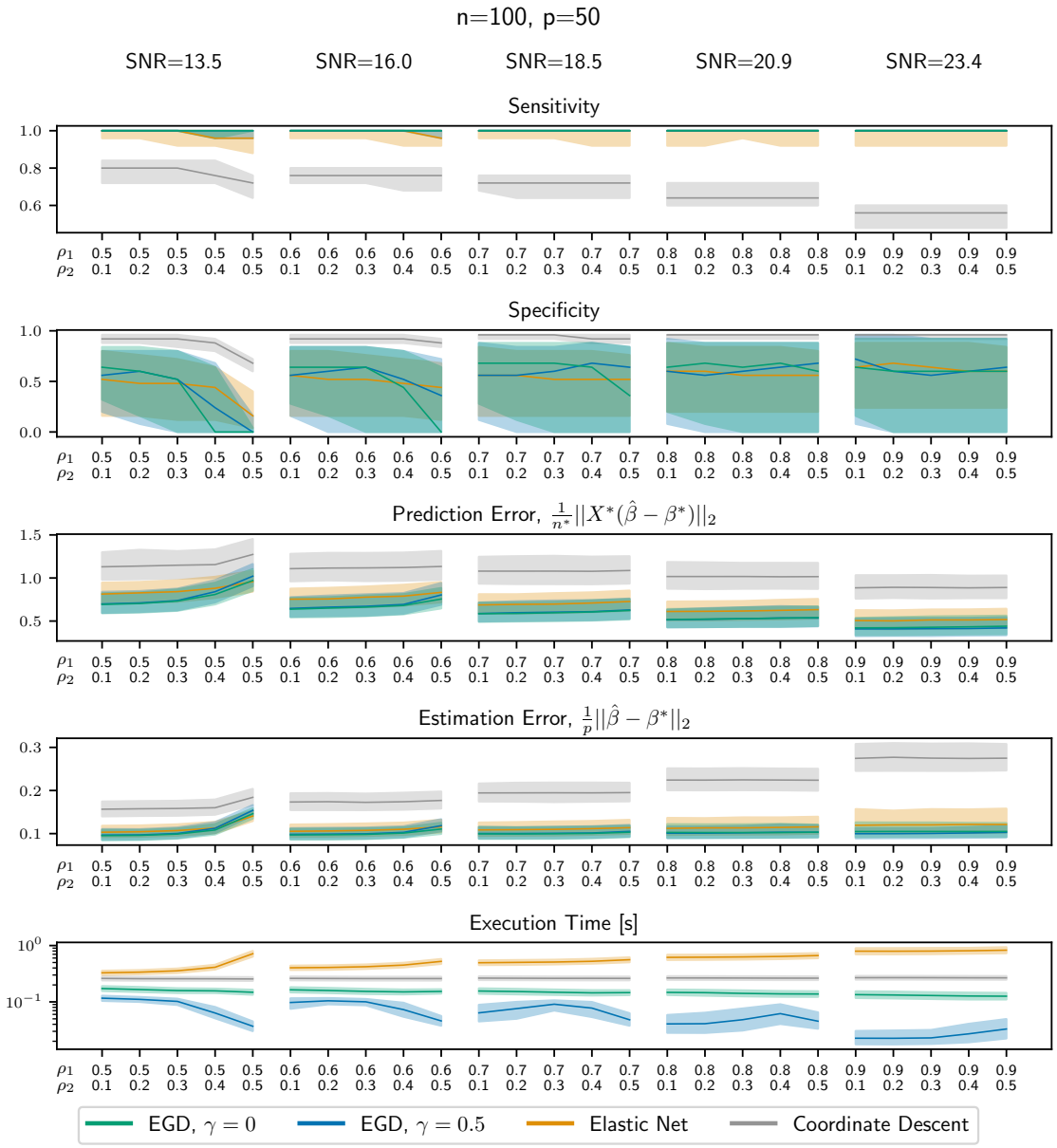

Figure 9: Median and first and third quartiles for the sensitivity, specificity, prediction and estimation errors, and execution time in seconds, in the low-dimensional case. The value of ρ_1 is constant within each of the five panels in each row, while ρ_2 varies. The signal-to-noise ratio of the problem, which is essentially constant with respect to ρ_2, is stated for the different values of ρ_1. Compared to the elastic net, elastic gradient descent performs similarly, except for execution time, where it is much faster. Compared to coordinate descent, elastic gradient descent performs better in all aspects except specificity. Elastic gradient descent performs faster with momentum than without.

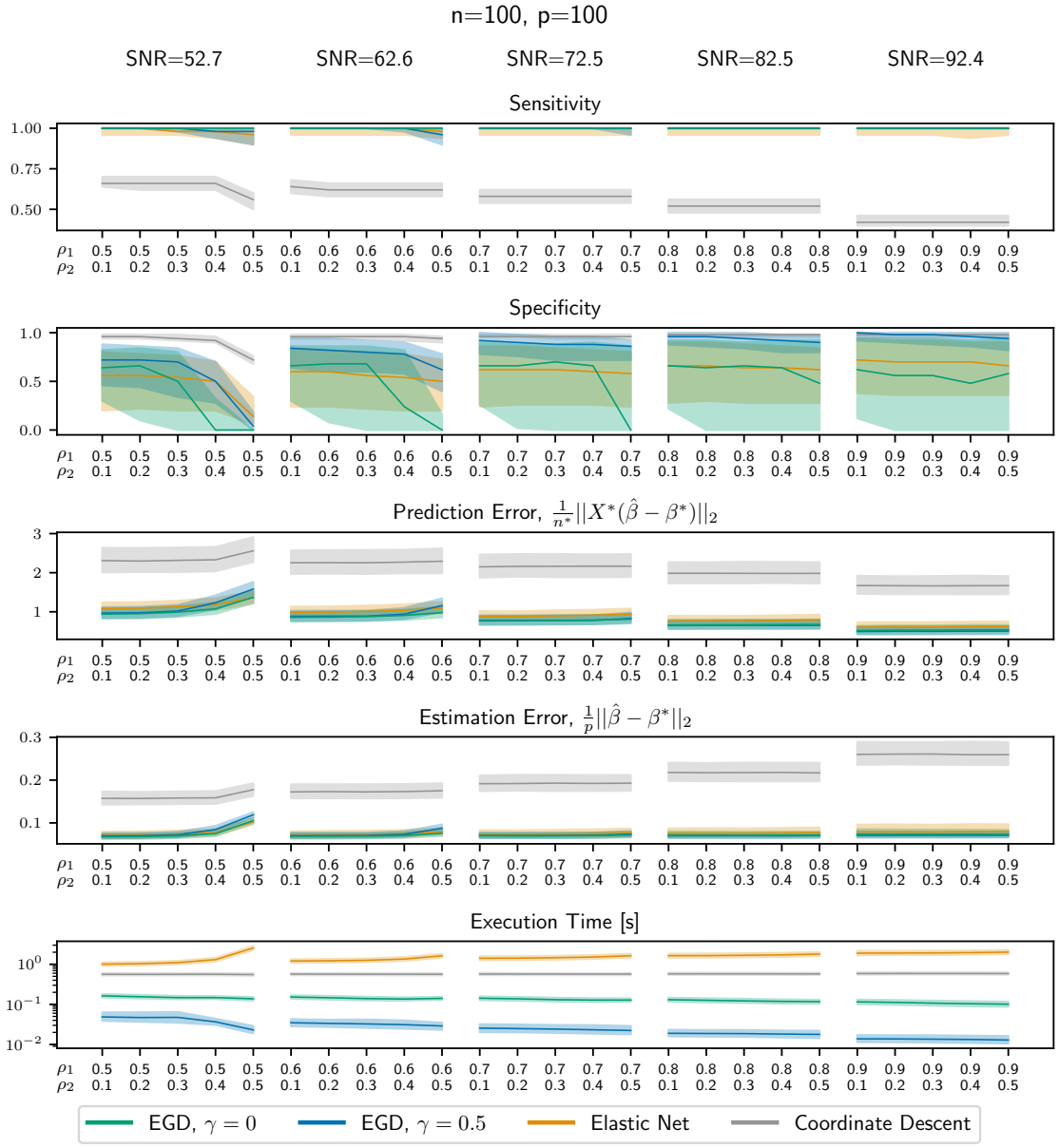

Figure 10: Median and first and third quartiles for the sensitivity, specificity, prediction and estimation errors, and execution time in seconds, when $n = p$. The value of ρ_1 is constant within each of the five panels in each row, while ρ_2 varies. The signal-to-noise ratio of the problem, which is essentially constant with respect to ρ_2, is stated for the different values of ρ_1. Compared to the elastic net, elastic gradient descent performs similarly, except for execution time, where it is much faster. Compared to coordinate descent, elastic gradient descent performs better in all aspects except specificity. Elastic gradient descent performs faster, and has better specificity, with momentum than without.

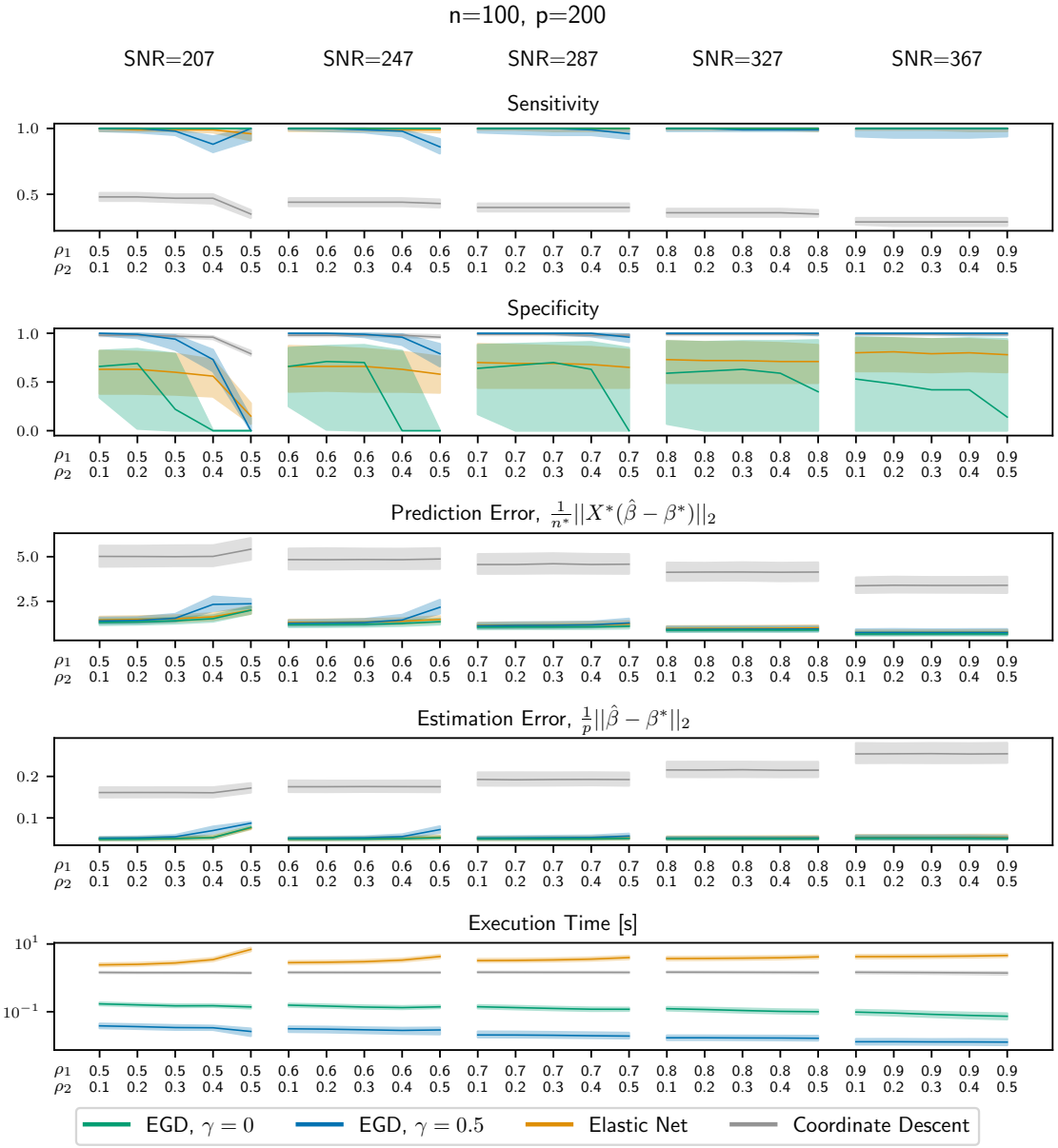

Figure 11: Median and first and third quartiles for the sensitivity, specificity, prediction and estimation errors, and execution time in seconds, in the high-dimensional case. The value of ρ_1 is constant within each of the five panels in each row, while ρ_2 varies. The signal-to-noise ratio of the problem, which is essentially constant with respect to ρ_2, is stated for the different values of ρ_1. Compared to the elastic net, elastic gradient descent performs similarly, or better, except for execution time, where it is much faster. Compared to coordinate descent, elastic gradient descent performs better in all aspects except specificity. Elastic gradient descent performs faster, and has better specificity, with momentum than without.

where exp denotes the matrix exponential.

