Forside

Eksamensinformation
NFYK10020E - Physics Thesis 60 ECTS, Niels Bohr Institute - Kontrakt:131696 (Øyvind Andreas Winton)

Besvarelsen afleveres af
Øyvind Andreas Winton
dqc205@alumni.ku.dk

Eksamensadministratorer
Eksamensteam, tel 35 33 64 57
eksamen@science.ku.dk

Bedømmere
Aslak Grinsted
Eksaminator
aslak@nbi.ku.dk
☎ +4535320510

Besvarelsesinformationer

Titel: Scientific machine learning for discovering basal dynamics of Greenland outlet glaciers
Titel, engelsk: Scientific machine learning for discovering basal dynamics of Greenland outlet glaciers
Tro og love-erklæring: Ja
Indeholder besvarelsen fortroligt materiale: Nej

MSc thesis in Computational Physics

Scientific machine learning for discovering basal dynamics of Greenland outlet glaciers

Author: Øyvind Andreas Winton (student ID: dcq205)
Supervisors: Aslak Grindsted (NBI, University of Copenhagen), Allan P. Engsig-Karup (DTU Compute) and Sebastian B. Simonsen (DTU Space)

Submitted on 1 August 2022

Abstract

Basal dynamics of outlet glaciers are essential in predicting the Greenland ice sheets' contribution to sea-level rise. Despite this, they are poorly constrained by observations and generalizable models. This study combines physical models and machine learning to discover relations for basal stress from data through new system identification and parameter estimation techniques under the shallow shelf approximation. The most generalizable model discovered was $\tau \propto u^{-1/2} + 1.5us$, which explains 30% of the variance of a spatio-temporal extrapolation test data set. The most generalizable extended power-law formulation was $\tau \propto u^{0.31}s^{0.45}$ explaining 24% variance. Fitting parametric models to each glacier individually yielded models that explain 77% of the variance in temporal extrapolation. The models for individual glaciers generally have negative exponents for velocity, contrary to the commonly assumed positive exponent. No relation between basal stress and meltwater was identified. This study lays the foundation for further attempts to discover ice-flow dynamics from data.

Contents

1 Introduction

Outlet glaciers play an essential role in the mass loss of the Greenland Ice Sheet, directly impacting global sea levels (Church and White, 2011). Basal processes modulate solid ice discharge into the ocean, constituting a large part of ice sheet mass loss (Mankoff et al., 2021). However, retrieving direct observations of basal regions is challenging, so our understanding of the processes that modulate glacier sliding is limited to modelling and surface observations (Jay-Allemand et al., 2011; Stearns and van der Veen, 2018). Whether or not a predictive model of basal processes exists is not known (Cuffey and Paterson, 2010). This study aims to discover such a model through the combination of a physical model and large data sets of observations.

Ice-flow modelling relates various properties and observables of glaciers in time and space. Various models exist, derived from the Navier-Stokes equations governing fluid dynamics (Cuffey and Paterson, 2010). Depending on the question, they differ in their simplifications and assumptions, making them useful in different domains (Bueler and Brown, 2009). The most complex models are used for contemporary time scales and individual glaciers or drainage basins. More simplified models are well-suited for describing the interior of ice sheets or for century-scale evolution modelling. Particularly simplified are the shallow models, hereunder the Shallow Shelf Approximation (SSA).

The approach of inferring basal properties through inversion of ice-flow models have been widely used since introduced by MacAyeal (1989). The idea is to tune parameters related to basal resistance, sliding and deformation in ice-flow models of varying complexity to match the output of the ice-flow model velocity field to observed velocities. Joughin and Alley (2011) apply this to the Ross Ice Shelf in Antarctica to learn the spatial distribution of soft- and hard-bedded flow; Jay-Allemand et al. (2011) apply this to Variegated Glacier in Alaska to understand how the subglacial drainage systems change during a surge; Winton et al. (2022) apply this to Hagen Bræ in Greenland, to investigate changes in flow resistance at the base during a surge period. Common to these methods is the spatio-temporal tuning of a parameter in the deterministic models to fit observed velocities.

Machine Learning (ML) is a term that covers a large variety of methods, with origins in applied statistics (Bishop, 2006). The general idea of supervised ML is to learn patterns from labelled training data and then make predictions for input data not seen by the model. Typically, the functional form of the models is less restricted than in deterministic modelling, which makes it a very flexible framework for many problems where the underlying dynamics are poorly constrained. Neural Networks (NNs) are a group of highly non-linear and flexible class models that have been proven to approximate any function arbitrarily good (Cybenkot, 1989). NNs can be applied to many problems where the underlying dynamics are not sufficiently constrained to represent explicitly in equations or algorithms.

Deterministic modelling and machine learning meet in an emerging field coined Scientific Machine Learning (SciML) (Rackauckas et al., 2020). It is a term that covers methods that utilize the developments in scientific computing, in particular optimization and differential equations, along with the more recent developments in machine learning and automatic differentiation. It is said to illuminate the black-box of machine learning by combining it with the white-box (deterministic) modelling to enter the spectrum of grey-box modelling. Rather than a purely deterministic approach or a purely data-driven approach, these new methodologies allow for solving problems where, e.g. only parts of the system's dynamics are known or where data are scarce.

The application of Machine Learning within glaciology is limited, with most papers published in the last few years. Bolibar et al. (2020) use a Neural Network and a simple glacier evolution scheme to reconstruct mountain glaciers' surface mass balance (SMB) in the French Alps, filling out observational gaps in time and space. Brinkerhoff et al. (2020); Jouvet et al. (2021) use a Neural Network to learn ice-flow dynamics from deterministic model outputs for use as a surrogate model to speed up evaluation. Zhang et al. (2021) use NNs to automatically delineate satellite imagery to delineate calving-fronts in Greenland. Jenkins et al. (2021) use classification methods to analyze seismic data from Ross Ice Shelf.

This study aims to combine scientific computing and machine learning methods to learn about Greenland outlet glaciers' basal processes. The basal stress is calculated by applying the SSA to the flowline of 34 outlet glaciers. The goal is to gain insight into what variables control the basal stress in time and space, using various methods that combine deterministic modelling and data-driven modelling from least squares through neural networks.

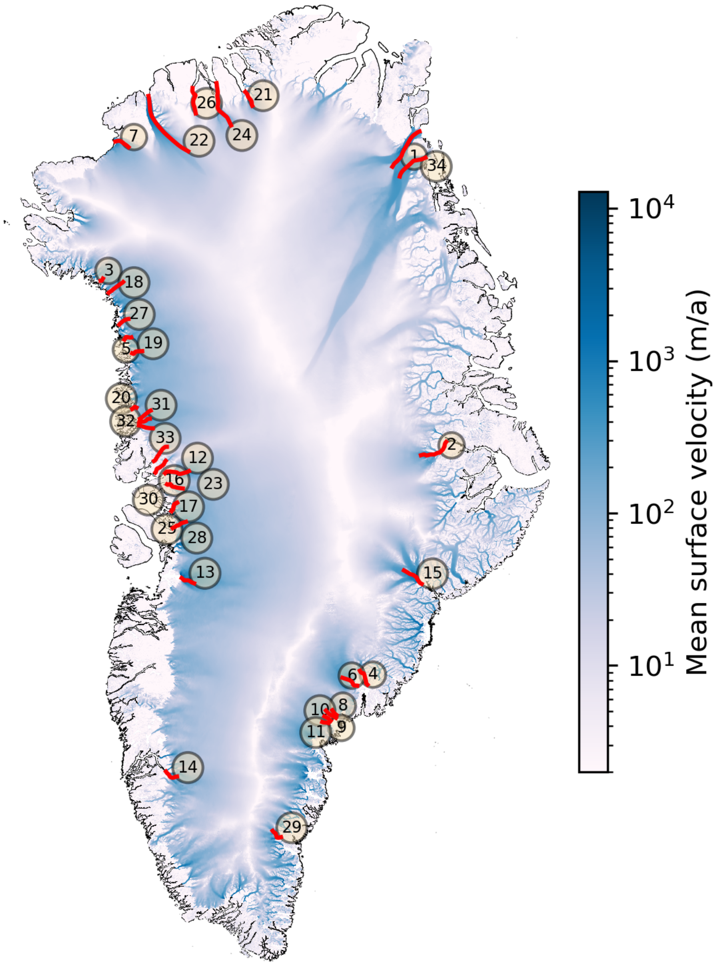

Figure 1: Illustration of flowlines on an outline of Greenland, with mean surface velocity from Joughin et al. (2018). The flowlines used in processing are cut off up and downstream compared to the ones shown in the figure. The numbering of glaciers is described in Table A.6. Some glaciers are located too near for the labels to be easily identifiable. However, the analysis in this study does not relate to individual glaciers, and the numbering of glaciers is done to give an overview.

2 Data and study area

The focus is on Greenland marine-terminating outlet glaciers. The study area comprises 34 marine-terminating glaciers in Andersen et al. (2019). The location of the glaciers can be seen in Figure 1 with the names listed in Table A.6. An overview of the scaled data is shown in Figure 3.

2.1 Data sources

Data	Variable	Exponent in Eq. (4)	Reference
Velocity	u	m_1	Solgaard and Kusk (2022)
Ice thickness	t	m_2	Morlighem et al. (2017)
Surface topography	s	m_3	Howat et al. (2014)
RACMO runoff	ro	m_4	Noël et al. (2019)
Basal melt from friction	fr	m_5	Karlsson et al. (2021)
Basal melt from ground flux	gf	m_6	Karlsson et al. (2021)
Strain rate	u_x	m_7	Derived from velocities

Table 1: Overview of used data, variable name and associated exponent in Eqs. (4).