For $\hat{\boldsymbol{\beta}}_{\mathbf{0}} = \mathbf{0}$ the gradient flow estimate becomes

$$\hat{\boldsymbol{\beta}}_{\text{gf}}(t) = \left(\boldsymbol{I} - \exp\left(-\frac{t}{1-\gamma}\hat{\boldsymbol{\Sigma}}\right)\right)\hat{\boldsymbol{\beta}}^{\text{OLS}}. \tag{18}$$

C.2 Coordinate Flow

When linear regression is solved using coordinate descent with momentum, $\hat{\boldsymbol{\beta}}$ is updated iteratively according to

$$\begin{aligned}\hat{\boldsymbol{\beta}}(t+\Delta t) =& \hat{\boldsymbol{\beta}}(t) + \gamma\left(\hat{\boldsymbol{\beta}}(t) - \hat{\boldsymbol{\beta}}(t-\Delta t)\right) - \Delta t \cdot \boldsymbol{I}_{\text{cd}}(t) \cdot \text{sign}(\boldsymbol{g}(t)) \\ =& \hat{\boldsymbol{\beta}}(t) + \gamma\left(\hat{\boldsymbol{\beta}}(t) - \hat{\boldsymbol{\beta}}(t-\Delta t)\right) + \Delta t \cdot \boldsymbol{I}_{\text{cd}}(t) \cdot \text{sign}(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t)).\end{aligned} \tag{19}$$

When the magnitudes of two or more gradient components are all close to the maximum gradient value, the parameter to update changes in almost every time step, that is the 1 in $\boldsymbol{I}_{\text{cd}}$ changes position in almost every time step. As $\Delta t \to 0$, we would like to replace $\boldsymbol{I}_{\text{cd}}$ with a coordinate flow version, $\boldsymbol{I}_{\text{cf}}$. In contrast to $\boldsymbol{I}_{\text{cd}}$, where only one diagonal element is 1 and the rest are 0, for $\boldsymbol{I}_{\text{cf}}$ we allow multiple diagonal elements to be non-zero, as long as the sum of the diagonal elements is 1. The only 1 in $\boldsymbol{I}_{\text{cd}}$ is now distributed along the diagonal of $\boldsymbol{I}_{\text{cf}}$, with $(\boldsymbol{I}_{\text{cf}})_{dd} > 0$ if and only if d belongs to the active set, i.e. the set of indices between which the 1 in $\boldsymbol{I}_{\text{cd}}$ alters. This leads us to define coordinate flow according to Definition 11, where details behind the definition are presented in Section C.4.1.

The coordinate flow estimate, $\hat{\boldsymbol{\beta}}_{\text{cf}}(t)$, changes linearly in time with a slope controlled by the piece-wise constant matrix $\boldsymbol{I}_{\text{cf}}$. At certain times, t_i, $\boldsymbol{I}_{\text{cf}}$ is updated to a new constant matrix. These times correspond to changes in the active set, S_A, which specifies which parameters are updated, namely the ones with non-zero slopes. Since $\hat{\boldsymbol{\beta}}$ being linear in t implies that also $\|\hat{\boldsymbol{\beta}}\|_1$ is linear in t, coordinate flow can be thought of as a dual formulation of the forward stagewise version LARS algorithm, providing $\hat{\boldsymbol{\beta}}$ as a function of t rather than of $\|\hat{\boldsymbol{\beta}}\|_1$.

Definition 11 (Coordinate Flow)**.**

- $\hat{\boldsymbol{\beta}}_{\text{cf}}(0) = \mathbf{0}$, $t_0 = 0$, $t_{i_{\max}}$ *is the time of convergence, i.e.* $\hat{\boldsymbol{\beta}}_{\text{cf}}(t_{i_{\max}}) = \hat{\boldsymbol{\beta}}^{\text{OLS}}$.
- *For* $0 < i < i_{\max}$*,*

$$\hat{\boldsymbol{\beta}}_{\text{cf}}(t) = \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i) + \frac{t-t_i}{1-\gamma}\boldsymbol{I}^i_{\text{cf}} \cdot \text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i))\right), \quad t \in [t_i, t_{i+1}), \tag{20}$$

where $\{\boldsymbol{I}^i_{\text{cf}}\}_{i=0}^{i_{\max}}$ *are constant diagonal matrices, with non-zero diagonal components given by*

$$\left(\boldsymbol{I}^i_{\text{cf}}\right)_{S^i_A, S^i_A} = \textit{diag}\left((\boldsymbol{B}_{:,1})^{-1} \odot \text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i))\right)_{S^i_A}\right),$$

where $\boldsymbol{B}$ *is a square matrix, stated in the construction (Section C.4.1), that depends on* $\text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i))\right)_{S^i_A}$ *and* $\hat{\boldsymbol{\Sigma}}_{S^i_A, S^i_A}$.

- $S_A^i := \left\{d : \left(\boldsymbol{I}_{\text{cf}}^i\right)_{dd} > 0\right\}$.
- *The times* $\{t_i\}_{i=0}^{i_{\max}}$*, when* $\boldsymbol{I}_{\text{cf}}^i$ *is updated are given by*

$$
\begin{aligned}
&t_0 = 0\\
&\Delta t_{i,d,1} = \frac{\left(\hat{\boldsymbol{\Sigma}}_{d,:} - \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)\left(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i)\right)}{\left(\hat{\boldsymbol{\Sigma}}_{d,:} - \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)\boldsymbol{I}_{\text{cf}}^i \cdot \text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i))\right)}\\
&\Delta t_{i,d,2} = \frac{\left(\hat{\boldsymbol{\Sigma}}_{d,:} + \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)\left(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i)\right)}{\left(\hat{\boldsymbol{\Sigma}}_{d,:} + \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)\boldsymbol{I}_{\text{cf}}^i \cdot \text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{cf}}(t_i))\right)}\\
&t_{i+1} = t_i + \min_{\substack{d\notin S_A^i,\ k=1,2\\ \Delta t_{i,d,k}>0}} \Delta t_{i,d,k}.
\end{aligned}
$$

C.3 Elastic Gradient Flow

When linear regression is solved using elastic gradient descent, $\hat{\boldsymbol{\beta}}$ is updated iteratively according to

$$
\begin{aligned}
\hat{\boldsymbol{\beta}}(t+\Delta t) =& \hat{\boldsymbol{\beta}}(t) + \gamma\left(\hat{\boldsymbol{\beta}}(t) - \hat{\boldsymbol{\beta}}(t-\Delta t)\right) - \Delta t \cdot \boldsymbol{I}_{\text{egd}}(\alpha, t)\left(\alpha \cdot \text{sign}(\boldsymbol{g}(t)) + (1-\alpha)\cdot \boldsymbol{g}(t)\right)\\
=& \hat{\boldsymbol{\beta}}(t) + \gamma\left(\hat{\boldsymbol{\beta}}(t) - \hat{\boldsymbol{\beta}}(t-\Delta t)\right)\\
&+ \Delta t \cdot \boldsymbol{I}_{\text{egd}}(\alpha, t)\left(\alpha \cdot \text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t))\right) + (1-\alpha)\cdot \hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t))\right).
\end{aligned}
\tag{21}
$$

Compared to coordinate descent, where only the parameter with maximum gradient is updated, this time the parameters with large enough, but not necessarily the largest, gradients are updated. Just as for coordinate descent and flow, we replace $\boldsymbol{I}_{\text{egd}}(\alpha, t)$ with $\boldsymbol{I}_{\text{egf}}(\alpha, t)$. Again, $\boldsymbol{I}_{\text{egf}}(\alpha, t)$ is a diagonal matrix with $\left(\boldsymbol{I}_{\text{egf}}(\alpha, t)\right)_{dd} \in [0, 1]$, but in contrast to $\boldsymbol{I}_{\text{cf}}(t)$ it is not piece-wise constant.

We define elastic gradient flow according to Definition 12, where details behind the definition are presented in Section C.4.2. The active and inactive sets of coordinate flow are now generalized to the free, coupled, and inactive sets for elastic gradient flow. Similar to coordinate flow, $\boldsymbol{I}_{\text{egf}}$ is recalculated at certain times, t_i. However, between the times of recalculation only the entries corresponding to the free and inactive sets, $(\boldsymbol{I}_{\text{egf}})_{S_F\cup S_0, S_F\cup S_0}$, remain constant, while the entries corresponding to the coupled set, $(\boldsymbol{I}_{\text{egf}})_{S_C,S_C}$, change with time on the interval $(0, 1)$.

Definition 12 (Elastic Gradient Flow)**.**

- $\hat{\boldsymbol{\beta}}_{\text{egf}}(0) = \boldsymbol{0}$*,* $t_0 = 0$*,* $t_{i_{\max}}$ *is the time of convergence, i.e.* $\hat{\boldsymbol{\beta}}_{\text{egf}}(t_{i_{\max}}) = \hat{\boldsymbol{\beta}}^{\text{OLS}}$*.*
- *For* $0 < i < i_{\max}$*,*

$$
\begin{aligned}
\hat{\boldsymbol{\beta}}_{\text{egf}}(t) =& \hat{\boldsymbol{\beta}}_{\text{egf}}(t_i) + \left((1-\alpha)\hat{\boldsymbol{\Sigma}}\right)^{-1}\left(\boldsymbol{I} - \exp\left(\boldsymbol{\Omega}^i(t_i, t)\right)\right)\\
&\cdot\left(\alpha \cdot \text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{egf}}(t_i))\right) + (1-\alpha)\cdot\hat{\boldsymbol{\Sigma}}\left(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{egf}}(t_i)\right)\right),\\
&t \in [t_i, t_{i+1}),
\end{aligned}
\tag{22}
$$

where

- $\boldsymbol{\Omega}^i(t_i, t)$ *is the Magnus expansion (Magnus, 1954) of* $-\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\boldsymbol{I}^i_{\text{egf}}(\alpha, t)$,
- $\left\{\boldsymbol{I}^i_{\text{egf}}(\alpha, t)\right\}_{i=0}^{i_{\max}}$ *are diagonal matrices with elements in* $[0, 1]$, *such that*
  - $\left(\boldsymbol{I}^i_{\text{egf}}\right)_{S^i_F, S^i_F}(\alpha, t) = \boldsymbol{I}$.
  - $\left(\boldsymbol{I}^i_{\text{egf}}\right)_{S^i_0, S^i_0}(\alpha, t) = \boldsymbol{0}$.
  - $\left(\boldsymbol{I}^i_{\text{egf}}\right)_{S^i_C, S^i_C}(\alpha, t)$ *is defined through its Taylor expansion:*

$$\left(\boldsymbol{I}^i_{\text{egf}}\right)_{S^i_C, S^i_C}(\alpha, t) := \sum_{k=0}^{\infty} \left(\left(\boldsymbol{I}^i_{\text{egf}}\right)_{S^i_C, S^i_C}\right)^{(k)}(t_i) \frac{(t - t_i)^k}{k!},$$

where

$$\left(\left(\boldsymbol{I}^i_{\text{egf}}\right)_{S^i_C, S^i_C}\right)^{(k)}(t_i) = \operatorname{diag}\left(\boldsymbol{A}^{-1}\boldsymbol{b}(k) \odot \frac{1}{\alpha \cdot \operatorname{sign}(\boldsymbol{g}(t_i)) + (1-\alpha) \cdot \boldsymbol{g}(t_i)}\right), \quad k = 0, 1, \ldots, \tag{23}$$

where matrix $\boldsymbol{A}$ *and vectors* $\boldsymbol{b}(k)$, *both stated in the construction (Section C.4.2), depend on* $\boldsymbol{g}(t_i)$, $\hat{\boldsymbol{\Sigma}}$ *and* $\left(\boldsymbol{I}^i_{\text{egf}}\right)^{(l)}(\alpha, t_i)$, $l < k$.

- $S^i_F(t) := \{d : (\boldsymbol{I}_{\text{egf}})_{dd}(t) = 1\}$. *The free set.*
- $S^i_0(t) := \{d : (\boldsymbol{I}_{\text{egf}})_{dd}(t) = 0\}$. *The inactive set.*
- $S^i_C(t) := \{d : (\boldsymbol{I}_{\text{egf}})_{dd}(t) \in (0, 1)\}$. *The coupled set.*

Remark 1: Note that Equation 22 is defined even when $\hat{\boldsymbol{\Sigma}}$ is not invertible. By Taylor expanding $\hat{\boldsymbol{\Sigma}}^{-1}\left(\boldsymbol{I} - \exp\left(\boldsymbol{\Omega}^i(t_i, t)\right)\right)$, it can be seen that $\hat{\boldsymbol{\Sigma}}^{-1}$ is canceled by (at least) one $\hat{\boldsymbol{\Sigma}}$ from $\left(\boldsymbol{I} - \exp\left(\boldsymbol{\Omega}^i(t_i, t)\right)\right)$.
Remark 2: Since implementing elastic gradient flow is quite computationally heavy, the formulation should be seen as a tool for providing a deeper understanding of elastic gradient descent, rather than as a substitute.
Remark 3: For coordinate flow, $\{\boldsymbol{I}_{\text{cf}}(\alpha, t)\}_{i=0}^{i_{\max}}$ and $\{t_i\}_{i=1}^{i_{\max}}$ could be calculated analytically. Due to the exponential function in Equation 22, this is not the case for elastic gradient flow. However, $\boldsymbol{I}_{\text{egf}}(\alpha, t)$ can be expressed by its Taylor expansion of arbitrary order, using the derivatives from Equation 23. The second order expansion of $\boldsymbol{\Omega}^i(t_i, t)$ is presented in Section C.5.

To calculate $\{t_i\}_{i=1}^{i_{\max}}$, the following criteria have to be evaluated numerically, selecting t_i as the one that occurs first.

1. $|g_d(t)| = \alpha \cdot |g_m(t)|$ for $d \in S_0$. A parameter leaves the inactive set.
2. $|g_d(t)| = \alpha \cdot |g_m(t)|$ for $d \in S_F$. A parameter leaves the free set.

3. $(\boldsymbol{I}_{\text{egf}})_{dd}(\alpha, t) \in \{0, 1\}$ for $d \in S_C$. A parameter leaves the coupled set.
4. $|g_d(t)| = |g_m(t)|$ for $d \in S_F$, $d \neq m$. The maximum gradient component changes.

Remark 4: Similar to how lasso and ridge regression are special cases of the elastic net, coordinate and gradient descent (flow) are special cases of elastic gradient descent (flow). Remembering that $\boldsymbol{I}_{\text{egd}}(0, t) = \boldsymbol{I}$, it is trivial that Equations 17 and 19 are special cases of Equation 21. The flow versions require slightly more work: When $\alpha = 0$, all variables belong to the free set at all times, which means that $\boldsymbol{I}_{\text{egf}}(0, t) = \boldsymbol{I}$ and there are no update times t_i. Furthermore, since $\hat{\boldsymbol{\Sigma}}\boldsymbol{I}_{\text{egf}}(0, t) = \hat{\boldsymbol{\Sigma}}$ is independent of t, $\exp(-\boldsymbol{\Omega}(0, t))$ reduces to $\exp\left(-\frac{1}{1-\gamma}\int_0^t \hat{\boldsymbol{\Sigma}} dt\right) = \exp\left(-\frac{t}{1-\gamma}\hat{\boldsymbol{\Sigma}}\right)$ (see Section C.5 for details), which commutes with $\hat{\boldsymbol{\Sigma}}^{-1}$, and for $\hat{\boldsymbol{\beta}}_{\boldsymbol{0}} = \boldsymbol{0}$ Equation 22 simplifies to Equation 18. When $\alpha = 1$, all parameters are either in the inactive or in the coupled set; except when only one parameter is non-zero, then the free set consists of that single parameter and the coupled set is empty. That is, with $S_A := S_F \cup S_C$ the definition of the t_i's for coordinate flow and elastic gradient flow coincide. Letting $(1 - \alpha) \to 0$ and using $\lim_{x\to 0} \frac{\boldsymbol{I} - \exp(-x(\boldsymbol{A} + x\boldsymbol{B}))}{x} = \boldsymbol{A}$, Equation 22 simplifies to Equation 20.