Surface topography is obtained from the Greenland Ice Mapping Project (GIMP) (Howat et al., 2014). The digital elevation model (DEM) approximates the mean elevation over the years 2003-2009, ignoring temporal changes in elevation, and is a combination of three different data sets; laser altimetry from ICESat, stereo-photogrammetry from ASTER and photogrammetry from SPOT-5. The DEM is provided on a 30 m grid, and the reported RMS error is 8.5 m for the ice-covered regions.

Bedrock topography is obtained from BedMachine v3 (Morlighem et al., 2017). The product is a compilation of various radar-derived ice thickness measurements from 1993 to 2016. Along the coastal margin where radar-derived ice thickness is either lacking or of low quality, the data set has been extended using a mass conservation approach (Morlighem et al., 2011) that uses high-resolution surface velocity data to infer ice thicknesses that are physically consistent with ice flow dynamics. The data are provided on a 150 m grid, subsampled from resolutions of 400 m in the coastal regions, with an average reported error of 57 m for used data points. A histogram of errors for the used data points is shown in Figure 2.

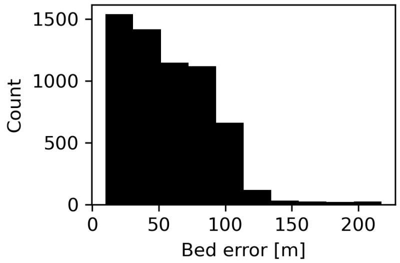

Figure 2: Histogram of reported errors interpolated to grid points used in this study. Mean reported error is 57 m.

Surface velocity is obtained from the PROMICE velocity products (Solgaard et al., 2021; Solgaard and Kusk, 2022). The products are processed from single-look complex images (SLC) acquired with synthetic aperture radar (SAR) from Sentinel-1A/B. The twin satellites are in an identical near-polar orbit with a repeat cycle of 12 days, phased 6 days apart. Velocity calculations are based on offset tracking, which requires an image pair, where the displacement of a feature is tracked, yielding the surface-parallel velocity. Data sets are provided for every 12 days, with data sets spanning 24 days. The temporal baseline of the data is from September 2016 to present, with the most recent data set used in this study being from December 2021. The data are provided on a 500 m grid, with an effective resolution of 800-900 m caused by the window size in the offset tracking. Comparison to GPS and ground control points gives a standard deviation of 10 m/y for the velocity magnitude.

Melt and runoff is obtained from regional climate models (RCMs). The used model is the Regional Atmospheric Climate Model (RACMO) and is an extension of the data set in Noël et al. (2019). The forcing is based on ERA5 (2015-2021) reanalyses, and the data is provided upscaled from a 5.5 km resolution to a 1 km grid on a daily resolution from 2015-2021. A conservative estimate of runoff uncertainty is provided at 20%, while average runoff biases reach 5% for a validation period spanning 1976-2016. Runoff represents the combination of surface melt and rain, including losses from refreezing in the firn.

Basal melt is obtained from Karlsson et al. (2021). Three energy sources for basal melt are used; geothermal heat flux (assumed constant in time), frictional heat from sliding at the base (mean of 1995-2015), and heat from surface meltwater (mean of 1995-2010). The resulting basal melt is provided for each source. The melt from surface meltwater has negative values scattered around the outlet glaciers and is thus left out as most methods require positive input data. The dataset is provided on a 1 km grid. Uncertainties in this data set are plentiful, as the processes and physics are poorly constrained. The reported total for geothermal flux is $5.3 + 2.8/-2.2$ Gt per year; the total for friction is 10.9 ± 3.0 Gt per year; the total for surface water is 5.2 ± 1.6 Gt per year.

Glaciers and flowlines. A flowline for each glacier was determined based on MEaSUREs Multi-year Greenland Ice Sheet Velocity Mosaic (Joughin et al., 2018), with a temporal span from 1995 to 2015. From a group of starting points upstream, a flowline was created by taking a small step in the direction of the velocity field, repeating this until the flowline reached the calving front. A group of flowlines for each glacier was determined, and for each glacier, a flowline was chosen based on visual inspection, with the heuristic goal of choosing the most central flowline. The flowlines were calculated with a step size of 2 m and downsampled to 250 m, yielding the spatial grid size for discretization $\Delta x = 250$ m.

2.2 Data processing

The data mentioned above are supplied on regular grids in a Polar Stereographic projection. Each data set is linearly interpolated to all the flowline coordinates (at each time step, where applicable). Topographic and velocity data are smoothed with a Gaussian filter with a kernel width of 2 times the average ice thickness of each glacier, stabilizing the calculation of driving and extensional stresses (McCormack et al., 2019). The RCM data are summed in time from the previous mean acquisition day for each mean acquisition day of the velocity data. All interpolation is done in Polar Stereographic projection. Calculations of distances along flowlines are calculated in corresponding UTM projections.

To avoid the grounding line, the data are cut off 5 km from the PROMICE calving front line (Andersen et al., 2019).[1] Further, the domain of each glacier was cut off upstream to focus on the downstream and more ice stream-like parts of the glaciers, based on heuristic visual inspection of where the strain rates start to increase, and the

[1] It is noted that the length of the ice shelf from the grounding line to the calving front might, in some cases, be longer than 5 km, which introduces possible grounding line dynamics not taken into account in the calculation of stresses.

extensional stresses start to play a role in the stress balance. Regions with negative driving or basal stresses have been removed from the data set. The extensional stress diverges for strain rates of 0 s^{-1}, and the time series where this divergence occurs has been removed from the data set. Velocities have been thresholded such that all velocities with an uncertainty of more than 25% of the magnitude of the velocity at that point have been removed.

The RCM data (melt and runoff) have been summed in time and space, assuming that upstream and previous melt/runoff affect the basal stress. Temporal summing is performed for each observation, and the sum of the 19 previous observations has been added to this grid point. Spatial summing is done in a cumulative sense; for each observation, all upstream observations have been summed and added to this grid point. This processing was done with a somewhat arbitrary temporal baseline of 20 observations (approx. 240 days). A more data-driven approach to selecting this hyperparameter (temporal baseline for summing RCM data) is presented in Section 3, with a method named prodCNN.

As seen in Figure 3, all the included data sets from RCMs (here features from both RACMO and HIRHAM have been displayed) are highly correlated, with correlations ranging from 0.73 to 0.97. This correlation proved difficult in optimization, as the corresponding columns are close to parallel, yielding poor fits to observation in synthetic testing. Following this, it was chosen to drop all RCM features, except RACMO runoff summed in time. RACMO was chosen due to the higher temporal and spatial resolution in the data set available during this study.

The data have been organized in a data matrix, with each row corresponding to a point in time and space and each column being a feature. Rows containing at least one non-positive value have been removed. While not necessary for all methods, to streamline analysis, it was chosen to work with the same data set for all methods, where applicable.

2.3 Test and training data

For cross-validation, the data have been split into four test data sets and one training data set, similar to the splitting in Bolibar et al. (2020). The split is illustrated in Figure 4. Leave-glaciers-out (LGO), leave-years-out (LYO) and leave-glaciers-and-years-out (LGYO) test the models' ability to extrapolate in spatial and temporal

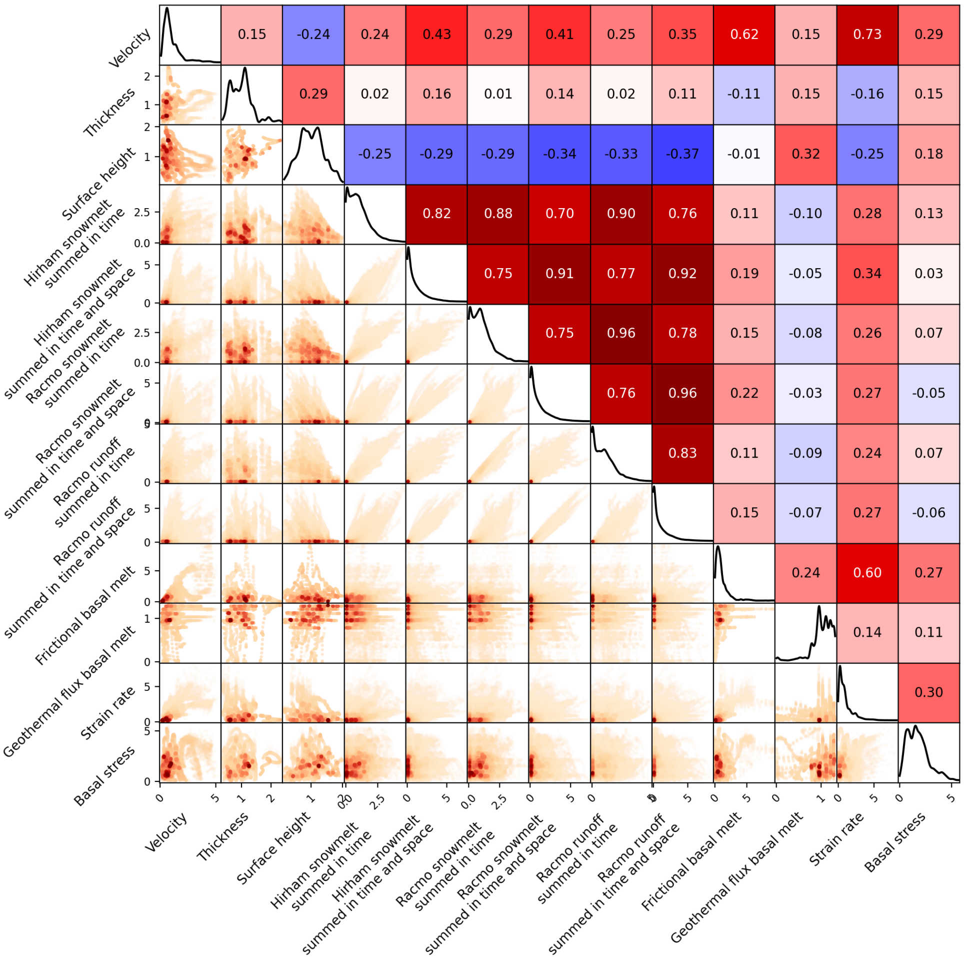

Figure 3: A visualization of the 13 features on the ≈ 260.000 data points in time and space. **Diagonal:** Kernel density estimation of the distribution of each variable. **Below diagonal:** variables plotted against each other, with higher density regions in darker colours. **Above diagonal:** linear correlation coefficient for each pair of variables, with positive correlations in reds and negative correlations in blue.