In gradient flow (and descent), all parameters may update freely according to their gradient values, while both Definitions 11 and 12 split the parameters into groups, with different update rules. For coordinate flow (and descent), some parameters are not allowed to update at all, while others update, but in a coupled fashion, making sure that the gradients are always equal. Elastic gradient flow (and descent) combines all of these three update properties. The free set contains the indices of the parameters for which $|g_d| > \alpha \cdot |g_m|$, which are updated according to their gradient value. The inactive set contains the indices of the parameters for which $|g_d| < \alpha \cdot |g_m|$, which are not updated. The coupled set contains the indices of the parameters for which $|g_d| = \alpha \cdot |g_m|$. In the discrete case this corresponds to $(\boldsymbol{I}_{\text{egd}})_{dd}$ fluctuating between 0 and 1, and $|g_d|$ oscillating around $\alpha \cdot |g_m|$, while in the continuous case $(\boldsymbol{I}_{\text{egf}}) \in (0, 1)$ and $|g_d| = \alpha \cdot |g_m|$. Since $(\boldsymbol{I}_{\text{egf}})_{dd}$, and hence the update speed of these parameters, depends on the value of $|g_m|$, we refer to them as coupled. These three sets are illustrated in Section 4.2.

C.4 Construction of Coordinate and Elastic Gradient Flow

In this section, we present the details behind the definitions of coordinate and elastic gradient flow. The following notation is used: Uppercase boldface letters are used for matrices and lowercase boldface letters for vectors. Slices of matrices and vectors are marked by subscripts, which might be either a single index, a set of indices, or a colon that denotes an entire row/column. Complements are denoted with a minus sign. We will give two examples for $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and $S_A = \{3, 5\}$: $\boldsymbol{A}_{S_A,-1}$, denotes a $2 \times (n-1)$ matrix consisting of rows 3 and 5, and all but the first column of matrix $\boldsymbol{A}$. $\boldsymbol{A}_{:,-S_A}$ denotes an $m \times (n-2)$ matrix consisting of all rows and all columns except 3 and 5. Time derivatives of order k are denoted interchangeably with $\frac{\partial^k}{\partial t^k}(\cdot)$ and $(\cdot)^{(k)}$ and $\odot$ denotes element-wise multiplication.

C.4.1 Construction of Coordinate Flow

If at some time interval, $[t_1, t_2]$, the magnitudes of two or more gradient components are all close to the maximum gradient value, the index m (where $m = \text{argmax}_d |g_d|$ and $(\boldsymbol{I}_{\text{cd}})_{mm} = 1$,

as defined above) might alternate very frequently between these elements (or equivalently, the one in $\boldsymbol{I}_{\text{cd}}$ frequently changes position along the diagonal) during the interval; we denote the corresponding set of indices $S_A(t_1, t_2)$. If we could look at an even finer time scale than Δt, we would observe the same behavior on the sub-interval $[t, t + \Delta t]$. To consider a finer time scale than Δt, we split the time step Δt into K sub-steps rewriting Equation 19 as

$$\begin{aligned}\hat{\boldsymbol{\beta}}(t + \Delta t) =& \hat{\boldsymbol{\beta}}(t) + \gamma \left(\hat{\boldsymbol{\beta}}(t) - \hat{\boldsymbol{\beta}}(t - \Delta t)\right) \\ &+ \sum_{k=0}^{K-1} \frac{\Delta t}{K} \cdot \boldsymbol{I}_{\text{cf}} \left(t + k\frac{\Delta t}{K}\right) \text{sign}\left(\hat{\boldsymbol{\Sigma}} \left(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}} \left(t + k\frac{\Delta t}{K}\right)\right)\right).\end{aligned} \tag{24}$$

If $d \in S_A(t, t + \Delta t)$, then $|g_d(\tau)| > 0$ for $\tau \in [t, t + \Delta t]$, i.e. g_d does not change sign, which means that $\text{sign}(g_d)$ remains constant on the interval. If, on the other hand, $d \notin S_A(t, t + \Delta t)$, then g_d might change sign on the interval, but then $(\boldsymbol{I}_{\text{cd}})_{dd} = 0$, and the value of $\text{sign}(g_d)$ is not considered. This means that Equation 24 can be written as

$$\hat{\boldsymbol{\beta}}(t + \Delta t) = \hat{\boldsymbol{\beta}}(t) + \gamma \left(\hat{\boldsymbol{\beta}}(t) - \hat{\boldsymbol{\beta}}(t - \Delta t)\right) + \frac{\Delta t}{K} \sum_{k=0}^{K-1} \boldsymbol{I}_{\text{cf}} \left(t + k\frac{\Delta t}{K}\right) \text{sign}(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t))).$$

Rearranging and letting first $K \to \infty$, then $\Delta t \to 0$, assuming that the limits exist, we obtain

$$(1 - \gamma) \cdot \frac{\partial \hat{\boldsymbol{\beta}}(t)}{\partial t} = \boldsymbol{I}_{\text{cf}}^{\infty}(t) \text{sign}(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t))),$$

where

$$\boldsymbol{I}_{\text{cf}}^{\infty}(t) := \lim_{\Delta t \to 0} \lim_{K \to \infty} \frac{1}{K} \sum_{k=0}^{K-1} \boldsymbol{I}_{\text{cd}} \left(t + k\frac{\Delta t}{K}\right)$$

is the limit of averages of matrices where one diagonal element equals one and the remaining elements equal zero. Since it is not obvious that this limit exists, we instead define $\boldsymbol{I}_{\text{cf}}$ as an average of matrices of type $\boldsymbol{I}_{\text{cd}}$, i.e.,

- $(\boldsymbol{I}_{\text{cf}})_{dd} \in [0, 1]$
- $\sum_d (\boldsymbol{I}_{\text{cf}})_{dd} = 1$
- $(\boldsymbol{I}_{\text{cf}})_{dd}(t) > 0 \iff d \in S_A(t)$

and obtain

$$(1 - \gamma) \cdot \frac{\partial \hat{\boldsymbol{\beta}}(t)}{\partial t} = \boldsymbol{I}_{\text{cf}}(t) \text{sign}(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t))). \tag{25}$$

Repeating the reasoning just after Equation 24, we can say that if $d \in S_A^i := S_A(t_i, t_{i+1})$, then g_d does not change sign for $t \in [t_i, t_{i+1})$ and

$$-\text{sign}(g_d(t)) = -\text{sign}(g_d(t_i)) = \text{sign}(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t_i)))_d =: (\boldsymbol{s}^i)_d.$$

If, on the other hand, $d \notin S_A^i$, then $\left(\boldsymbol{I}_{\text{cf}}^i\right)_{dd}(t) = 0$ and the value of $(\boldsymbol{s}^i)_d$ is not considered. This means that Equation 25 can be written as

$$(1 - \gamma) \cdot \frac{\partial \hat{\boldsymbol{\beta}}(t)}{\partial t} = \boldsymbol{I}_{\text{cf}}(t) \boldsymbol{s}^i, \quad t \in [t_i, t_{i+1}),$$

to which the solution is given by

$$\hat{\boldsymbol{\beta}}(t) = \hat{\boldsymbol{\beta}}(t_i) + \frac{1}{1-\gamma}\int_{t_i}^{t} \boldsymbol{I}_{\text{cf}}(\tau) d\tau \boldsymbol{s}^i.$$

We now show that for $t \in [t_i, t_{i+1})$, $\int_{t_i}^{t} \boldsymbol{I}_{\text{cf}}(\tau)d\tau = (t-t_i)\boldsymbol{I}^i_{\text{cf}}$, and calculate $\boldsymbol{I}^i_{\text{cf}}$.

Assume $S^i_A = \{m_1, m_2, \dots, m_{p_m}\}$ at $t = t_i$, which implies $|g_{m_1}(t_i)| = |g_{m_2}(t_i)| = \dots = |g_{m_{p_m}}(t_i)|$. If $d \notin S^i_A$ at time t, then $\left(\boldsymbol{I}^i_{\text{cf}}\right)_{dd}(t) = 0$, so we focus on the sub-matrix $(\boldsymbol{I}_{\text{cf}})_{S^i_A, S^i_A}(t)$, which is a $p_m \times p_m$ matrix containing only the rows and columns for which $d \in S^i_A$ at time t_i.

We want to construct $(\boldsymbol{I}_{\text{cf}})_{S^i_A, S^i_A}(t)$ such that the elements of S^i_A do not change for $t \in [t_i, t_{i+1})$, which implies $|g_{m_1}(t)| = |g_{m_2}(t)| = \dots = |g_{m_{p_m}}(t)|$. Let's start with $|g_{m_1}(t)| = |g_{m_2}(t)|$.

$$\begin{aligned}
0 =& |g_{m_2}(t)| - |g_{m_1}(t)| = |\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t))|_{m_2} - |\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t))|_{m_1} \\
=& s^i_{m_2} \cdot (\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t)))_{m_2} - s^i_{m_1} \cdot (\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t)))_{m_1} \\
=& \underbrace{s^i_{m_2} \cdot (\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t_i)))_{m_2}}_{=|g_{m_2}(t_i)|} - s^i_{m_2} \cdot \left(\hat{\boldsymbol{\Sigma}} \frac{1}{1-\gamma}\int_{t_i}^{t} \boldsymbol{I}_{\text{cf}}(\tau)d\tau \boldsymbol{s}^i\right)_{m_2} \\
& - \underbrace{s^i_{m_1} \cdot (\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t_i)))_{m_1}}_{=|g_{m_1}(t_i)|} + s^i_{m_1} \cdot \left(\hat{\boldsymbol{\Sigma}} \frac{1}{1-\gamma}\int_{t_i}^{t} \boldsymbol{I}_{\text{cf}}(\tau)d\tau \boldsymbol{s}^i\right)_{m_1} \\
=& \frac{1}{1-\gamma}\left(s^i_{m_1} \cdot \hat{\boldsymbol{\Sigma}}_{m_1,:} - s^i_{m_2} \cdot \hat{\boldsymbol{\Sigma}}_{m_2,:}\right)\int_{t_i}^{t} \boldsymbol{I}_{\text{cf}}(\tau)d\tau \boldsymbol{s}^i \\
\overset{(a)}{=}& \frac{1}{1-\gamma}\left(s^i_{m_1} \cdot \hat{\boldsymbol{\Sigma}}_{m_1,S^i_A} - s^i_{m_2} \cdot \hat{\boldsymbol{\Sigma}}_{m_2,S^i_A}\right)\int_{t_i}^{t} (\boldsymbol{I}_{\text{cf}})_{S^i_A,S^i_A}(\tau)d\tau \boldsymbol{s}^i_{S^i_A} \\
\iff 0 =& \left(s^i_{m_1} \cdot \hat{\boldsymbol{\Sigma}}_{m_1,S^i_A} - s^i_{m_2} \cdot \hat{\boldsymbol{\Sigma}}_{m_2,S^i_A}\right)\int_{t_i}^{t} (\boldsymbol{I}_{\text{cf}})_{S^i_A,S^i_A}(\tau)d\tau \boldsymbol{s}^i_{S^i_A},
\end{aligned}$$

where (a) follows from the fact that $\left(\boldsymbol{I}^i_{\text{cf}}\right)_{dd}(t) = 0$ for $d \notin S^i_A$. Repeating the same calculations for all combinations of indices in S^i_A gives us $p_m - 1$ independent equations. Together with

$$\begin{aligned}
1 =& \sum_{m \in S^i_A} (\boldsymbol{I}_{\text{cf}})_{mm}(t) \iff t - t_i = \sum_{m \in S^i_A} \int_{t_i}^{t} (\boldsymbol{I}_{\text{cf}})_{mm}(\tau)d\tau = \mathbf{1}^\top \int_{t_i}^{t} (\boldsymbol{I}_{\text{cf}})_{S^i_A,S^i_A}(\tau)d\tau \mathbf{1} \\
=& (\boldsymbol{s}^i_{S^i_A})^\top \int_{t_i}^{t} (\boldsymbol{I}_{\text{cf}})_{S^i_A,S^i_A}(\tau)d\tau \boldsymbol{s}^i_{S^i_A},
\end{aligned}$$

we obtain the following system of linear equations

$$\underbrace{\begin{bmatrix} —(\boldsymbol{s}^i_{S^i_A})^\top— \\ —\left(s^i_{m_1}\cdot\hat{\boldsymbol{\Sigma}}_{m_1,S^i_A} - s^i_{m_2}\cdot\hat{\boldsymbol{\Sigma}}_{m_2,S^i_A}\right)— \\ —\left(s^i_{m_2}\cdot\hat{\boldsymbol{\Sigma}}_{m_2,S^i_A} - s^i_{m_3}\cdot\hat{\boldsymbol{\Sigma}}_{m_3,S^i_A}\right)— \\ \vdots \\ —\left(s^i_{m_{p_m-1}}\cdot\hat{\boldsymbol{\Sigma}}_{m_{p_m-1},S^i_A} - s^i_{m_{p_m}}\cdot\hat{\boldsymbol{\Sigma}}_{m_{p_m},S^i_A}\right)— \end{bmatrix}}_{=:\boldsymbol{B}} \begin{bmatrix} | \\ \int_{t_i}^t (\boldsymbol{I}_{\text{cf}})_{S^i_A,S^i_A}(\tau)d\tau\,\boldsymbol{s}^i_{S^i_A} \\ | \end{bmatrix} = \underbrace{\begin{bmatrix} t-t_i \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{=:\boldsymbol{b}}.$$

Solving this system, we obtain

$$\begin{aligned} &\int_{t_i}^t (\boldsymbol{I}_{\text{cf}})_{S^i_A,S^i_A}(\tau)d\tau\,\boldsymbol{s}^i_{S^i_A} = \boldsymbol{B}^{-1}\boldsymbol{b} = (t-t_i)\cdot(\boldsymbol{B}_{:,1})^{-1} \\ \iff &\int_{t_i}^t (\boldsymbol{I}_{\text{cf}})_{S^i_A,S^i_A}(\tau)d\tau = (t-t_i)\cdot\operatorname{diag}\left((\boldsymbol{B}_{:,1})^{-1}\odot\boldsymbol{s}^i_{S^i_A}\right) =: (t-t_i)\left(\boldsymbol{I}^i_{\text{cf}}\right)_{S^i_A,S^i_A}, \end{aligned}$$

where $\odot$ denotes element-wise multiplication.