domains not seen by the model in training. LGO tests the models' performance on

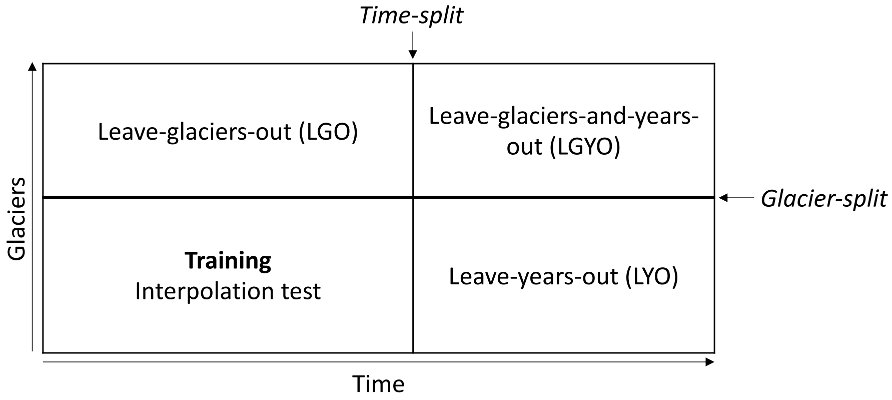

Figure 4: Schematic of how the data has been split into a training data set for training and four different test data sets for evaluation of the performance of estimators. The time-split, glacier-split and training/interpolation-split are defined to yield 5 data sets of approximately the same size. This splitting of data allows quantifying a model's ability to interpolate, as well as temporally and spatially extrapolate.

glaciers not seen by the models during training but during a period where the model is trained. LYO tests the models' performance on the glaciers it has been trained on, but for times after the training period. LGYO tests the models' performance on data from glaciers that the model has not been trained on and observations after the training data period. Finally, the models' are also evaluated on an interpolation test data set, chunks (in space) of data points from the spatio-temporal domain of the training data, which the model has not seen in training. This division allows quantifying how generalizable the predictions of the different models are in different degrees of extrapolation. The time- and glacier splits were defined to yield roughly similar data set sizes.

2.4 Data terminology and standardization

This study's overarching goal is to determine the basal stress τ, which will generally be denoted as the target variable; it is the variable that models are trained to yield a matching output. The data described earlier in this section will be denoted features. This is standard terminology used within machine learning, where typically, the objective is learning a model that maps from features to target variables.

All data have been re-scaled to dimensionless variables on similar scales[2]. The target variable was scaled to have unit variance, common in ML. The scales of the target and features are summarized in Table 2. Besides improved convergence in optimization, a benefit is that a baseline MSE is 1.0 if the model is simply by the mean of all target observations. All features were scaled with the mean of each feature. Scaling features with variance proved problematic for later applications, where some datasets had features close to constant over the domain, which yielded numeric instabilities.

For methods requiring positive features and targets, no centring has been performed. For methods allowing negative features and targets, data have been centred such that the training data are zero-mean. All standardization was done with means and standard deviations calculated from the training data set.

Variable	Scale	Units
τ	212	kPa
u	3.01	m d^{-1}
t	1063	m
s	839	m
ro	3.75	mm w.e. d^{-1}
fr	0.23	m y^{-1}
gf	0.0056	m y^{-1}
ux	0.00016	d^{-1}

Table 2: Overview of scaling of each data variable.

[2] This has the implication that all figures, except Figures 1 and 2, contain only dimensionless quantities, without any specification of units.

3 Methods

The theoretical foundation of the study is presented in this section. The ice-flow dynamics are introduced. Neural networks are introduced as a concept. Finally, the different numerical methods to learn from data are presented, along with key assumptions and important choices of parameters.

3.1 Ice-flow dynamics

Glaciers are assumed to behave like a slow non-Newtonian fluid, their flow properties governed by Glen's flow law (Glen, 1958). The Navier-Stokes equations fully describe the flow (Cuffey and Paterson, 2010), where momentum advection and inertia terms have been neglected to yield the Stokes equations. Solving the full system of equations requires many boundary conditions and is computationally expensive, which has led to various simplifications to model the evolution of glaciers and ice sheets (Bueler, 2021). The shallow shelf approximation (SSA) applied in this study was developed in Morland et al. (1987); MacAyeal (1989), and has been widely used in studies (e.g. Schoof, 2007; Sergienko et al., 2008; Habermann et al., 2012; Tsai et al., 2015; Habermann et al., 2017). The details of the derivation are left out, but the assumptions are provided here. The first assumption is neglecting vertical normal stresses under the hydrostatic assumption, which allows for eliminating pressure from the momentum balance. Further terms are eliminated by neglecting horizontal derivatives of vertical velocity. Assuming that vertical shearing is dominated by basal sliding for ice streams, vertical shearing is dropped. Vertically integrating the equations and adding a basal stress term yields the SSA. In one dimension, flowline stress balance is (Tsai et al., 2015)

$$\underbrace{2A^{-1/n}\left(t|u_x|^{1/n-1}u_x\right)_x}_{\text{extensional stress}} - \underbrace{\tau}_{\text{basal stress}} - \underbrace{\rho g t s_x}_{\text{driving stress}} = 0, \tag{1}$$

where derivates are notated $u_x = \frac{\partial u(x)}{\partial x}$, $A = 9.3 \cdot 10^{-25}$ s^{-1} Pa^{-3} is the depth-averaged temperature-dependent rheological coefficient in Glen's flow law, $n = 3$ is the corresponding exponent, t is ice thickness, u is surface velocity, $\rho = 900$ kg m^{-3} is the density of ice, $g = 9.8$ m s^{-2} is the gravitational acceleration and s is the surface elevation. The three terms in Eq. (1) are extensional stress held by viscous deformation, basal shear stress held at the base by till strength, and gravitational driving stress (Bueler and Brown, 2009). Under the assumption of negligible vertical shearing, u will refer to both the velocity in the ice column, and it will be assumed in the rest of the study that $u_{\text{surface}} = u_{\text{base}}$.

The basal stress term is poorly constrained, with a variety of different suggestions for its parametrization. Weertman (1957) proposed a power-law relationship between basal stress and velocity

$$\tau = cu^{1/m}, \tag{2}$$

where c is the basal stress parameter and m is a constant, commonly assumed $m = 3$ (e.g Weertman, 1974; Schoof, 2007; Tsai et al., 2015). The value of m can be interpreted in terms of the hardness of the bed. For $n = 3$, values of $m = 3$ are associated with hard-bed sliding (Cuffey and Paterson, 2010). Higher values of m are found for deforming beds, as in Gillet-Chaulet et al. (2016).

Coulomb friction relates basal stress to water pressure and is expressed (Tsai et al., 2015)

$$\tau_b = f(\sigma_0 - p), \tag{3}$$

where f is a friction coefficient, $\sigma_0 = \rho gh$ is the ice pressure, and $p = \rho_w gb$ is the water pressure. The difference is called the effective pressure, $N = \sigma_0 - p$. Tsai et al. (2015) proposes a sliding law that changes from power-law to Coulomb friction closer to the grounding line, as one problem with power-law sliding is the discontinuity at the grounding line, which is not the case for Coulomb friction. Other formulations relate the basal stress to both effective pressure and velocity, such as in Jay-Allemand et al. (2011).

In this study, a data-driven approach is taken. By taking in features of the flow and geometry (velocity, strain rate, thickness, surface height) along with quantities related to water (surface runoff and basal melt), the idea is to learn how these relate to basal stress under the assumption of the SSA. An expanded variant of the power-law (Eq. (2)) is proposed:

$$\tau = m_0 \cdot u^{m_1} \cdot t^{m_2} \cdot s^{m_3} \cdot ro^{m_4} \cdot fr^{m_5} \cdot gf^{m_6} \cdot u_x^{m_7}, \tag{4}$$

where m_i are coefficients to be learned from data, ro is surface runoff, fr is basal melt due to friction, and gf is basal melt due to geothermal heat flux. Commonly in inversion studies, the basal stress parameter C in Eq. (2) is fitted as spatio-temporally variable. This study assumes it as a function with fixed parameters but of spatio-temporally variable features. The basal stress parameter C is sometimes informally referred to as a garbage term, collecting everything that does not fit by tuning the parameter in time and space. Thus, this approach can be interpreted as an attempt to open the garbage term, with $C = m_0 \cdot t^{m_2} \cdot s^{m_3} \cdot ro^{m_4} \cdot fr^{m_5} \cdot gf^{m_6} \cdot u_x^{m_7}$.

This formulation allows for interpreting the model and learning qualitatively about how basal stress relates to the various parameters. For positive exponents, there is a positive correlation between τ and the feature in question, and vice-versa for negative exponents. For exponents near 1, this relationship is nearly linear. For exponents near 0, basal stress is nearly invariant to variations in this feature. Further, this formulation allows scaling the data without consequence for the parameters (except for m_0), allowing for interpreting the models between different scalings.

3.2 Neural networks

Neural networks (NNs) are a branch of machine learning that has been widely applied to solve various problems. Particular successful applications are within speech and image recognition, where a specific type of NN involving discrete convolutions has aided in reaching new levels of accuracy (Schmidhuber, 2014).

At the core, an NN is a function that makes a non-linear mapping from and to arbitrary (but finite) input and output dimensions. The simplest NN is the feed-forward neural network (FFNN). Input is transformed with an affine transformation, and a non-linear function (called activation function) is applied element-wise to the transformed input:

$$y = \sigma\left(Wx + b\right), \tag{5}$$

where the output y is a $N \times 1$ vector, the weight matrix W is $M \times N$, the bias b is a $M \times 1$ vector, the input x is a $N \times 1$ vector, and the non-linear function σ is performed element-wise, where M is the number of features and N is the dimension of the output. This represents an FFNN with one input layer, no hidden layers, and one output layer. The training process is to update the weights and bias to minimize some loss function, which becomes a high-dimensional non-linear optimization problem. Extending to multi-layer (deep) NNs is simple, shown here for NN with two hidden layers

$$NN(x) = W_3\sigma_2\left(W_2\sigma_1\left(W_1x + b_1\right) + b_2\right) + b_3. \tag{6}$$

Usually, no activation function is applied to the final layer for regression problems. In NN terminology, the layer is the non-linear mapping, with the number of nodes being the number of elements. The affine transformation is, in this interpretation, a transformation between layers. The input is referred to as the input layer, the output as the output layer and any layers between are called hidden layers. The number of nodes in the input and output layers determines the input and output dimensions of the network. By choice of affine transformation, the number of neurons in the hidden layers is merely restricted to natural numbers, which can vary between hidden layers.

Dropout is a common tactic to prevent overfitting of NNs (Bishop, 2006). It simulates training multiple NNs with varying architectures in parallel: for each hidden node and at each iteration, there is a probability p that the node is dropped from the network. This has the effect that the output does not become overly sensitive to individual nodes in the network, resembling training an ensemble of networks and taking the mean of the output. A common value is $p = 0.3$ (Pedregosa et al., 2012). A commonly used activation function is the rectified linear unit (ReLU), which passes positive numbers without modification, and maps negative numbers to zero.

NNs are commonly trained using variations of stochastic gradient descent, an algorithm approximating gradient descent by calculating the gradient for a randomly selected subset of data. Adam (Kingma and Ba, 2014) is a specific algorithm that many common machine learning libraries default to for optimization (Pedregosa et al., 2012), its wide adaptation rooted in it being computationally efficient, having low memory requirements and being well suited for problems with large amounts of data and/or parameters. It revolves around stochastic gradient descent but adds the notion of momentum in its minimization efforts. The idea can be expressed

$$d_t = \gamma d_{t-1} + \eta \hat{\nabla}_{\mathbf{m_t}} L(\mathbf{m_t}), \tag{7}$$

$$\mathbf{m}_{t+1} = \mathbf{m}_t - d_t, \tag{8}$$

where d_t is the current step, which encodes γ of the previous step d_{t-1}, as well as η of the stochastic gradient with respect to the current model parameters $\hat{\nabla}_{\mathbf{m_t}}$ of the loss function at the current parameters $L(\mathbf{m_t})$. The idea of momentum is retaining part of the previous update step direction in the current update step.

Two main characteristics are the foundation for the success of NNs in a wide range of applications. First, it is generally assumed that NNs are universal function approximators (Bishop, 2006), with this property only proved for specific cases (e.g. Cybenkot, 1989). This means that a NN can approximate any continuous function to arbitrary precision, given enough layers and neurons. Other universal function approximators exist, such as the Taylor and Fourier series, but are challenging to work with in higher dimensions, suffering from the curse of dimensionality. NNs have been shown to overcome this issue (Donoho, 2014).

A second key to the success of NNS, besides more technical developments in optimization techniques, wide implementations of automatic differentiation and parallelization of training processes, is the incorporation of inductive biases into NN architectures (Rackauckas et al., 2020). One example is image segmentation, where convolutional filters apply the prior domain knowledge that image pixels are closely spatially correlated with neighbour pixels. Another is using Green's embedded in the structure of NNs to learn differential operators (Li et al., 2020).

NNs can be used as function approximators where the functional form is not well-constrained and have been applied in different contexts in this study.

3.3 From data to model: basal stress estimation

The target variable basal stress τ has been generated from observations and finite differences

$$\tau = A^{-1/n}\left(t|u_x|^{1/n-1}u_x\right)_x - \rho g t s_x, \tag{9}$$

where a centred difference scheme has been applied to all derivatives, with discretization errors $\mathcal{O}(\Delta x^2)$ (Bingham et al., 2020). The grid size has been set to $\Delta x = 250$ m and will not be further discussed. Finite difference issues caused by noise are reduced by the Gaussian filtering introduced in Section 2.

An overview of methods presented in this section is in Table 3.

3.3.1 Linear least-squares (LSQ and NTLSQ)

Taking the logarithm of both sides linearises Eq. (4) in terms of the model parameters

$$\begin{aligned}\log\tau = \log m_0 + m_1\log u + m_2\log t + m_3\log s + m_4\log ro \\ + m_5\log fr + m_6\log gf + m_7\log u_x,\end{aligned} \tag{10}$$

which requires that all features are positive numbers. A major implication of taking the logarithm and optimizing in log-space is that the optimization is shifted to focus on the relative difference between prediction and target rather than the absolute difference. A benefit to this is that the model is trying to learn dynamics from all glaciers equally, rather than particularly those with high stresses. On its own, this might be a desirable feature of a model working with a large range of data.

Method	Abbreviation	Model	Interpr.	Comput.
Linear least squares	LSQ	Eq. (10)	High	Low
N-term least squares	NTLSQ	Eq. (10)	High	Low
Non-linear least squares	NLSQ	Eq. (4)	High	Low
N-term non-linear least squares	NTNLSQ	Eq. (4)	High	Low
Individual non-linear least squares	INLSQ	Eq. (4)	High	Low
Markov Chain Monte Carlo	MCMC	Eq. (4)	High	High
Sparse identification of non-linear dynamics	SINDy	Eq. (18)	High	Medium
Non-linear least squares with convolution of runoff	prodCNN	Eq. (20)	High	Medium
Non-linear least squares closed with neural network	NLSQ+NN	Eq. (21)	Medium	High
Feed Forward Neural Network	FFNN	Eq. (22)	Low	High

Table 3: High-level overview of different methods applied to the problem. Interpretability and computational cost are qualitative categories given to provide an overview. For interpretability, the ratings are loosely defined as follows. High: the model is fully symbolic. Medium: the model has symbolic and non-symbolic parts. Low: there are only non-symbolic parts. For the computational cost, the ratings are loosely defined based on the training time until convergence, run on a MacBook Pro from 2018. High: runtime on the order of hours. Medium: runtime on the order of minutes. Low: runtime on the order of seconds. (Runtime does not include processing and splitting of data.)

However, a downside to this effect is that absolute underestimates for a given datum are penalized more than absolute overestimates since for the same absolute value of over- and underestimate, the relative value is higher for underestimates than for overestimates. This yields an asymmetry in the misfit.

Eq. (10) can be solved by linear least squares (LSQ), where the model parameters are determined to yield the smallest squared residual between prediction and observation (Aster et al., 2013). Organizing the logarithm of features as columns in a matrix $\mathbf{G}$, adding to this a column of $\exp\{1\}$, Eq. (10) can be expressed

$$\mathbf{d} = \mathbf{G}\mathbf{m}, \tag{11}$$

where $\mathbf{d}$ is a vector of the target variable $\log \tau$, and m is a vector containing the model parameters from Eq. (10). with the least-squares estimator

$$\mathbf{m} = (\mathbf{G}^\top \mathbf{G})^{-1} \mathbf{G}^\top \mathbf{d}. \tag{12}$$

A variant of this is sequentially thresholded least squares (STLSQ) (Brunton et al., 2016). The idea is to retrieve a sparse solution vector **m** by removing elements of **m** with a magnitude below some given threshold, solving the problem iteratively until convergence. To illustrate the effect of iteratively removing features from the solution, a slightly different approach was devised, inspired by STLSQ, which will be referred to as N-term LSQ (NTLSQ). For a given N, a solution to Eq. (11) is sought, with only the N features with the highest magnitude exponent being active. This is done by removing terms one at a time, solving the problem at each iteration with one coefficient less until a solution with N exponents is reached.

3.3.2 Non-linear least squares (NLSQ, NTNLSQ and INLSQ)

NLSQ is an extension of linear least squares and is a linearization of the problem in an iterative scheme (Aster et al., 2013). In this study, it is applied to solve the problem in Eq. (4). The least-squares solution is obtained by iterations of

$$\mathbf{m}_{k+1} = \mathbf{m}_k + \left(\mathbf{G}(\mathbf{m}_k)^\top \mathbf{G}(\mathbf{m}_k)\right)^{-1} \mathbf{G}(\mathbf{m}_k)^\top (\mathbf{d} - \tau(\mathbf{m}_k)), \tag{13}$$

where k is the current iteration number, $\tau(\mathrm{m}_k)$ is the basal stress as a function of the model parameters at the current iteration $\mathbf{m}_k$, **d** is the target variable, and the Jacobian is defined as derivatives of the basal stress wrt model parameters, with elements defined

$$G_{ij}(\mathbf{m}_k) = \frac{\partial \tau_i(\mathbf{m}_k)}{\partial m_j}. \tag{14}$$

As this is an iterative process, it requires a starting guess, defined as the LSQ solution. The stopping criterium is defined by the relative change in loss function

$$|\Delta MSE| < 10^{-8} MSE. \tag{15}$$

NLSQ was applied with the features determined by NTLSQ, which will be referred to as NTNLSQ. NLSQ was also applied to each glacier individually, yielding scale and exponents varying for each glacier. This is referred to as INLSQ.