For some combinations of $\hat{\boldsymbol{\Sigma}}$ and $\hat{\boldsymbol{\beta}}^{\text{OLS}} - \boldsymbol{\beta}(t)$, we might obtain a solution where $\left(\boldsymbol{I}^i_{\text{cf}}\right)_{dd} < 0$ for some d. Since this is obviously not feasible (since it would correspond to a negative time step), in these cases, we remove the corresponding parameter from S^i_A, which implies $\left(\boldsymbol{I}^i_{\text{cf}}\right)_{dd} = 0$. Then $\left(\boldsymbol{I}^i_{\text{cf}}\right)_{S^i_A,S^i_A}$ is recalculated for the new S^i_A. If $\left(\boldsymbol{I}^i_{\text{cf}}\right)_{dd} < 0$ for more than one parameter simultaneously, only the parameter with the largest (absolute valued) negative value is removed. This procedure is repeated until all $\left(\boldsymbol{I}^i_{\text{cf}}\right)_{dd} \in [0,1]$.

What is left to do is to compute the times when some new d enters S^i_A, i.e. when $|g_d(t)| = |g_m(t)|$ for $m \in S^i_A$, $d \notin S^i_A$. Since the absolute gradient value is identical for all $m \in S^i_A$, we use m_1. First assume $\operatorname{sign}(g_d(t)) = \operatorname{sign}(g_{m_1}(t))$. Then

$$\begin{aligned} 0 &= g_{m_1}(t) - g_d(t) = -\hat{\boldsymbol{\Sigma}}_{m_1,:}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t)) + \hat{\boldsymbol{\Sigma}}_{d,:}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t)) \\ &= -\hat{\boldsymbol{\Sigma}}_{m_1,:}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t_i)) + (t-t_i)\hat{\boldsymbol{\Sigma}}_{m_1,:}\boldsymbol{I}^i_{\text{cf}}\boldsymbol{s}^i + \hat{\boldsymbol{\Sigma}}_{d,:}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t_i)) - (t-t_i)\hat{\boldsymbol{\Sigma}}_{d,:}\boldsymbol{I}^i_{\text{cf}}\boldsymbol{s}^i \\ &= \left(\hat{\boldsymbol{\Sigma}}_{d,:} - \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t_i)) - (t-t_i)\left(\hat{\boldsymbol{\Sigma}}_{d,:} - \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)\boldsymbol{I}^i_{\text{cf}}\boldsymbol{s}^i \\ \iff t &= t_i + \underbrace{\frac{\left(\hat{\boldsymbol{\Sigma}}_{d,:} - \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t_i))}{\left(\hat{\boldsymbol{\Sigma}}_{d,:} - \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)\boldsymbol{I}^i_{\text{cf}}\boldsymbol{s}^i}}_{=:\Delta t_{i,d,1}}. \end{aligned}$$

Repeating the same calculations when $\operatorname{sign}(g_d(t)) = -\operatorname{sign}(g_{m_1}(t))$ results in

$$t = t_i + \underbrace{\frac{\left(\hat{\boldsymbol{\Sigma}}_{d,:} + \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}(t_i))}{\left(\hat{\boldsymbol{\Sigma}}_{d,:} + \hat{\boldsymbol{\Sigma}}_{m_1,:}\right)\boldsymbol{I}^i_{\text{cf}}\boldsymbol{s}^i}}_{=:\Delta t_{i,d,2}}.$$

Putting this together, we obtain

$$t_{i+1} = t_i + \min_{\substack{d\notin S^i_A,\ k=1,2 \\ \Delta t_{i,d,k}>0}} \Delta t_{i,d,k}.$$

C.4.2 Construction of Elastic Gradient Flow

This time the differential equation of interest becomes

$$(1-\gamma)\cdot\frac{\partial\hat{\boldsymbol{\beta}}(t)}{\partial t}=\boldsymbol{I}_{\text{egf}}(\alpha,t)\left(\alpha\cdot\text{sign}\left(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}}-\hat{\boldsymbol{\beta}}(t))\right)+(1-\alpha)\cdot\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}}-\hat{\boldsymbol{\beta}}(t))\right). \quad (26)$$

Since the vector $\text{sign}(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}}-\hat{\boldsymbol{\beta}}(t)))$ is constant for $d\notin S_0$, Equation 26 simplifies to

$$(1-\gamma)\cdot\frac{\partial\hat{\boldsymbol{\beta}}_{\text{egf}}(t)}{\partial t}=\boldsymbol{I}^i_{\text{egf}}(\alpha,t)\left(\alpha\cdot\boldsymbol{s}^i+(1-\alpha)\cdot\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}}-\hat{\boldsymbol{\beta}}_{\text{egf}}(t))\right),\quad t\in[t_i,t_{i+1}), \quad (27)$$

where $\boldsymbol{s}^i:=-\text{sign}(g_d(t))=-\text{sign}(g_d(t_i))=\text{sign}(\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}}-\hat{\boldsymbol{\beta}}_{\text{egf}}(t_i)))$.

Let

$$\boldsymbol{\eta}(t):=\alpha\cdot\boldsymbol{s}^i+(1-\alpha)\cdot\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}}-\hat{\boldsymbol{\beta}}_{\text{egf}}(t)).$$

Then

$$\hat{\boldsymbol{\beta}}_{\text{egf}}(t)=-\left((1-\alpha)\hat{\boldsymbol{\Sigma}}\right)^{-1}\left(\boldsymbol{\eta}(t)-\left(\alpha\cdot\boldsymbol{s}^i+(1-\alpha)\cdot\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}}-\hat{\boldsymbol{\beta}}_{\text{egf}}(t_i))\right)\right)+\hat{\boldsymbol{\beta}}_{\text{egf}}(t_i) \quad (28)$$

and

$$\frac{\partial\boldsymbol{\eta}(t)}{\partial t}=-(1-\alpha)\hat{\boldsymbol{\Sigma}}\frac{\partial\hat{\boldsymbol{\beta}}_{\text{egf}}(t)}{\partial t}\iff\frac{\partial\hat{\boldsymbol{\beta}}_{\text{egf}}(t)}{\partial t}=-\left((1-\alpha)\hat{\boldsymbol{\Sigma}}\right)^{-1}\frac{\partial\boldsymbol{\eta}(t)}{\partial t}.$$

We can now rewrite Equation 27 in terms of $\boldsymbol{\eta}$:

$$\frac{\partial\boldsymbol{\eta}(t)}{\partial t}=-\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\boldsymbol{I}^i_{\text{egf}}(\alpha,t)\boldsymbol{\eta}(t),\quad t\in[t_i,t_{i+1}),$$

which gives us

$$\boldsymbol{\eta}(t)=\exp\left(\boldsymbol{\Omega}^i(t_i,t)\right)\boldsymbol{\eta}(t_i),\quad\in[t_i,t_{i+1}), \quad (29)$$

where exp is the matrix exponential and $\boldsymbol{\Omega}^i(t_i,t)$ is the Magnus expansion (Magnus, 1954) of $-\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\boldsymbol{I}^i_{\text{egf}}(\alpha,t)$. Plugging Equation 29 into Equation 28, we obtain

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{\text{egf}}(t)=&\hat{\boldsymbol{\beta}}_{\text{egf}}(t_i)\\&+\left((1-\alpha)\hat{\boldsymbol{\Sigma}}\right)^{-1}\left(\boldsymbol{I}-\exp\left(\boldsymbol{\Omega}^i(t_i,t)\right)\right)\cdot\left(\alpha\cdot\boldsymbol{s}^i+(1-\alpha)\cdot\hat{\boldsymbol{\Sigma}}\left(\hat{\boldsymbol{\beta}}^{\text{OLS}}-\hat{\boldsymbol{\beta}}_{\text{egf}}(t_i)\right)\right),\\&t\in[t_i,t_{i+1}).\end{aligned}$$

We now calculate the time derivatives of $\left(\boldsymbol{I}^i_{\text{egf}}\right)_{S^i_C,S^i_C}(t_i)$. Let $m:=\text{argmax}_d|g_d(t_i)|$. If $c\in S^i_C$, then $(\boldsymbol{I}^i_{\text{egf}})_{cc}(t)\notin\{0,1\}$ and $|g_c(t)|=\alpha|g_m(t)|$. This means that for $t\in[t_i,t_{i+1})$

$$\left|\frac{g_c(t)}{g_m(t)}\right|=\alpha, \quad (30)$$

which we want to solve for $(\boldsymbol{I}^i_{\text{egf}})_{cc}(\alpha,t):=(\boldsymbol{I}_{\text{egf}})_{cc}(\alpha,t)$, $t\in[t_i,t_{i+1})$.

Since $\left|\frac{g_c(t_i)}{g_m(t_i)}\right|=\alpha$, Equation 30 holds if

$$\left(\left|\frac{g_c(t_i)}{g_m(t_i)}\right|\right)^{(k+1)}=0,\ k=0,1,\dots, \quad (31)$$

because requiring that the derivative is 0 for t_i implies that the function remains constant for $t > t_i$, at least if the derivative remains zero, which is why we require the second derivative to be 0, and so on. Using Lemma 13, we obtain

$$\begin{aligned} &\left(\left|\frac{g_c(t_i)}{g_m(t_i)}\right|\right)^{(k+1)} = 0 \\ &\qquad\qquad \Longleftrightarrow \\ g_c^{(k+1)}(t_i) \cdot g_m(t_i) - g_c(t_i) \cdot g_m^{(k+1)}(t_i) &= -g_m^2(t_i) \cdot \operatorname{sign}\left(\frac{g_c(t_i)}{g_m(t_i)}\right) \cdot \mathcal{O}\left(\left(\frac{g_c(t_i)}{g_m(t_i)}\right)^{(k)}\right). \end{aligned} \tag{32}$$

If we solve Equation 31 for k's in increasing order, when solving for $k+1$,

$$\left(\left|\frac{g_c(t_i)}{g_m(t_i)}\right|\right)^{(l)} = 0,\ l = 1, 2, \dots k$$

and Equation 32 simplifies to

$$\left(\left|\frac{g_c(t_i)}{g_m(t_i)}\right|\right)^{(k+1)} = 0 \iff g_c^{(k+1)}(t_i) \cdot g_m(t_i) - g_c(t_i) \cdot g_m^{(k+1)}(t_i) = 0. \tag{33}$$

Using Lemma 14, abbreviating $\boldsymbol{\zeta}^i := (\alpha \cdot \operatorname{sign}(\boldsymbol{g}(t_i)) + (1-\alpha) \cdot \boldsymbol{g}(t_i))$ and $\boldsymbol{c}^{k-1} := \mathcal{O}((\boldsymbol{I}_{\text{egf}}^i)^{(k-1)}(t_i)$, we obtain

$$g_d^{(k+1)}(t_i) = \frac{1}{1-\gamma}\left(-\hat{\boldsymbol{\Sigma}}_{d,:}(\boldsymbol{I}_{\text{egf}}^i)^{(k)}(t_i)\boldsymbol{\zeta}^i + c_d^{k-1}\right), \tag{34}$$

where $\boldsymbol{c}^{k-1}$ can be calculated by first setting $(\boldsymbol{I}_{\text{egf}}^i)^{(k)}(t_i) = \mathbf{0}$, to remove the first term in Equation 34, and then evaluating $\boldsymbol{g}^{(k+1)}(t_i)$ using Equations 16 and 22:

$$\begin{aligned} \boldsymbol{c}^{k-1} &= (1-\gamma)\boldsymbol{g}^{(k+1)}(t_i) = (1-\gamma)\hat{\boldsymbol{\Sigma}}\boldsymbol{\beta}^{(k+1)}(t_i) \\ &= -(1-\gamma)\hat{\boldsymbol{\Sigma}}\left.\frac{d^{k+1}\exp\left(\boldsymbol{\Omega}^i(t_i,t)\right)}{dt^k}\right|_{t=t_i}\left(\frac{\alpha}{1-\alpha}\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{s}^i + \hat{\boldsymbol{\beta}}^{\text{OLS}} - \boldsymbol{\beta}(t_i)\right), \end{aligned}$$

where the derivative of the matrix exponential can be calculated using Lemma 15.

Now, inserting Equation 34 into Equation 33, we obtain.

$$
\begin{aligned}
&\frac{1}{1-\gamma}\left(-\hat{\boldsymbol{\Sigma}}_{c,:}(\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i + c_c^{k-1}\right)\cdot g_m(t_i) \\
&- g_c(t_i)\cdot\frac{1}{1-\gamma}\left(-\hat{\boldsymbol{\Sigma}}_{m,:}(\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i + c_m^{k-1}\right) = 0 \\
\iff &(-\hat{\boldsymbol{\Sigma}}_{c,:}(\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i) + c_c^{k-1})\cdot g_m(t_i) - g_c(t_i)\cdot(-\hat{\boldsymbol{\Sigma}}_{m,:}(\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i) + c_m^{k-1}) = 0 \\
\iff &c_m^{k-1}\cdot g_c(t_i) - c_c^{k-1}\cdot g_m(t_i) = \left(g_c(t_i)\hat{\boldsymbol{\Sigma}}_{m,:} - g_m(t_i)\cdot\hat{\boldsymbol{\Sigma}}_{c,:}\right)(\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i \\
&\overset{(a)}{=} \begin{bmatrix} g_c(t_i)\hat{\boldsymbol{\Sigma}}_{m,c} - g_m(t_i)\cdot\hat{\boldsymbol{\Sigma}}_{c,c} & g_c(t_i)\hat{\boldsymbol{\Sigma}}_{m,-c} - g_m(t_i)\cdot\hat{\boldsymbol{\Sigma}}_{c,-c}\end{bmatrix}\begin{bmatrix}\left((\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i\right)_c \\ \left((\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i\right)_{-c}\end{bmatrix} \\
\iff &= \underbrace{c_m^{k-1}\cdot g_c(t_i) - g_m(t_i)\cdot c_c^{k-1} - \left(g_c(t_i)\hat{\boldsymbol{\Sigma}}_{m,-c} - g_m(t_i)\cdot\hat{\boldsymbol{\Sigma}}_{c,-c}\right)\left((\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i\right)_{-c}}_{=:b} \\
&= \underbrace{\left(g_c(t_i)\hat{\boldsymbol{\Sigma}}_{m,c} - g_m(t_i)\cdot\hat{\boldsymbol{\Sigma}}_{c,c}\right)}_{=:a}\underbrace{\left((\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i\right)_c}_{=(\boldsymbol{I}^i_{\text{egf}})^{(k)}_{cc}(t_i)\zeta^i_c} \\
\iff &(\boldsymbol{I}^i_{\text{egf}})^{(k)}_{cc}(t_i) = \frac{b}{a\cdot\zeta^i_c},
\end{aligned}
$$

where in (a), the columns of $\hat{\boldsymbol{\Sigma}}$ and the rows of $(\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i$ are split into the two sets c and $-c$.