3.3.3 Markov chain Monte Carlo (MCMC)

MCMC is a probabilistic approach to the problem of parameter estimation (Bishop, 2006). The foundation is Bayes' Theorem, which gives the proportionality between the posterior distribution and the product of the prior distribution and the likelihood,

$$p(\mathbf{m}|\mathbf{X}) \propto p(\mathbf{m})p(\mathbf{X}|\mathbf{m}), \tag{16}$$

where p is the probability distribution, $\mathbf{m}$ are the estimated model parameters and $\mathbf{X}$ is the matrix of observations. The posterior distribution over the model parameters is estimated by sampling the model parameter space and evaluating the product of prior and likelihood at the sampled set of parameters. The sampling follows the same general strategy: for a given proposed set of parameters, the parameters are accepted if they increase the posterior compared to the current parameters but are only accepted with a certain probability of they decrease the posterior. Following a choice of forward model and collection of data, this outlines the need for defining three characteristics of the problem: sampling strategy, prior of parameters and choice of likelihood function.

The most straightforward sampling strategies resemble a random walk in the parameter space, and while simpler to implement, they suffer from slow convergence for high-dimensional problems. Other more sophisticated methods use multiple previous steps in determining the distribution of the following sample. Hamiltonian Monte Carlo (HMC) is such an algorithm, and it draws upon ideas from the physics of conservation of the Hamiltonian in its search around parameter space (Neal, 2012). While details of the algorithm will be left out, the method can be conceptualized: imagining the negative posterior distribution as a surface, the sampler can be imagined as a particle on this surface that will tend to gravitate towards the minima. Random perturbations to its path help to get out of local minima, and the goal of this sampling is to sample more often where the posterior has a higher density and vice versa. In this study, an algorithm called the No-U-Turn Sampler (NUTS) is applied, an extension of HMC that eliminates hand-tuning of the step size and the number of steps (Hoffman and Gelman, 2014). Common to all sampling strategies is a warm-up/burn-in, where the first n samples are discarded, after which sampling starts.

The prior distribution of parameters was defined as normal distributions with unit variance, with the scale centred on a mean of 1 and all exponents on a mean of 0. MSE was chosen as the likelihood function. A common strategy in MCMC is running multiple independent realizations from different initial conditions. This leads to a more thorough exploration of parameter space and increases the certainty of the results. For all chains, a burn-in period of 2000 samples was used, and 10000 samples were subsequently sampled.

A significant advantage of probabilistic inference is that rather than obtaining a point estimate of the model parameters, a distribution is obtained, allowing putting confidence intervals on the obtained parameters. Further, it allows calculating the covariance between different parameters, giving further insight into how the different model parameters could be related. Finally, a value σ is also learned from data, representing the noise added to match the forward model to observations. However, it is not possible to split this noise into model and data uncertainty unless one of the terms is well-constrained, and the other could be assumed to be the remainder of the total noise.

MCMC is applied to estimate the parameters in Eq. (4).

3.3.4 Sparse identification of non-linear dynamics (SINDy)

SINDy was introduced by Brunton et al. (2016). It is designed for dynamical systems with both temporal and spatial dimensions and is motivated by problems in fluid dynamics. The assumption is that the system's dynamics can be represented by a sparse linear combination of library functions of the state space. A sparsity-promoting optimization algorithm determines the choice of terms on a linear inverse problem

$$\dot{\mathbf{X}} = \mathbf{\Theta}(\mathbf{X})\Xi, \tag{17}$$

where $\dot{\mathbf{X}}$ is a matrix whose columns are time derivatives of the states, $\mathbf{\Theta}$ is the library matrix whose columns are each candidate functions of the state space variables $\mathbf{X}$, and Ξ is a sparse matrix that optimizes some cost function, activating only a few functions from the library matrix. An important part of the work lies in choosing an appropriate set of candidate functions, which can be any function of one or more state-space variables.

The standard SINDy formulation is suitable for problems where the governing equations are assumed to be expressed in the sparse form shown above, posed as a set of ODEs. For the problem in this study, the approach was modified to reflect the assumed dynamics

$$\tau = \mathbf{\Theta}(\mathbf{X})\xi, \tag{18}$$

where τ is a vector containing inferred basal stress from observations, $\mathbf{X}$ is a matrix whose columns are observed features, and ξ is a sparse vector. Due to the number of features available, even when selecting a small subset (5) and just a few functions of one and two variables as candidate functions, the number of columns in $\mathbf{\Theta}$ increases quickly and finding a sparse solution ξ becomes subject to model and data uncertainty (Mangan et al., 2017).

Powers and a few cross-terms were chosen as library functions, inspired by the standard libraries suggested by Brunton et al. (2016). The used library functions are $x, x^{1/2}, x^{1/3}, x^{1/4}, x^{-1/2}, x^{-1/3}, x^{-1/4}, xy, \frac{x}{y}, xyz$, where any feature takes the place of x, y, z. Due to convergence issues and the results from NLSQ and similar methods, the set of features used for SINDy was reduced to u, s, t, u_x. With the library described above and the four features, this yields 44 columns in $\mathbf{\Theta}$.

An optimization algorithm with regularization and constraints is chosen to regularise the problem. The problem is constrained by requiring all non-zero elements of ξ to be positive. This reduces problems with near-parallel columns of $\mathbf{\Theta}$ appearing with opposite signs and limits all features to contribute positively to basal stress. The optimization is carried out with an algorithm called constrained Sparse Relaxed Regularized Regression (SR3), introduced in Zheng et al. (2019); Champion et al. (2019), where the objective function for optimization is

$$\frac{1}{2}\|\tau - \mathbf{\Theta}\xi\|_2^2 + \lambda\|\mathbf{w}\|_1 + \frac{1}{2}\|\xi - \mathbf{w}\|_2^2, \quad s.t.\mathbf{w} \geq \mathbf{0}, \tag{19}$$

where $\lambda \approx 4 \cdot 10^4$ is the regularization parameter, and $\mathbf{w}$ is an auxiliary variable, a relaxation of ξ. The auxiliary variable is regularized with the sparsity-promoting l_1 penalization, which is the convex relaxation of the true sparsity norm l_0. The relaxation of ξ improves convergence under optimization. The optimization algorithm's details and implementation will not be discussed further.

An important parameter to determine is the strength of regularization, λ. This was determined by calculating test losses for different values of regularization, choosing the value that yields the lowest test loss (Brunton et al., 2016), with the results shown in Figure A.27. While the training loss will increase monotonously as regularization increases, the test loss will typically have a defined minimum, which is not at the bounds of the domain.

3.3.5 Non-linear regression with convolution of runoff (prodCNN)

The idea of prodCNN is similar to regular non-linear regression, with the modification that runoff is convoluted with a learned spatio-temporal filter. For each datum $\tau_{k,x}$ the model is given each feature (except runoff) at the same point in time and space. The model is given a matrix for runoff that includes earlier time steps and upstream grid points. Before calculating the product in Eq. (4), the runoff matrix is convoluted with a filter of the same size, yielding a weighted sum of the matrix. The filter weights, the scale and the exponents are learned through optimization.

$$\tau = m_0 \cdot u^{m_1} \cdot t^{m_2} \cdot s^{m_3} \cdot \mathrm{conv}(ro)^{m_4} \cdot fr^{m_5} \cdot gf^{m_6} \cdot u_x^{m_7}, \tag{20}$$

where conv is the convolution operation. This method aims to learn how basal stress is affected by upstream and earlier runoff. It assumes that earlier and upstream runoff affects the basal stress at a given point and that the spatio-temporal weighting of this effect is similar across all glaciers at all times. Further, it is assumed that earlier and upstream runoff can only contribute positively; thus, the weights of the learned filter are passed through a sigmoid function before convolution. To facilitate comparison of the runoff exponent with other models, the filter is normalized to sum to 1.

3.3.6 Non-linear regression closed with neural network (NLSQ+NN)

This method is inspired by methods presented in Rackauckas et al. (2020), where differential operators are learned as parameter estimation closed with a neural network. The idea is to fit an NLSQ model with a subset of features plus a neural network of all features, where the runoff is convoluted similar to in prodCNN

$$\tau = m_0 \cdot u^{m_1} \cdot s^{m_3} + NN\left(u, t, s, \mathrm{conv}(ro), fr, gf, u_x\right). \tag{21}$$

The goal is that part of the relation can be learned by the NLSQ method, with the NN learning the residual. This has been shown in Rackauckas (2019) to generalize models by learning non-linear parts of the dynamics not expressed by the functional form.

3.3.7 Feed forward neural network (FFNN)

The idea is to learn the parameters of an FFNN such that

$$\tau = NN\left(u, t, s, ro, fr, gf, u_x\right), \tag{22}$$

where the neural network inputs are the input features as described in Eq. (4). This method has limited interpretability and acts as a test of whether or not a function can be discovered that maps from the input features to the target.

The objective function was set to the MSE of target and prediction. A relatively simple network is used, with seven input nodes, six hidden layers with 48 nodes, one output node, and the ReLU activation function. Dropout was set to $p = 0.3$. The optimization was carried out using Adam with a learning rate of 0.005. The number of input nodes is determined by the number of feature and target variables. The number of layers and nodes was determined by synthetic testing (see Section 4). The activation function, dropout rate and learning rate are common choices and will not be further discussed.

3.4 Coefficient of determination and mean squared loss

The coefficient of determination is defined (Glantz and Slinker, 2000)

$$R^2 = 1 - \frac{\sum_i (\tau_i - \hat{\tau}_i)^2}{\sum_i (\tau_i - \bar{\tau})}, \tag{23}$$

where τ_i are observations, $\hat{\tau}_i$ are predictions, and $\bar{\tau}$ denotes the mean. It is a measure of a model's ability to predict the target variable and can be interpreted as the explained variance of a given model. While there are limitations in its interpretation across different model types and degrees of freedom, it does yield an interpretable measure of fit, which will be used to compare different methods. The highest value is 1.0 is achieved for a perfect fit to data. A value of 0.0 indicates that the fit is as good as a baseline model, which is the average of the observations. Values lower than 0.0 indicate models that fit worse than a baseline model. The R^2 score of a model will be interpreted as the proportion of variance explained by the model.

Mean squared loss is defined

$$\text{MSE} = \frac{\sum_i^N (\tau_i - \hat{\tau}_i)}{N}, \tag{24}$$

where N is the number of data points. It is equal or proportional to the cost functions that most methods optimize for. A value of 0.0 indicates a perfect fit. A value of 1.0 is the baseline model, which is the average of observations, assuming that the training data have been scaled to unit variance.

All R^2 scores for training data should be above 0.0 and MSE below 1.0, with failures indicating that there is something in the method that is not working. Test data can fail to meet these requirements without there necessarily being an error in the method. Exceptions could occur for heavily regularized or constrained optimization problems, as seen in Figure A.27 for values of the regularization parameter.

3.5 Uncertainties

A distinction will be made between two fundamentally different types of uncertainty and noise, based on Hüllermeier and Waegeman (2021). Data uncertainty covers the systemic and random errors in variables. All observations in this study are, in principle, a combination of instruments doing measurements of some physical quantity (e.g. temperature or radio echo delay), with post-processing transforming it into a higher-level product (e.g. altitude, numerical weather model or surface velocity). Further processing is then done in filling gaps, interpolation etc.

Model uncertainty covers the systemic errors in the underlying models that generate data. Presumably, the underlying models governing basal dynamics are more variable than the parameterizations fitted to in this study, and the dynamics might vary across time and space. In synthetic testing, this is simulated by having the parameters of the generating model change for different data points, while the parameterizations fit to the observations do not allow for such variation.

3.6 Implementation details

The workflow from raw data to results is summarized here. Data are interpolated to flowlines of glaciers. Basal stress is calculated for each point in space and time. All features (data) and the target (basal stress) are organized as columns in a matrix, along with relevant metadata (name of glacier, date). Data are split into the test and training data sets. All data are scaled with scaled from the training data set.

A regression model is trained on the training data. Predictions are made on all five data sets, and the R^2 and MSE scores are calculated.

All the methods presented are implemented using Python 3.9 and Python 3.10. LSQ, NLSQ and related methods are implemented in SciPy (Virtanen et al., 2020) and NumPy (Harris et al., 2020). MCMC is implemented in NumPyro (Phan et al., 2019). SINDy is implemented in PySINDy (Kaptanoglu et al., 2021). Neural Networks are implemented using PyTorch (Paszke et al., 2019). The methods presented in the discussion are inspired by the DiffEqFlux.jl package for the Julia programming language (Rackauckas et al., 2020).

4 Synthetic testing of methods

In this section, the methods introduced in Section 3 are tested against synthetically generated data. The main objective of the synthetic test is to validate whether or not the methods can learn the model that generated the data in the presence of both model and data noise. Experiments 1-3 and 6-7 test the methods under model noise, with synthetic data generated by different models (varying the exponent for u or the scale), or by models with non-linear terms not in the assumed functional form. All experiments test the methods under data uncertainty, with 10% noise added to the target. Further, the NLSQ methods are tested with 0% and 100% noise added to the target variable. After adding noise to the target, the absolute value is taken to ensure that the target is positive.

Two measures of success are provided. The R^2 score is provided for test data set for each method, which measures the fit to data. Further, a score is introduced to measure the model error. It is defined as the l_1-norm of the exponent errors

$$\text{Model error} = \sum_{i=1}^{i=7} |m_i - \hat{m_i}|, \tag{25}$$

where m_i are the synthetic exponents and $\hat{m_i}$ are the learned exponents. The error is only calculated for the seven exponents in Eq. (4). For experiments 1, 2, 6 and 7, where the exponent for u is variable, the error has been calculated from the mean exponent of 0.36.

4.1 LSQ and NLSQ

Seven synthetic data experiments have been conducted for the synthetic test of least-squares methods. The experiments are mainly oriented around model uncertainty; most experiments are from a generating model that cannot be expressed by the assumed functional form in the least-squares approaches (Eqs. (4) and (10)). The details of the experiment are listed in Table 4. Common to all synthetic experiments is that zero-mean Gaussian noise with variance equal to 10% of each entry is added to the basal stress (target for inversion), except for the NLSQ experiments with 0% and 100% noise. Synthetic tests without data and model yield perfect fits, as seen in Table 4 for experiments 4 and 5 with no noise.

		Test R^2					Model error			
Exp. # and main challenge	True model	LSQ	NLSQ 0%	NLSQ 10%	NLSQ 100%	SINDy	LSQ	NLSQ 0%	NLSQ 10%	NLSQ 100%
1. Varying exponent	$3.52 \cdot u^{1/m}$	0.79	0.90	0.78	0.04	0.20	0.01	0.01	0.01	0.13
2. Varying exponent and scale	$c \cdot u^{1/m}$	0.53	0.58	0.53	0.05	0.47	0.01	0.01	0.01	0.13
3. Varying scale	$c \cdot u^{1/3}$	0.52	0.58	0.52	0.04	0.49	0.00	0.01	0.01	0.11
4. Many active terms	$1.26 \cdot u^{1/3} \cdot t^{1/7} \cdot ro^{1/2}$	0.89	1.00	0.98	0.25	0.59	0.22	0.00	0.01	0.30
5. Many active terms	$0.22 \cdot u \cdot t^{1/4} \cdot ro^{1/2} \cdot fr^{1/5} \cdot gf^{1/2}$	0.89	1.00	0.99	0.61	0.29	0.60	0.00	0.00	0.36
6. Includes non-linearity not in assumption	$c \cdot u^{1/m} \cdot \tanh t$	0.55	0.53	0.49	0.05	0.44	0.04	0.00	0.00	0.12
7. Includes non-linearity not in assumption	$c \cdot u^{1/m} \cdot \frac{1}{1+\exp(-ro)}$	0.48	0.69	0.65	0.07	0.50	0.08	0.21	0.22	0.07

Table 4: Overview of synthetic experiments, test R^2 and model errors for LSQ, NLSQ and SINDy. For relevant experiments, m is sampled uniformly from the integer set $\{2, 3, 4\}$ and c is sampled uniformly from an interval spanning approximately $[2; 4]$ (the exact interval is modified to keep the target variable on unity variance). 10% noise is added to the target variable unless else is noted. The scales of the true models differ for various amounts of noise added, with the values shown for a realization of 10% noise added.

The results of the synthetic experiments for NLSQ are presented graphically in Figure 5 with the R^2 scores and model errors summarized in Table 4. The results for NLSQ with 0% and 100% noise are in Figures A.20 and A.21. The results for LSQ are in Figure A.19. The resulting equations, thresholded for terms with a magnitude less than 0.01, are printed onto each frame. The x-axis in each frame represents the synthetic stress, with predictions on the y-axis. A black diagonal line illustrates a perfect fit of predictions onto the synthetic target.

LSQ and NLSQ yield similar results in terms of error, equations and visual appearance. The three first experiments represent the classic power-law sliding, with the model uncertainty that the underlying model is not the same - i.e. the parameters m and c are variable in the spatio-temporal domain. The case is, however, that both methods recover the mean exponent. For experiments 1 and 2, where the underlying model has exponents 1/2, 1/3 and 1/4, both methods find ≈ 0.36, which corresponds to the average of the true exponents.

Experiments 4 and 5 test the methods without model uncertainty but with a complex synthetic model comprising many active features. For experiments 4 and 5, both methods find exponents close to the underlying model; however, LSQ yields a poorer fit than NLSQ for higher stresses - this reflects that with LSQ, it is the relative residual for each point that is minimized, rather than the absolute as in classical

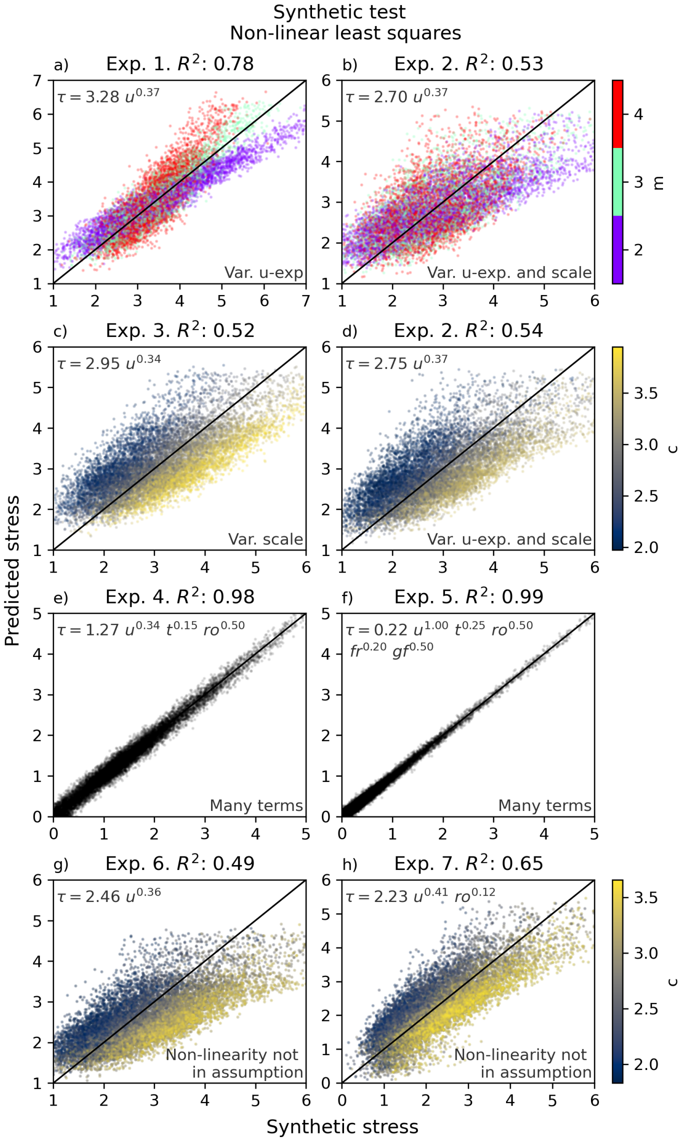

Figure 5: Results of synthetic test on seven experiments for NLSQ with 10% noise added to the target. The experiment numbers are detailed in Table 4, with a short explanation in the lower right of each panel of this figure. The resulting equations are in the upper left of each panel.