Writing this as a system of linear equations, we obtain

$$
\begin{aligned}
\boldsymbol{A} :=& \left(\boldsymbol{g}_{S^i_C}(t_i)\hat{\boldsymbol{\Sigma}}_{m,S^i_C} - g_m(t_i)\cdot\hat{\boldsymbol{\Sigma}}_{S^i_C,S^i_C}\right) \\
\boldsymbol{b} :=& c_m^{k-1}\cdot\boldsymbol{g}_{S^i_C}(t_i) - g_m(t_i)\cdot\boldsymbol{c}^{k-1}_{S^i_C} \\
&- \left(\boldsymbol{g}_{S^i_C}(t_i)\hat{\boldsymbol{\Sigma}}_{m,S^i_F\cup S^i_0} - g_m(t_i)\cdot\hat{\boldsymbol{\Sigma}}_{S^i_C,S^i_F\cup S^i_0}\right)\left((\boldsymbol{I}^i_{\text{egf}})^{(k)}(t_i)\boldsymbol{\zeta}^i\right)_{S^i_F\cup S^i_0} \\
\left(\boldsymbol{I}^i_{\text{egf}}\right)^{(k)}_{S^i_C,S^i_C}(t_i) =& \text{diag}(\boldsymbol{A}^{-1}\boldsymbol{b}/\boldsymbol{\zeta}^i_{S^i_C}),
\end{aligned}
$$

where the division is element-wise.

For $k=0$, we have $\boldsymbol{c}^{k-1} = (\boldsymbol{I}^i_{\text{egf}})^{(-1)}(t_i) = \boldsymbol{0}$, $(\boldsymbol{I}^i_{\text{egf}})_{S^i_0,S^i_0}(t_i) = \boldsymbol{0}$ and $(\boldsymbol{I}^i_{\text{egf}})_{S^i_F,S^i_F}(t_i) = \boldsymbol{I}$, which means that $\boldsymbol{b}$ simplifies to

$$
\boldsymbol{b}_{k=0} = \left(g_m(t_i)\cdot\hat{\boldsymbol{\Sigma}}_{S^i_C,S^i_F} - \boldsymbol{g}_{S^i_C}(t_i)\hat{\boldsymbol{\Sigma}}_{m,S^i_F}\right)\boldsymbol{\zeta}^i_{S^i_F}.
$$

For $k\geq 1$, we have $(\boldsymbol{I}^i_{\text{egf}})^{(k)}_{S^i_F\cup S^i_0,S^i_F\cup S^i_0}(t_i) = \boldsymbol{0}$, which means that $\boldsymbol{b}$ simplifies to

$$
\boldsymbol{b}_{k\geq 1} = c_m^{k-1}\cdot\boldsymbol{g}_{S^i_C}(t_i) - g_m(t_i)\cdot\boldsymbol{c}^{k-1}_{S^i_C}.
$$

Similar to for coordinate flow, $(\boldsymbol{I}_{\text{egf}})_{cc}(\alpha,t)\in[0,1]$ might not always hold and the corresponding modification, in this case, amounts to: If at any time t_i, $(\boldsymbol{I}^i_{\text{egf}})_{cc}\leq 0$ for

some parameter in S_C^i, then that parameter is moved from S_C^i to S_0^i; if $(\boldsymbol{I}_{\text{egf}}^i)_{dd} \geq 1$ for some parameter in S_C^i, then that parameter is moved from S_C^i to S_F^i. If $(\boldsymbol{I}_{\text{egf}}^i)_{dd} \notin [0,1]$ for more than one parameter simultaneously, only the parameter that deviates most from $[0,1]$ is removed. This procedure is repeated until all $(\boldsymbol{I}_{\text{egf}}^i)_{dd} \in [0,1]$, which means that $(\boldsymbol{I}_{\text{egf}}^i)_{cc} \in (0,1)$ for $c \in S_C^i$.

C.5 Magnus Expansion

According to Magnus (1954),

$$\begin{aligned}\boldsymbol{\Omega}^i(t_i,t) = &\int_{t_i}^t \boldsymbol{A}(\tau_1)d\tau_1 + \frac{1}{2}\int_{t_i}^t\int_{t_i}^{\tau_1} [\boldsymbol{A}(\tau_1),\boldsymbol{A}(\tau_2)]\, d\tau_2 d\tau_1 \\ &+ \frac{1}{4}\int_{t_i}^t\int_{t_i}^{\tau_1}\int_{t_i}^{\tau_2} [\boldsymbol{A}(\tau_1),[\boldsymbol{A}(\tau_2),\boldsymbol{A}(\tau_3)]]\, d\tau_3 d\tau_2 d\tau_1 + \dots,\end{aligned} \tag{35}$$

where the commutator is defined according to $[\boldsymbol{A},\boldsymbol{B}] := \boldsymbol{A}\boldsymbol{B} - \boldsymbol{B}\boldsymbol{A}$. For $\boldsymbol{A}(t) = -(1-\alpha)\boldsymbol{I}_{\text{egf}}^i(\alpha,t)\hat{\boldsymbol{\Sigma}}$, the first two terms in Equation 35, together with its time derivatives, are calculated below.

C.5.1 First Term

$$\begin{aligned}\boldsymbol{\Omega}_1^i(t_i,t) &= \int_{t_i}^t -\frac{1-\alpha}{\hat{\boldsymbol{\Sigma}}}1 - \gamma\boldsymbol{I}_{\text{egf}}^i(\alpha,\tau)d\tau \overset{(a)}{=} -\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\int_{t_i}^t\sum_{l=0}^{\infty}\frac{(\tau-t_i)^l}{l!}(\boldsymbol{I}_{\text{egf}}^i)^{(l)}(\alpha,t_i)d\tau \\ &= -\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\sum_{l=0}^{\infty}\frac{(t-t_i)^{l+1}}{(l+1)!}(\boldsymbol{I}_{\text{egf}}^i)^{(l)}(\alpha,t_i). \\ (\boldsymbol{\Omega}_1^i)^{(k)}(t_i,t) &= -\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\sum_{l=0}^{\infty}\frac{(t-t_i)^{l+1-k}}{(l+1-k)!}(\boldsymbol{I}_{\text{egf}}^i)^{(l)}(\alpha,t_i) \\ (\boldsymbol{\Omega}_1^i)^{(k)}(t_i,t_i) &= -\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}_{\text{egf}}^i)^{(k-1)}(\alpha,t_i),\end{aligned}$$

where in (a), we use the Taylor expansion of $\boldsymbol{I}_{\text{egf}}^i(\alpha,t)$ around t_i, and the last step uses that $(t_i-t_i)^{l+1-k} = \begin{cases}1 \text{ if } l = k-1\\ 0 \text{ else.}\end{cases}$

C.5.2 Second Term

$$\begin{aligned}\boldsymbol{\Omega}_2^i(t_i,t) &= \frac{1}{2}\int_{t_i}^t\int_{t_i}^{\tau_1}\left[-\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\boldsymbol{I}_{\text{egf}}^i(\alpha,\tau_1),\ -\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\boldsymbol{I}_{\text{egf}}^i(\alpha,\tau_2)\right]d\tau_2 d\tau_1 \\ &= \frac{1}{2}\left(\frac{1-\alpha}{1-\gamma}\right)^2\int_{t_i}^t\int_{t_i}^{\tau_1}\left[\hat{\boldsymbol{\Sigma}}\boldsymbol{I}_{\text{egf}}^i(\alpha,\tau_1),\ \hat{\boldsymbol{\Sigma}}\boldsymbol{I}_{\text{egf}}^i(\alpha,\tau_2)\right]d\tau_2 d\tau_1.\end{aligned}$$

Focusing on the commutator, and writing $\boldsymbol{I}^i_{\text{egf}}(\alpha, t)$ as its Taylor expansion, we obtain

$$\begin{aligned}
&\left[\hat{\boldsymbol{\Sigma}}\boldsymbol{I}^i_{\text{egf}}(\alpha,\tau_1),\ \hat{\boldsymbol{\Sigma}}\boldsymbol{I}^i_{\text{egf}}(\alpha,\tau_2)\right]\\
=&\left[\hat{\boldsymbol{\Sigma}}\sum_{l_1=0}^{\infty}\frac{(\tau_1-t_i)^{l_1}}{l_1!}(\boldsymbol{I}^i_{\text{egf}})^{(l_1)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}\sum_{l_2=0}^{\infty}\frac{(\tau_2-t_i)^{l_2}}{l_2!}(\boldsymbol{I}^i_{\text{egf}})^{(l_2)}(\alpha,t_i)\right]\\
=&\sum_{l_1=0}^{\infty}\sum_{l_2=0}^{\infty}\frac{(\tau_1-t_i)^{l_1}}{l_1!}\frac{(\tau_2-t_i)^{l_2}}{l_2!}\left[\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_1)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_2)}(\alpha,t_i)\right]\\
\overset{(a)}{=}&\sum_{l_1=1}^{\infty}\sum_{l_2=0}^{l_1-1}\frac{1}{l_1!l_2!}\left((\tau_1-t_i)^{l_1}(\tau_2-t_i)^{l_2}-(\tau_2-t_i)^{l_2}(\tau_1-t_i)^{l_1}\right)\\
&\cdot\left[\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_1)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_2)}(\alpha,t_i)\right],
\end{aligned}$$

where in (a), we use

$$\begin{aligned}
&[\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l)}(\alpha,t_i)]=0\\
&[\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_2)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_1)}(\alpha,t_i)]=-[\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_1)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_2)}(\alpha,t_i)].
\end{aligned}$$

We now solve the integral with respect to τ_1 and τ_2,

$$\begin{aligned}
&\int_{t_i}^{t}\int_{t_i}^{\tau_1}\left((\tau_1-t_i)^{l_1}(\tau_2-t_i)^{l_2}-(\tau_2-t_i)^{l_2}(\tau_1-t_i)^{l_1}\right)d\tau_2 d\tau_1\\
&\qquad=\frac{l_1-l_2}{(l_1+1)!(l_2+1)!(l_1+l_2+2)}(t-t_i)^{l_1+l_2+2},
\end{aligned}$$

and putting it together, we obtain

$$\begin{aligned}
&\boldsymbol{\Omega}^i_2(t_i,t)\\
=&\frac{1}{2}\left(\frac{1-\alpha}{1-\gamma}\right)^2\sum_{l_1=1}^{\infty}\sum_{l_2=0}^{l_1-1}\frac{(l_1-l_2)(t-t_i)^{l_1+l_2+2}}{(l_1+1)!(l_2+1)!(l_1+l_2+2)}\left[\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_1)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_2)}(\alpha,t_i)\right]\\
=&\frac{1}{2}\left(\frac{1-\alpha}{1-\gamma}\right)^2\sum_{l_1=2}^{\infty}\sum_{l_2=1}^{l_1-1}\frac{(l_1-l_2)(t-t_i)^{l_1+l_2}}{l_1!l_2!(l_1+l_2)}\left[\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_1-1)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_2-1)}(\alpha,t_i)\right].
\end{aligned}$$

Differentiating k times with respect to t yields

$$\begin{aligned}
(\boldsymbol{\Omega}^i_2)^{(k)}(t_i,t)=&\frac{1}{2}\left(\frac{1-\alpha}{1-\gamma}\right)^2\sum_{l_1=2}^{\infty}\sum_{l_2=1}^{l_1-1}\frac{(l_1-l_2)(l_1+l_2)!\cdot(t-t_i)^{l_1+l_2-k}}{l_1!l_2!(l_1+l_2)(l_1+l_2-k)!}\\
&\cdot\left[\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_1-1)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_2-1)}(\alpha,t_i)\right]\\
(\boldsymbol{\Omega}^i_2)^{(k)}(t_i,t_i)=&\frac{1}{2}\left(\frac{1-\alpha}{1-\gamma}\right)^2\sum_{l_2=1}^{\lfloor\frac{k-1}{2}\rfloor}\frac{(k-2l_2)(k-1)!}{l_2!(k-l_2)!}\\
&\cdot\left[\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(k-l_2-1)}(\alpha,t_i),\ \hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\text{egf}})^{(l_2-1)}(\alpha,t_i)\right],
\end{aligned}$$

where the last step uses that the only surviving term in the first sum is when $l_1 + l_2 - k = 0 \iff l_1 = k - l_2$.

Appendix D. Proofs

Proof of Proposition 2
Since L is assumed to be strongly convex, with Hessian bounded by M,

$$
\begin{aligned}
&L(\hat{\boldsymbol{\beta}} - \Delta t \cdot \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}}) = L(\hat{\boldsymbol{\beta}}) - \Delta t \cdot \nabla L(\hat{\boldsymbol{\beta}})^\top \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}} + \frac{\Delta t^2}{2}\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}}^\top \nabla^2 L(\hat{\boldsymbol{\beta}})\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}} \\
\leq & L(\hat{\boldsymbol{\beta}}) - \Delta t \cdot \nabla L(\hat{\boldsymbol{\beta}})^\top \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}} + \frac{M\Delta t^2}{2}\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}}^\top \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}} \\
= & L(\hat{\boldsymbol{\beta}}) - \Delta t \cdot \boldsymbol{g}^\top \boldsymbol{I}_{\text{egd}} \cdot (\alpha \cdot \text{sign}(\boldsymbol{g}) + (1-\alpha)\boldsymbol{g}) \\
& + \frac{M\Delta t^2}{2}(\alpha \cdot \text{sign}(\boldsymbol{g}) + (1-\alpha)\boldsymbol{g})^\top \underbrace{\boldsymbol{I}_{\text{egd}}^\top \boldsymbol{I}_{\text{egd}}}_{=\boldsymbol{I}_{\text{egd}}} \cdot (\alpha \cdot \text{sign}(\boldsymbol{g}) + (1-\alpha)\boldsymbol{g}). \\
= & L(\hat{\boldsymbol{\beta}}) - \Delta t \cdot \left(\alpha \cdot \underbrace{\boldsymbol{g}^\top \boldsymbol{I}_{\text{egd}} \text{sign}(\boldsymbol{g})}_{=\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1} + (1-\alpha) \cdot \underbrace{\boldsymbol{g}^\top \boldsymbol{I}_{\text{egd}} \boldsymbol{g}}_{=\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2} \right) \\
& + \frac{M\Delta t^2}{2} \left(\alpha^2 \cdot \underbrace{\text{sign}(\boldsymbol{g})\boldsymbol{I}_{\text{egd}} \text{sign}(\boldsymbol{g})}_{=\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_0} + 2\alpha(1-\alpha) \cdot \underbrace{\boldsymbol{g}^\top \boldsymbol{I}_{\text{egd}} \text{sign}(\boldsymbol{g})}_{=\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1} + (1-\alpha)^2 \cdot \underbrace{\boldsymbol{g}^\top \boldsymbol{I}_{\text{egd}} \boldsymbol{g}}_{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2} \right).
\end{aligned}
$$

Using the inequalities

$$
\begin{aligned}
\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1 &\geq \frac{1}{g_{\max}} \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 \\
\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_0 &\leq \frac{1}{g_{\min}^2} \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 \\
\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1 &\leq \frac{1}{g_{\min}} \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2,
\end{aligned}
$$

we obtain

$$
\begin{aligned}
& L(\hat{\boldsymbol{\beta}} - \Delta t \cdot \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}}) - L(\hat{\boldsymbol{\beta}}) \leq -\Delta t \cdot \left(\alpha \cdot \frac{1}{g_{\max}} \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 + (1-\alpha) \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 \right) \\
& + \frac{M\Delta t^2}{2} \left(\alpha^2 \cdot \frac{1}{g_{\min}^2} \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 + 2\alpha(1-\alpha) \cdot \frac{1}{g_{\min}} \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 + (1-\alpha)^2 \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 \right). \\
= & -\Delta t \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 \cdot \left(\frac{\alpha}{g_{\max}} + (1-\alpha) - \frac{M\Delta t}{2} \left(\frac{\alpha^2}{g_{\min}^2} + \frac{2\alpha(1-\alpha)}{g_{\min}} + (1-\alpha)^2 \right) \right). \\
= & -\Delta t \cdot \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 \cdot \left(\frac{\alpha}{g_{\max}} + (1-\alpha) - \frac{M\Delta t}{2} \left(\frac{\alpha}{g_{\min}} + (1-\alpha) \right)^2 \right).
\end{aligned}
$$

Thus, $L(\hat{\boldsymbol{\beta}} - \Delta t \cdot \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd}}) - L(\hat{\boldsymbol{\beta}}) \leq 0$ if

$$\begin{aligned}&\frac{\alpha}{g_{\max}}+(1-\alpha)-\frac{M\Delta t}{2}\left(\frac{\alpha}{g_{\min}}+(1-\alpha)\right)^2\geq 0\\ \iff &\Delta t\leq \frac{2}{M}\cdot\frac{\alpha g_{\max}^{-1}+(1-\alpha)}{(\alpha g_{\min}^{-1}+(1-\alpha))^2}=\frac{2}{M}\cdot\frac{g_{\min}^2}{g_{\max}}\cdot\frac{\alpha+(1-\alpha)g_{\max}}{(\alpha+(1-\alpha)g_{\min})^2}.\end{aligned} \tag{36}$$

Using $g_{\min}\geq\alpha\cdot g_{\max}$ and $g_{\min}>0$, we can remove $g_{\min}$ from Equation 36, lower the expression. From the two bounds on $g_{\min}$ we obtain

$$\frac{\alpha}{g_{\min}}\geq\frac{1_{\alpha>0}}{g_{\max}},$$

where $g_{\min}\geq\alpha$ is used for $\alpha>0$, and $g_{\min}>0$ is used for $\alpha=0$. Thus, we obtain

$$\Delta t\leq\frac{2}{M}\cdot\frac{\alpha g_{\max}^{-1}+(1-\alpha)}{\left(1_{\alpha>0}\cdot g_{\max}^{-1}+(1-\alpha)\right)^2}=\frac{2}{M}\cdot g_{\max}\cdot\frac{\alpha+(1-\alpha)g_{\max}}{(1_{\alpha>0}+(1-\alpha)g_{\max})^2}.$$

■

Proof of Lemma 3
The ridge estimate is defined as

$$\hat{\boldsymbol{\beta}}(\lambda):=(\boldsymbol{X}^\top\boldsymbol{X}+n\lambda\boldsymbol{I})^{-1}\boldsymbol{X}^\top\boldsymbol{y},$$

which can be rewritten as

$$\begin{aligned}(\boldsymbol{X}^\top\boldsymbol{X}+n\lambda\boldsymbol{I})^{-1}\boldsymbol{X}^\top\boldsymbol{y}&\overset{(a)}{=}(\underbrace{\boldsymbol{X}^\top\boldsymbol{X}}_{n\hat{\boldsymbol{\Sigma}}}+n\lambda\boldsymbol{I})^{-1}\underbrace{(\boldsymbol{X}^\top\boldsymbol{X})}_{n\hat{\boldsymbol{\Sigma}}}\underbrace{(\boldsymbol{X}^\top\boldsymbol{X})^+\boldsymbol{X}^\top\boldsymbol{y}}_{\hat{\boldsymbol{\beta}}^{\text{OLS}}}\\ &=\frac{1}{n}(\hat{\boldsymbol{\Sigma}}+\lambda\boldsymbol{I})^{-1}n\hat{\boldsymbol{\Sigma}}\hat{\boldsymbol{\beta}}^{\text{OLS}}=(\hat{\boldsymbol{\Sigma}}+\lambda\boldsymbol{I})^{-1}(\hat{\boldsymbol{\Sigma}}+\lambda\boldsymbol{I}-\lambda\boldsymbol{I})\hat{\boldsymbol{\beta}}^{\text{OLS}}\\ &=(\underbrace{(\hat{\boldsymbol{\Sigma}}+\lambda\boldsymbol{I})^{-1}(\hat{\boldsymbol{\Sigma}}+\lambda\boldsymbol{I})}_{\boldsymbol{I}}-\lambda(\hat{\boldsymbol{\Sigma}}+\lambda\boldsymbol{I})^{-1})\hat{\boldsymbol{\beta}}^{\text{OLS}}\\ &=\left(\boldsymbol{I}-\left(\boldsymbol{I}+\frac{1}{\lambda}\hat{\boldsymbol{\Sigma}}\right)^{-1}\right)\hat{\boldsymbol{\beta}}^{\text{OLS}},\end{aligned}$$

where (a) follows from $\boldsymbol{X}^\top=(\boldsymbol{X}^\top\boldsymbol{X})(\boldsymbol{X}^\top\boldsymbol{X})^+\boldsymbol{X}^\top$. ■

Proof of Proposition 4
We first note that when $\hat{\boldsymbol{\Sigma}}=\boldsymbol{I}$, $\boldsymbol{I}_{\text{egf}}^i(\alpha,t_1)\boldsymbol{I}$ and $\boldsymbol{I}_{\text{egf}}^i(\alpha,t_2)\boldsymbol{I}$ are diagonal and thus commute, which means that $\boldsymbol{\Omega}^i(t_i,t)$ reduces to $-\frac{1-\alpha}{1-\gamma}\int_{t_i}^t\boldsymbol{I}_{\text{egf}}^i(\alpha,\tau)d\tau$.

Since the data is uncorrelated and $\hat{\boldsymbol{\beta}}(t_0)=\boldsymbol{0}$, $|\hat{\beta}_d^{\text{egf}}(t)|\leq|\hat{\beta}_d^{\text{OLS}}|$ and for $\hat{\beta}_d^{\text{egf}}(t)\neq 0$, $\text{sign}(\hat{\beta}_d^{\text{egf}}(t))=\text{sign}(\hat{\beta}_d^{\text{OLS}})=\text{sign}(\hat{\beta}_d^{\text{OLS}}-\hat{\beta}_d^{\text{egf}}(t))$. This means that Equation 22 can be written as

$$\begin{aligned}\hat{\beta}_d^{\text{egf}}(t)=&\text{sign}(\hat{\beta}_d^{\text{OLS}})\cdot\min\left(|\hat{\beta}_d^{\text{egf}}(t_i)|+\frac{1}{1-\alpha}\left(1-\exp\left(-\frac{1-\alpha}{1-\gamma}\int_{t_i}^t(\boldsymbol{I}_{\text{egf}}^i)_{dd}(\alpha,\tau)d\tau\right)\right)\right.\\ &\left.\cdot\left(\alpha+(1-\alpha)\left(|\hat{\beta}_d^{\text{OLS}}|-|\hat{\beta}_d^{\text{egf}}(t_i)|\right)\right),|\hat{\beta}_d^{\text{OLS}}|\right),\end{aligned}$$

where the minimum, which is included to assure that $|\hat{\beta}_d^{\text{egf}}(t)| \leq |\hat{\beta}_d^{\text{OLS}}|$, becomes active once the OLS solution is reached.

We now compare this the closed form solution of the elastic net,

$$\hat{\beta}_d^{\text{en}}(\lambda) = \text{sign}(\hat{\beta}_d^{\text{OLS}}) \frac{\max(|\hat{\beta}_d^{\text{OLS}}| - \alpha\lambda, 0)}{1 + (1-\alpha)\lambda} = \text{sign}(\hat{\beta}_d^{\text{OLS}}) \frac{\left(|\hat{\beta}_d^{\text{OLS}}| - \min(\alpha\lambda, |\hat{\beta}_d^{\text{OLS}}|)\right)}{1 + (1-\alpha)\lambda}.$$

Requiring $\hat{\beta}_d^{\text{en}}(\lambda) = \hat{\beta}_d^{\text{egf}}(t)$, we obtain

$$\hat{\beta}_d^{\text{en}}(\lambda) = \hat{\beta}_d^{\text{egf}}(t) \iff \lambda_d = \max\left(\frac{|\hat{\beta}_d^{\text{OLS}}| - |\hat{\beta}_d^{\text{egf}}(t_i)| - \boldsymbol{v}}{\alpha + (1-\alpha)\left(|\hat{\beta}_d^{\text{egf}}(t_i)| + \boldsymbol{v}\right)}, 0 \right),$$

where

$$\boldsymbol{v} = \frac{1}{1-\alpha}\left(1 - \exp\left(-\frac{1-\alpha}{1-\gamma}\int_{t_i}^{t} (\boldsymbol{I}^i_{\text{egf}})_{dd}(\alpha,\tau)d\tau\right)\right)\left(\alpha + (1-\alpha)\left(|\hat{\beta}_d^{\text{OLS}}| - |\hat{\beta}_d^{\text{egf}}(t_i)|\right)\right),$$

and where the equivalence is tedious but straightforward to show.

Letting $(1-\alpha) \to 0$, using $\lim_{x\to 0} \frac{1-e^{-ax}}{x} = a$, we obtain

$$\lambda_d = \max\left(|\hat{\beta}_d^{\text{OLS}}| - |\hat{\beta}_d^{\text{cf}}(t_i)| - \frac{t - t_i}{1-\gamma}(\boldsymbol{I}^i_{\text{cf}})_{dd}, 0\right).$$

To calculate $\text{sign}\left(\frac{\partial \lambda_d(t)}{\partial t}\right)$, we note first that

$$\begin{aligned}\frac{\partial \boldsymbol{v}(t)}{\partial t} =& \frac{1}{1-\gamma}(\boldsymbol{I}^i_{\text{egf}})_{dd}(\alpha,t)\exp\left(-\frac{1-\alpha}{1-\gamma}\int_{t_i}^{t}(\boldsymbol{I}^i_{\text{egf}})_{dd}(\alpha,\tau)d\tau\right) \\ &\cdot \left(\alpha + (1-\alpha)\underbrace{\left(|\hat{\beta}_d^{\text{OLS}}| - |\hat{\beta}_d^{\text{egf}}(t_i)|\right)}_{\geq 0}\right) \geq 0,\end{aligned}$$

which implies

$$\begin{aligned}\frac{\partial \lambda_d(t)}{\partial t} =& \underbrace{\frac{\partial(\max(0,\lambda_d(t)))}{\partial \lambda_d}}_{=:f_1(t)\geq 0} \cdot \underbrace{\frac{1}{\left(\alpha + (1-\alpha)\left(|\hat{\beta}_d^{\text{egf}}(t_i)| + \boldsymbol{v}(t)\right)\right)^2}}_{=:f_2(t)\geq 0} \\ &\cdot \left(\frac{\partial}{\partial t}\left(|\hat{\beta}_d^{\text{OLS}}| - |\hat{\beta}_d^{\text{egf}}(t_i)| - \boldsymbol{v}(t)\right) \cdot \left(\alpha + (1-\alpha)\left(|\hat{\beta}_d^{\text{egf}}(t_i)| + \boldsymbol{v}(t)\right)\right)\right. \\ &\left. - \left(|\hat{\beta}_d^{\text{OLS}}| - |\hat{\beta}_d^{\text{egf}}(t_i)| - \boldsymbol{v}(t)\right) \cdot \frac{\partial}{\partial t}\left(\alpha + (1-\alpha)\left(|\hat{\beta}_d^{\text{egf}}(t_i)| + \boldsymbol{v}(t)\right)\right)\right) \\ =& f_1(t)\cdot f_2(t)\cdot\left(-\frac{\partial \boldsymbol{v}(t)}{\partial t}\cdot\cancel{\left(\alpha + (1-\alpha)\left(|\hat{\beta}_d^{\text{egf}}(t_i)| + \boldsymbol{v}(t)\right)\right)}\right. \\ &\left. - \left(|\hat{\beta}_d^{\text{OLS}}| - \cancel{|\hat{\beta}_d^{\text{egf}}(t_i)| - \boldsymbol{v}(t)}\right)\cdot(1-\alpha)\frac{\partial \boldsymbol{v}(t)}{\partial t}\right) \\ =& -\underbrace{f_1(t)}_{\geq 0}\cdot\underbrace{f_2(t)}_{\geq 0}\cdot\underbrace{\frac{\partial \boldsymbol{v}(t)}{\partial t}}_{\geq 0}\cdot\underbrace{\left(\alpha + (1-\alpha)\cdot|\hat{\beta}_d^{\text{OLS}}|\right)}_{\geq 0} \leq 0.\end{aligned}$$