least squares, caused by the log-transformation to linearize the problem. From an absolute residual point-of-view, this leads to overfitting the lower stress values at the cost of the higher stress values, which can be seen in the skewing of the scatter points in Figure A.19.

Experiments 6 and 7 represent model uncertainty, displaying how the methods work in cases where a function of the wrong form is attempted to fit the data. The functions in use are hyperbolic tangent and the logistic function, both monotonic and non-linear. In experiment 6, NLSQ recovers the correct exponent for u but does not recover the dependence on t. In experiment 7, NLSQ finds a higher exponent for u, and here recovers that the basal stress is also a function of ro, even though the generating function is different from the function space that NLSQ is fitting in.

In Table 4, a summary of the model errors is shown in LSQ and NLSQ with different noise levels. LSQ successfully finds the correct models, with a model error below 0.1 for all except for experiments 4 and 5, where many features are activated. NLSQ perfectly recovers the models with up to 10% noise, with model errors below 0.01, except for experiment 7. With 100% added noise, NLSQ is still performing relatively well, with five of the experiments yielding model errors below 0.13, with the highest errors seen for experiments 4 and 5, where there are many active terms.

The NLSQ models are most successful in learning the correct underlying model, even with high measurement and model uncertainty. LSQ works well for simpler models but suffers when many terms are activated.

4.2 MCMC

The method is tested with Experiments 3 and 5. For Experiment 5, where the challenge is the number of active terms, the method works seamlessly, expressing with 90% confidence that the parameters are within intervals less than 0.01, for all but the scale, where the interval is slightly larger. Further, this method can estimate the amount of noise added to the target variable with equivalent precision. The results of this more trivial test have been omitted for brevity.

For Experiment 3, where the challenge is that parameters come from distributions, the results are similar - 90% confidence intervals are very narrow. For all parameters, the mean of the true underlying value is within the interval, including the two parameters that come from distributions. The difference here is that the model recognizes that the amount of noise needed to model the observations is around 65%, much larger than the added measurement noise of 10%. An interpretation is that if the measurement noise is well-constrained, the remainder of the noise can be attributed to uncertainty in the underlying model.

Conclusively, MCMC recovers the correct underlying model while being able to model the amount of combined model and data noise in the system.

4.3 prodCNN

The method is tested with a setup similar to Experiments 4 and 5, where the main challenge posed by the synthetic setup is the number of active terms, with 10 % noise added to the target. The exponents are from a uniform distribution, and the synthetic filter is from a standard normal distribution, passed through a sigmoid function and normalized to sum to 1, as described in Section 3. The filter is square with a side length of 5.

The results are shown in Figure 6. It is found that the parameters are retrieved correctly to approximately two decimals. The learned filter resembles the generating filter, with maximum deviations of 0.01. These results indicate that the method could successfully find this relation, should it be in the dynamics of the underlying physics. One grave assumption of this method is that the filter is the same for all glaciers, along the whole flowline and at all times.

4.4 SINDy

The synthetic test results for SINDy are shown in Figure 7. By trial and error, the amount of regularization was determined to yield the simplest model that resembled the synthetic model. The results for the first three experiments in panels a)-d) are on par with the results for NLSQ in terms of R^2. For all four, the dominant term in the expression for τ is $u^{1/3}$. Smaller terms are present in all four, fitting the data and model noise. All the smaller terms active are positive root exponents of either u and u_x, which are correlated with $u^{1/3}$.

For experiments 4 and 5, where the synthetic model is a product of many features, SINDy does not find a model that fits the data well. Due to how SINDy is defined, it cannot recover large products except if these are given in the dictionary of functions. While testing the method, it was attempted to increase the number of cross-terms

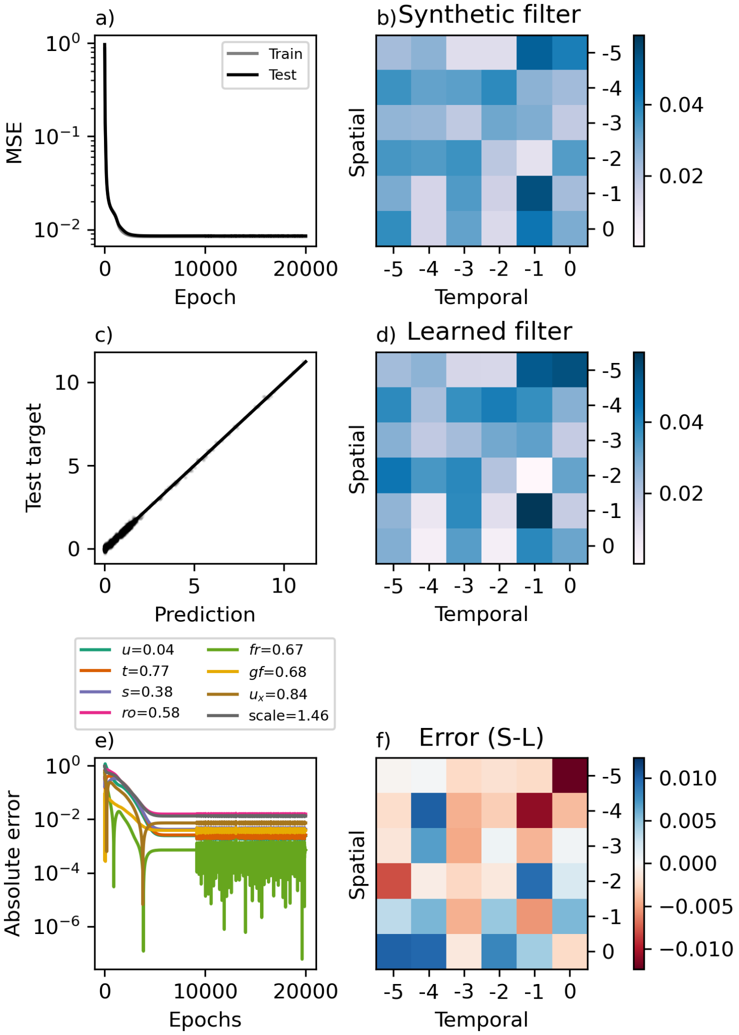

Figure 6: Synthetic test of the prodCNN model, where the functional form is similar to NLSQ, with the modification that the runoff is spatio-temporally convoluted at each point in time and space. The idea is to learn from data how upstream and earlier runoff can be used to predict basal stress. **a)** plot of decreasing losses during the training epochs (iterations). **c)** scatter plot of prediction and target, where the cloud is centred on the diagonal that represents a perfect fit. **e)** absolute error of learned parameter compared to true parameter, as a function of epochs. **b)** the synthetic filter used to generate data. All parameters are exponent on the variable equated to, except for scale. **d)** the learned filter from optimization. **f)** the difference between the synthetic and learned filter.

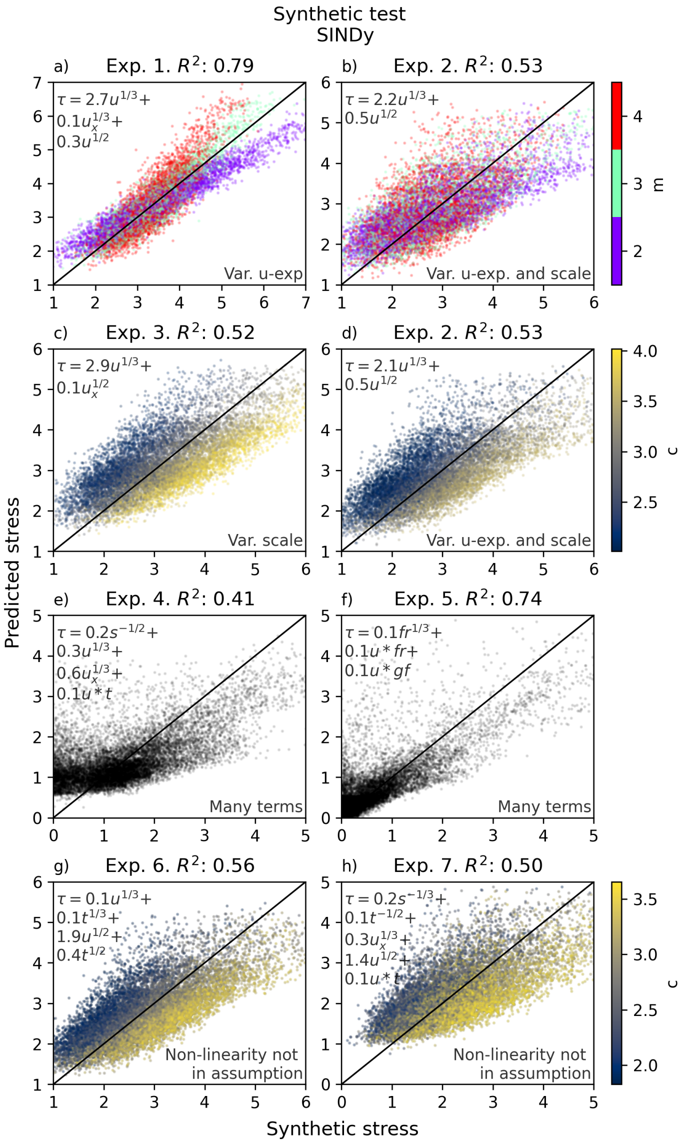

Figure 7: Results of synthetic test on seven experiments SINDy. The experiment numbers are detailed in Table 4, with a short explanation in the lower right of each panel of this figure. The resulting equations are in the upper left of each panel, with the displayed coefficients rounded to one decimal.

with different exponents. Due to the combinatorics and number of features, this quickly ill-conditioned the problem, and convergence issues emerged.