■

Proof of Proposition 6
For $c_\alpha \geq 0$,

$$\begin{aligned}\left\|c_\alpha \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}\right\|_1 =& c_\alpha \left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}\right\|_1 = c_\alpha \|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1 \left(\frac{\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1} + \frac{1-\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}\right) \\ =& c_\alpha \left(\alpha + (1-\alpha)\frac{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}\right) = c_\alpha \left(\alpha + (1-\alpha)\sqrt{q_1}\right).\end{aligned}$$

$$\begin{aligned}\left\|c_\alpha \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}\right\|_2^2 =& c_\alpha^2 \|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2^2 \left(\frac{\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1} + \frac{1-\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}\right)^2 \\ =& c_\alpha^2 \left(\alpha \frac{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1} + (1-\alpha)\right)^2 = c_\alpha^2 \left(\frac{\alpha}{\sqrt{q_1}} + 1 - \alpha\right)^2.\end{aligned}$$

Solving

$$\begin{aligned}1 =& \alpha \left\|c_\alpha \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}\right\|_1 + (1-\alpha)\left\|c_\alpha \boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}\right\|_2^2 \\ =& c_\alpha \alpha \left(\alpha + (1-\alpha)\cdot\sqrt{q_1}\right) + c_\alpha^2 (1-\alpha)\left(\alpha \cdot \frac{1}{\sqrt{q_1}} + 1 - \alpha\right)^2\end{aligned}$$

for c_α, the non-negative root is

$$c_\alpha = \frac{\sqrt{q_1(\alpha^2 q_1 + 4(1-\alpha))} - \alpha q_1}{2(1-\alpha)(\sqrt{q_1}(1-\alpha) + \alpha)}.$$

■

Proof of Proposition 7

$$\begin{aligned}\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}\right\|_1 =& \|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1 \left(\frac{\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1} + \frac{1-\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}\right) \\ =& \left(\alpha + (1-\alpha)\frac{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}\right).\end{aligned}$$

$$\begin{aligned}\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}\right\|_2^2 =& \|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2^2 \left(\frac{\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1} + \frac{1-\alpha}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}\right)^2 \\ =& \left(\alpha \frac{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1} + (1-\alpha)\right)^2.\end{aligned}$$

$$\begin{aligned}&\alpha \left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}\right\|_1 + (1-\alpha)\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,sd}}\right\|_2^2 \\ =& \alpha \left(\alpha + (1-\alpha)\cdot\frac{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}\right) + (1-\alpha)\left(\alpha \cdot \frac{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1} + 1 - \alpha\right)^2.\end{aligned}$$

Using the inequalities $1 \leq \frac{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2} \leq \sqrt{p_1}$ and the equalities $\alpha + (1-\alpha)^3 = 1 - \alpha(1-\alpha)(2-\alpha)$ and $\alpha^2 + 1 - \alpha = 1 - \alpha(1-\alpha)$, we obtain the lower bound

$$1 - \alpha(1-\alpha)\left(2 - \alpha - \frac{\alpha}{p_1} - \frac{2(1-\alpha)}{\sqrt{p_1}}\right) \overset{(a)}{\geq} 1 - \alpha(1-\alpha)(2-\alpha) \cdot \left(1 - \frac{1}{p_1}\right),$$

where (a) follows from $\sqrt{p_1} \leq p_1$ for $p_1 \geq 1$, and the upper bound

$$1 + \alpha(1-\alpha) \cdot (\sqrt{p_1} - 1).$$

Noting that $\alpha(1-\alpha)(2-\alpha) < 0.39$ for $\alpha \in [0,1]$, and $\alpha(1-\alpha) \leq \frac{1}{4}$ completes the proof. ■

Proof of Proposition 9
For $c_{\alpha,\Delta t} \geq 0$,

$$\begin{aligned}
\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs,c}}\right\|_1 =& \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1 \left(\frac{\alpha \cdot c_{\alpha,\Delta t} \cdot \Delta t}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1} + \frac{(1-\alpha)\sqrt{c_{\alpha,\Delta t} \cdot \Delta t}}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\right) \\
=& \left(\alpha \cdot c_{\alpha,\Delta t} \cdot \Delta t + (1-\alpha)\sqrt{c_{\alpha,\Delta t} \cdot \Delta t}\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\right) \\
=& \left(\alpha \cdot c_{\alpha,\Delta t} \cdot \Delta t + (1-\alpha)\sqrt{c_{\alpha,\Delta t} \cdot \Delta t \cdot q_1}\right).
\end{aligned}$$

$$\begin{aligned}
\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs,c}}\right\|_2^2 =& \|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2 \left(\frac{\alpha \cdot c_{\alpha,\Delta t} \cdot \Delta t}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1} + \frac{(1-\alpha)\sqrt{c_{\alpha,\Delta t} \cdot \Delta t}}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\right)^2 \\
=& \left(\alpha \cdot c_{\alpha,\Delta t} \cdot \Delta t\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1} + (1-\alpha)\sqrt{c_{\alpha,\Delta t} \cdot \Delta t}\right)^2 \\
=& \left(\alpha \cdot c_{\alpha,\Delta t} \cdot \Delta t \cdot \frac{1}{\sqrt{q_1}} + (1-\alpha)\sqrt{c_{\alpha,\Delta t} \cdot \Delta t}\right)^2.
\end{aligned}$$

Solving

$$\begin{aligned}
\Delta t =& \alpha \left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs,c}}\right\|_1 + (1-\alpha)\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs,c}}\right\|_2^2 \\
& \alpha \cdot \left(\alpha \cdot c_{\alpha,\Delta t} \cdot \Delta t + (1-\alpha)\sqrt{c_{\alpha,\Delta t} \cdot \Delta t \cdot q_1}\right) \\
& + (1-\alpha)\left(\alpha \cdot c_{\alpha,\Delta t} \cdot \Delta t \cdot \frac{1}{\sqrt{q_1}} + (1-\alpha)\sqrt{c_{\alpha,\Delta t} \cdot \Delta t}\right)^2
\end{aligned}$$

for $c_{\alpha,\Delta t}$, we obtain

$$c_{\alpha,\Delta t} = \left(\frac{\sqrt{2\alpha\sqrt{q_1(\alpha^2 q_1 + 4\Delta t(1-\alpha))} + q_1((1-\alpha)^3 - 2\alpha^2)} - (1-\alpha)\sqrt{q_1(1-\alpha)}}{\alpha\sqrt{4\Delta t(1-\alpha)}}\right)^2.$$

■

Proof of Proposition 10

$$\begin{aligned}\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs}}\right\|_1 =&\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1\left(\frac{\alpha\cdot\Delta t}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}+\frac{(1-\alpha)\sqrt{\Delta t}}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\right)\\ =&\left(\alpha\cdot\Delta t+(1-\alpha)\sqrt{\Delta t}\frac{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2}\right).\\ \left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs}}\right\|_2^2 =&\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2^2\left(\frac{\alpha\cdot\Delta t}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}+\frac{(1-\alpha)\sqrt{\Delta t}}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\right)^2\\ =&\left(\alpha\cdot\Delta t\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}+(1-\alpha)\sqrt{\Delta t}\right)^2.\end{aligned}$$

$$\begin{aligned}&\alpha\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs}}\right\|_1+(1-\alpha)\left\|\boldsymbol{\Delta}\hat{\boldsymbol{\beta}}_{\text{egd,gs}}\right\|_2^2\\ =&\alpha\left(\alpha\cdot\Delta t+(1-\alpha)\sqrt{\Delta t}\cdot\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\right)+(1-\alpha)\left(\alpha\cdot\Delta t\cdot\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1}+(1-\alpha)\sqrt{\Delta t}\right)^2.\end{aligned}$$

Using the inequalities $\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\leq\sqrt{p_1}$, $\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}\leq 1$ we obtain upper bound

$$\begin{aligned}&\alpha\cdot\left(\alpha\Delta t+(1-\alpha)\sqrt{\Delta t}\cdot\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\right)+(1-\alpha)\cdot\left(\alpha\Delta t\cdot\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}+(1-\alpha)\sqrt{\Delta t}\right)^2\\ \leq&\alpha\cdot\left(\alpha\Delta t+(1-\alpha)\sqrt{\Delta t}\cdot\sqrt{p_1}\right)+(1-\alpha)\cdot\left(\alpha\Delta t+(1-\alpha)\sqrt{\Delta t}\right)^2\\ \overset{(a)}{=}&\Delta t+\alpha(1-\alpha)\left((\alpha-3)\Delta t+\sqrt{\Delta t}\sqrt{p_1}+\alpha(\Delta t)^2+2(1-\alpha)\Delta t\sqrt{\Delta t}\right)\\ \overset{(b)}{\leq}&\Delta t+\alpha(1-\alpha)\left((\alpha-3)\Delta t+\sqrt{\Delta t}\sqrt{p_1}+\alpha\Delta t+2(1-\alpha)\Delta t\right)\\ =&\Delta t\left(1+\alpha(1-\alpha)\left(\sqrt{\frac{p_1}{\Delta t}}-1\right)\right),\end{aligned}$$

where in (a), we use $\alpha^2+(1-\alpha)^3=1+\alpha(1-\alpha)(\alpha-3)$ and (b) uses $(\Delta t)^2\leq\Delta t$, $\sqrt{\Delta t}\leq 1$ for $\Delta t\leq 1$.

The lower bound is obtain by using $\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\geq 1$, $\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}\geq\frac{1}{\sqrt{p_1}}$:

$$\begin{aligned}&\alpha\cdot\left(\alpha\Delta t+(1-\alpha)\sqrt{\Delta t}\cdot\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}\right)+(1-\alpha)\cdot\left(\alpha\Delta t\cdot\frac{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_2}{\|\boldsymbol{I}_{\text{egd}}\boldsymbol{g}\|_1}+(1-\alpha)\sqrt{\Delta t}\right)^2\\ \geq&\alpha\cdot\left(\alpha\Delta t+(1-\alpha)\sqrt{\Delta t}\right)+(1-\alpha)\cdot\left(\frac{\alpha\Delta t}{\sqrt{p_1}}+(1-\alpha)\sqrt{\Delta t}\right)^2\\ \overset{(a)}{=}&\Delta t+\alpha(1-\alpha)\left((\alpha-3)\Delta t+\sqrt{\Delta t}+\alpha\frac{(\Delta t)^2}{p_1}+2(1-\alpha)\frac{\Delta t\sqrt{\Delta t}}{\sqrt{p_1}}\right)\\ \overset{(b)}{\geq}&\Delta t+\alpha(1-\alpha)\left((\alpha-3)\Delta t+\Delta t+\alpha\frac{(\Delta t)^2}{p_1}+2(1-\alpha)\frac{(\Delta t)^2}{p_1}\right)\\ =&\Delta t\left(1-\alpha(1-\alpha)(2-\alpha)\left(1-\frac{\Delta t}{p_1}\right)\right),\end{aligned}$$

where in (a), we use $\alpha^2+(1-\alpha)^3 = 1+\alpha(1-\alpha)(\alpha-3)$ and (b) uses $\sqrt{\Delta t} \geq \Delta t$ for $\Delta t \leq 1$, and $\sqrt{p_1} \leq p_1$ for $p_1 \geq 1$.

Noting that $\alpha(1-\alpha)(2-\alpha) < 0.39$ for $\alpha \in [0,1]$, and $\alpha(1-\alpha) \leq \frac{1}{4}$ completes the proof. ■

Proof of Lemma 8
We assume without loss of generality that $\boldsymbol{g}$ is sorted, so that $|g_1| \geq |g_2| \geq \ldots$.

$$\frac{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_2^2}{\|\boldsymbol{I}_{\text{egd,sd}}\boldsymbol{g}\|_1} = \frac{g_1^2 \cdot \left(1+\sum_{i=2}^{p_1}\left(\frac{g_i}{g_1}\right)^2\right)}{|g_1| \cdot \left(1+\sum_{i=2}^{p_1}\left|\frac{g_i}{g_1}\right|\right)},$$

Since $(g_i/g_1)^2 \leq |g_i/g_1|$ and $|g_i| \geq |g_{i+1}|$, $\left(1+\sum_{i=2}^{p_1}\left(\frac{g_i}{g_1}\right)^2\right) / \left(1+\sum_{i=2}^{p_1}\left|\frac{g_i}{g_1}\right|\right)$ is a decreasing function of p_1. ■

Lemma 13.

$$\left(\left|\frac{f}{g}\right|\right)^{(k)} = \operatorname{sign}\left(\frac{f}{g}\right) \cdot \left(\frac{f^{(k)} \cdot g - f \cdot g^{(k)}}{g^2}\right) + \mathcal{O}\left(\left(\frac{f}{g}\right)^{(k-1)}\right),$$

where $\mathcal{O}\left(\left(\frac{f}{g}\right)^{(k-1)}\right)$ *denotes derivatives of* $\left(\frac{f}{g}\right)$ *of orders strictly lower than* k.

Proof
We first show that

$$\left(\frac{f}{g}\right)^{(k)} = \frac{f^{(k)}g - fg^{(k)}}{g^2} - \sum_{i=1}^{k-1}\binom{k}{i}\left(\frac{f}{g}\right)^{(k-i)}\frac{g^{(i)}}{g}: \tag{37}$$

$$\begin{aligned}
&\frac{f^{(k)}g - fg^{(k)}}{g^2} - \sum_{i=1}^{k-1}\binom{k}{i}\left(\frac{f}{g}\right)^{(k-i)}\frac{g^{(i)}}{g} = \frac{1}{g}\left(f^{(k)} - \left(\frac{f}{g}\right)g^{(k)} - \sum_{i=1}^{k-1}\binom{k}{i}\left(\frac{f}{g}\right)^{(k-i)}g^{(i)}\right)\\
=&\frac{1}{g}\left(f^{(k)} - \left(\sum_{i=0}^{k}\binom{k}{i}\left(\frac{f}{g}\right)^{(k-i)}g^{(i)} - \left(\frac{f}{g}\right)^{(k)}g\right)\right)\\
\overset{(a)}{=}&\frac{1}{g}\left(f^{(k)} - \left(\left(\frac{f}{g}g\right)^{(k)} - \left(\frac{f}{g}\right)^{(k)}g\right)\right)\\
=&\frac{1}{g}\left(f^{(k)} - f^{(k)} + \left(\frac{f}{g}\right)^{(k)}g\right) = \left(\frac{f}{g}\right)^{(k)},
\end{aligned}$$

where (a) follows from the general Leibniz rule, $(fg)^{(k)} = \sum_{i=0}^{k}\binom{k}{i}f^{(k-i)}g^{(i)}$. Now, according to the chain rule for higher order derivatives, Faà di Bruno's formula,

$$\frac{df(g(x))}{dx^k} = f'(g(x)) \cdot g^{(k)}(x) + \mathcal{O}(g^{(k-1)}),$$

where $\mathcal{O}(g^{(k-1)})$ denotes terms with derivatives of g of orders strictly lower than k. Applying this, we obtain

$$\left(\left|\frac{f}{g}\right|\right)^{(k)} = \operatorname{sign}\left(\frac{f}{g}\right)\cdot\left(\frac{f}{g}\right)^{(k)} + \mathcal{O}\left(\left(\frac{f}{g}\right)^{(k-1)}\right).$$

Applying Equation 37 completes the proof. ■

Lemma 14.
With $\boldsymbol{g}(t) = -\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} - \boldsymbol{\beta}(t))$ *for* $\boldsymbol{\beta}(t)$ *given by Equation 22,*

$$\boldsymbol{g}^{(k)}(t_i) = \frac{1}{1-\gamma}\left(-\hat{\boldsymbol{\Sigma}}(\boldsymbol{I}^i_{\mathrm{egf}})^{(k-1)}(t_i)(\alpha\cdot\operatorname{sign}(\boldsymbol{g}(t_i)) + (1-\alpha)\cdot\boldsymbol{g}(t_i)) + \mathcal{O}((\boldsymbol{I}^i_{\mathrm{egf}})^{(k-2)}(t_i))\right),$$

where $\mathcal{O}((\boldsymbol{I}^i_{\mathrm{egf}})^{(k-2)}(t_i))$ *depends only on derivatives of order* $k-2$ *and lower, and* $\mathcal{O}((\boldsymbol{I}^i_{\mathrm{egf}})^{(k)}(t_i)) = \boldsymbol{0}$ *for* $k < 0$.

Proof
We begin by showing that

$$\boldsymbol{\Omega}^{(k)}(t_i,t_i) = -\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\cdot(\boldsymbol{I}^i_{\mathrm{egf}})^{(k-1)}(t_i) + \frac{1}{1-\gamma}\mathcal{O}((\boldsymbol{I}^i_{\mathrm{egf}})^{(k-2)}(t_i)), \tag{38}$$

where $\mathcal{O}((\boldsymbol{I}^i_{\mathrm{egf}})^{(k-2)}(t_i))$ depends only on derivatives of order $k-2$ and lower, and $\mathcal{O}((\boldsymbol{I}^i_{\mathrm{egf}})^{(k)}(t_i)) = \boldsymbol{0}$ for $k < 0$.

According to Magnus (1954)

$$\begin{aligned}\boldsymbol{\Omega}^i(t_i,t) = &\int_{t_i}^t \boldsymbol{A}(\tau_1)d\tau_1 + \frac{1}{2}\int_{t_i}^t\int_{t_i}^{\tau_1}[\boldsymbol{A}(\tau_1),\boldsymbol{A}(\tau_2)]\,d\tau_2 d\tau_1\\ &+\frac{1}{4}\int_{t_i}^t\int_{t_i}^{\tau_1}\int_{t_i}^{\tau_2}[\boldsymbol{A}(\tau_1),[\boldsymbol{A}(\tau_2),\boldsymbol{A}(\tau_3)]]\,d\tau_3 d\tau_2 d\tau_1 + \ldots,\end{aligned}$$

where the commutator is defined according to $[\boldsymbol{A},\boldsymbol{B}] := \boldsymbol{A}\boldsymbol{B} - \boldsymbol{B}\boldsymbol{A}$.

Multiple applications of the fundamental theorem of calculus result in

$$(\boldsymbol{\Omega}^i)^{(k)}(t_i,t_i) = \boldsymbol{A}^{(k-1)}(t_i) + \frac{1}{2}\mathcal{O}(\boldsymbol{A}^{(k-2)}(t_i)) + \frac{1}{4}\mathcal{O}(\boldsymbol{A}^{(k-3)}(t_i)) + \ldots$$

where $\mathcal{O}(\boldsymbol{A}^{(k)}(t_i))$ depends only on derivatives of order k and lower, and $\mathcal{O}(\boldsymbol{A}^{(k)}(t_i)) = \boldsymbol{0}$ for $k < 0$. Setting $\boldsymbol{A}(t) = -\frac{1-\alpha}{1-\gamma}\hat{\boldsymbol{\Sigma}}\boldsymbol{I}^i_{\mathrm{egf}}(t)$ results in Equation 38.

Next, we note that derivatives of order k only appear in terms where $i = 0$ or $i = k$ in Equation 41 and obtain

$$\frac{d^k\exp(\boldsymbol{X}(t))}{dt^k} = \sum_{n=1}^{\infty}\frac{1}{n!}\left(\boldsymbol{X}(t)\frac{d^k\boldsymbol{X}(t)^{n-1}}{dt^k} + \frac{d^k\boldsymbol{X}(t)}{dt^k}\boldsymbol{X}(t)^{n-1} + \mathcal{O}(\boldsymbol{X}^{(k-1)}(t))\right), \tag{39}$$

where $\mathcal{O}(\boldsymbol{X}^{(k-1)}(t))$ depends only on derivatives of order $k-1$ and lower, and $\mathcal{O}(\boldsymbol{X}^{(0)}(t)) = \boldsymbol{0}$. Since $\boldsymbol{\Omega}(t_i,t_i) = \boldsymbol{0}$, for $n \in \mathbb{N}_0$,

$$\boldsymbol{\Omega}^n(t_i,t_i) = \begin{cases}\boldsymbol{I}, & n = 0\\ \boldsymbol{0}, & n > 0,\end{cases}$$

and inserting $\boldsymbol{\Omega}(t_i, t_i)$ into Equation 39 we obtain

$$\begin{aligned}
&\frac{d^k \exp(\boldsymbol{\Omega}(t_i,t_i))}{dt^k} \\
&= \sum_{n=1}^{\infty} \frac{1}{n!} \left(\underbrace{\boldsymbol{\Omega}(t_i,t_i)}_{=\mathbf{0}} \cdot (\boldsymbol{\Omega}^{n-1})^{(k)}(t_i,t_i) + \boldsymbol{\Omega}^{(k)}(t_i,t_i) \cdot \underbrace{\boldsymbol{\Omega}(t_i,t_i)^{n-1}}_{\boldsymbol{I},\ n=1;\ \mathbf{0},\ n>1} + \mathcal{O}(\boldsymbol{\Omega}^{(k-1)}(t_i,t_i)) \right) \\
&= \boldsymbol{\Omega}^{(k)}(t_i,t_i) + \mathcal{O}(\boldsymbol{\Omega}^{(k-1)}(t_i,t_i)) = -\frac{1-\alpha}{1-\gamma} \hat{\boldsymbol{\Sigma}} \cdot (\boldsymbol{I}^i_{\text{egf}})^{(k-1)}(t_i) + \frac{1}{1-\gamma} \mathcal{O}((\boldsymbol{I}^i_{\text{egf}})^{(k-2)}(t_i)).
\end{aligned} \tag{40}$$

Now

$$\begin{aligned}
\boldsymbol{g}^{(k)}(t_i) =& (-\hat{\boldsymbol{\Sigma}}(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{egf}}(t)))^{(k)} = \hat{\boldsymbol{\Sigma}} \hat{\boldsymbol{\beta}}^{(k)}_{\text{egf}}(t_i) \\
&\overset{(a)}{=} -\frac{1}{1-\alpha} \exp(\boldsymbol{\Omega}^i(t_i,t_i))^{(k)} \left(\alpha \cdot \boldsymbol{s}^i + (1-\alpha) \cdot \hat{\boldsymbol{\Sigma}} \left(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{egf}}(t_i) \right) \right) \\
&\overset{(b)}{=} \frac{1}{1-\gamma} \left(\hat{\boldsymbol{\Sigma}} \cdot (\boldsymbol{I}^i_{\text{egf}})^{(k-1)}(t_i) \left(\alpha \cdot \underbrace{\boldsymbol{s}^i}_{=-\text{sign}(\boldsymbol{g}(t_i))} + (1-\alpha) \cdot \underbrace{\hat{\boldsymbol{\Sigma}} \left(\hat{\boldsymbol{\beta}}^{\text{OLS}} - \hat{\boldsymbol{\beta}}_{\text{egf}}(t_i) \right)}_{=-\boldsymbol{g}(t_i)} \right) \right. \\
&\quad \left. + \mathcal{O}((\boldsymbol{I}^i_{\text{egf}})^{(k-2)}(t_i)) \right) \\
&= \frac{1}{1-\gamma} \left(-\hat{\boldsymbol{\Sigma}} (\boldsymbol{I}^i_{\text{egf}})^{(k-1)}(t_i) \left(\alpha \cdot \text{sign}(\boldsymbol{g}(t_i)) + (1-\alpha) \cdot \boldsymbol{g}(t_i) \right) + \mathcal{O}((\boldsymbol{I}^i_{\text{egf}})^{(k-2)}(t_i)) \right),
\end{aligned}$$

where (a) follows from Equation 22 and (b) follows from Equation 40. ■

Lemma 15.

$$\frac{d^k \exp(\boldsymbol{X}(t))}{dt^k} = \sum_{n=1}^{\infty} \frac{1}{n!} \sum_{i=0}^{k} \binom{k}{i} \cdot \frac{d^i \boldsymbol{X}(t)}{dt^i} \cdot \frac{d^{k-i} \boldsymbol{X}(t)^{n-1}}{dt^{k-i}}. \tag{41}$$

Proof
For $n \geq 1$, according to the general Leibniz rule, $(fg)^{(k)} = \sum_{i=0}^{k} \binom{k}{i} f^{(i)} g^{(k-i)}$,

$$\frac{d^k \boldsymbol{X}(t)^n}{dt^k} = \frac{d^k \left(\boldsymbol{X}(t) \boldsymbol{X}(t)^{n-1} \right)}{dt^k} = \sum_{i=0}^{k} \binom{k}{i} \cdot \frac{d^i \boldsymbol{X}(t)}{dt^i} \cdot \frac{d^{k-i} \boldsymbol{X}(t)^{n-1}}{dt^{k-i}}.$$

Inserting this into the Taylor expansion of the matrix exponential, we obtain

$$\frac{d^k \exp(\boldsymbol{X}(t))}{dt^k} = \sum_{n=1}^{\infty} \frac{1}{n!} \frac{d^k \boldsymbol{X}(t)^n}{dt^k} = \sum_{n=1}^{\infty} \frac{1}{n!} \sum_{i=0}^{k} \binom{k}{i} \cdot \frac{d^i \boldsymbol{X}(t)}{dt^i} \cdot \frac{d^{k-i} \boldsymbol{X}(t)^{n-1}}{dt^{k-i}}.$$

■

References

Alnur Ali, J Zico Kolter, and Ryan J Tibshirani. A continuous-time view of early stopping for least squares regression. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1370–1378. PMLR, 2019.

Stephen P Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge university press, 2004.

Luis M Candanedo, Véronique Feldheim, and Dominique Deramaix. Data driven prediction models of energy use of appliances in a low-energy house. *Energy and Buildings*, 140:81–97, 2017.

Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.

Bradley Efron, Trevor Hastie, Iain Johnstone, Robert Tibshirani, et al. Least angle regression. *Annals of Statistics*, 32(2):407–499, 2004.

Jerome Friedman and Bogdan E Popescu. Gradient directed regularization. *Unpublished manuscript, http://www-stat. stanford. edu/˜ jhf/ftp/pathlite. pdf*, 2004.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1, 2010.

Trevor Hastie, Jonathan Taylor, Robert Tibshirani, Guenther Walther, et al. Forward stagewise regression and the monotone lasso. *Electronic Journal of Statistics*, 1:1–29, 2007.

François Kawala, Éric Gaussier, Ahlame Douzal-Chouakria, and Eustache Diemert. A study of different keyword activity prediction problems in social media. *International Journal of Social Network Mining*, 2(3):224–255, 2016.

Wilhelm Magnus. On the exponential solution of differential equations for a linear operator. *Communications on Pure and Applied Mathematics*, 7(4):649–673, 1954.

R Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3):291–297, 1997.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

Saharon Rosset, Ji Zhu, and Trevor Hastie. Boosting as a regularized path to a maximum margin classifier. *Journal of Machine Learning Research*, 5:941–973, 2004.

Noah Simon, Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A sparse-group lasso. *Journal of Computational and Graphical Statistics*, 22(2):231–245, 2013.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 58(1):267–288, 1996.

Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

Ryan J Tibshirani. A general framework for fast stagewise algorithms. *Journal of Machine Learning Research*, 16(1):2543–2588, 2015.

Gregory Vaughan, Robert Aseltine, Kun Chen, and Jun Yan. Stagewise generalized estimating equations with grouped variables. *Biometrics*, 73(4):1332–1342, 2017.

Simon N Wood, Zheyuan Li, Gavin Shaddick, and Nicole H Augustin. Generalized additive models for gigadata: Modeling the uk black smoke network daily data. *Journal of the American Statistical Association*, 112(519):1199–1210, 2017.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

Mimi Zhang. Forward-stagewise clustering: An algorithm for convex clustering. *Pattern Recognition Letters*, 128:283–289, 2019.

Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.