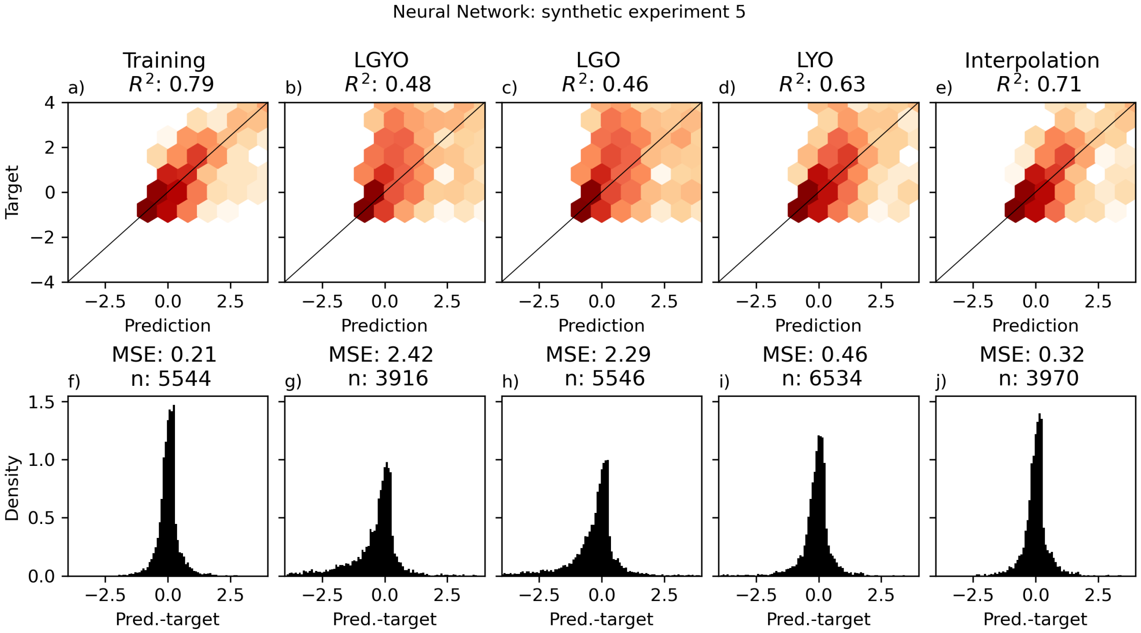

Figure 8: Synthetic test of FFNN approach, with synthetic experiment 5 (see Table 4). **a)-e):** density plots of target-prediction for training and test data sets. The diagonal line indicates a perfect fit. R^2 for each data set is in the subplot title. **f)-j):** error histograms for training and test data sets. MSE and the number of data points for each data set in the subplot title. LGYO (leave-glaciers-and-years-out), LGO (leave-glaciers-out) and LYO (leave-years-out) are three different aspects of extrapolation, which are divided to yield insight into the generalizability of a given model.

For experiments 6 and 7, where the synthetic model contains non-linearities that are not in the library, SINDy is somewhat successful in fitting the data trend, although the suggested model is off. However, for experiment 6, where NLSQ learns an almost correct model, except for the dependency on t, SINDy discovers that the model is dependent on u and t and yields a higher R^2 score than NLSQ. This indicates that even if the functional form discovered by SINDy cannot be interpreted by itself, the functional dependencies could reflect the true model if the model fits well to data.

4.5 FFNN

The method was tested in Experiment 5, where the main challenge posed by the synthetic setup is the number of active terms, with 10 % noise added to the target. The results are shown in Figure 8. The network fits well on training data, with an R^2 of 0.79. Interpolation and temporal extrapolation (LYO) have slightly worse fits with an R^2 of 0.63 and 0.71. Spatial (LGO) and spatio-temporal extrapolation (LGYO) yield R^2 scores of 0.48 and 0.26.

Compared to the parameter estimation methods, where the different test methods were almost identical (not shown), the FFNN gives very different results for training and the different test data sets. This indicates that while the FFNN has been tuned to fit the data, the model it has found is not similar to the synthetic generating model, underlying a general problem with NNs: their ability to extrapolate (and thus generalize) is limited.

Synthetic testing was further used to determine the complexity of the NN. The number of layers and neurons was increased simultaneously to find the hyper-parameters that yielded the lowest extrapolation losses. The optimum parameters were found to be 6 hidden layers with 48 neurons.

5 Results

Method	**Scale and exponents**								R^2					**Figure**
	scale	u	t	s	ro	fr	gf	u_x						
	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	Train	LGYO	LGO	LYO	Interp.	
Group A														
LSQ	1.66	-0.03	0.01	0.68	0.03	-0.10	0.01	0.28	0.19	0.01	0.01	0.21	0.21	A.22
NTLSQ	1.63	-	-	0.64	-	-	-	0.26	0.14	0.09	0.09	0.14	0.14	A.23
NLSQ	1.92	0.02	0.01	0.58	0.03	-0.08	0.01	0.26	0.25	0.01	0.01	0.25	0.25	9
NTNLSQ	1.89	-	-	0.47	-	-	-	0.27	0.21	0.08	0.08	0.21	0.20	A.25
NLSQ winter	1.90	0.01	0.00	0.63	0.05	-0.09	0.01	0.26	0.28	0.02	0.00	0.31	0.28	A.26
MCMC median	1.92	0.02	0.01	0.58	0.03	-0.08	0.01	0.26	0.25	0.01	0.01	0.25	0.25	-
MCMC 95%	1.93	0.03	0.02	0.59	0.03	-0.08	0.01	0.26	0.25	0.02	0.02	0.25	0.25	-
MCMC 5%	1.91	0.01	-0.01	0.57	0.03	-0.09	0.00	0.25	0.25	-0.01	-0.01	0.25	0.25	-
prodCNN	1.88	-0.08	-0.10	0.44	-0.01	-0.12	0.04	0.32	0.16	-0.12	-0.12	0.18	0.19	A.28
Group B														
SINDy	-	-	-	-	-	-	-	-	0.12	0.30	0.27	0.09	0.08	11
INLSQ median	2.14	-0.65	-0.44	0.01	0.00	-0.05	-	0.16	0.78	-	-	0.77	-	13
INLSQ $f(u, s)$	2.18	-0.76	-	-0.20	-	-	-	-	0.66	-	-	0.65	-	-
FFNN 1000 epochs	-	-	-	-	-	-	-	-	0.95	-0.86	-0.98	0.95	0.95	A.29
FFNN 5 epochs	-	-	-	-	-	-	-	-	0.92	-0.66	-0.73	0.90	0.92	12
NLSQ $f(u, s)$ + NN	1.82	0.31	-	0.45	-	-	-	-	0.94	-1.41	-1.54	0.93	0.93	A.31
Group C														
NLSQ $m = 3$	1.92	1/3	-0.17	0.55	0.03	-0.11	-0.02	0.10	0.20	0.06	0.04	0.19	0.18	A.32
NLSQ no strain	1.87	0.31	-0.11	0.55	0.04	-0.07	-0.02	-	0.20	0.12	0.10	0.18	0.17	-
NLSQ $f(u, s, fr)$	1.79	0.33	-	0.49	-	-0.08	-	-	0.17	0.16	0.14	0.16	0.15	-
NLSQ $f(u, s)$	1.82	0.31	-	0.45	-	-	-	-	0.15	0.24	0.22	0.13	0.11	10
NLSQ $f(u)$	1.74	0.17	-	-	-	-	-	-	0.03	0.14	0.12	0.04	0.02	-

Table 5: Summary of results basal stress estimation methods. The columns correspond to the scale and exponents of features in the basal stress formulations. An overview of the methods is shown in Table 3. For MCMC, the 5th, 95th and 50th (median) percentile are shown from 5 independent chains. For INLSQ, the medians of the retrieved exponent are shown. The results are grouped into three groups: Group A are methods that fit one model parametric basal stress model to all glaciers. Group B is SINDy plus methods that fit to individual glaciers or fit with non-symbolic models. Group C are variations of NLSQ with fixed parameters or using subsets of features.
The basal stress model is $\tau = m_0 \cdot u^{m_1} \cdot t^{m_2} \cdot s^{m_3} \cdot ro^{m_4} \cdot fr^{m_5} \cdot gf^{m_6} \cdot u_x^{m_7}$.
Key abbreviations are repeated. **LSQ**: linear least squares. **NTLSQ**: N-term linear least squares. **NLSQ**: non-linear least squares. **NTNLSQ**: N-term non-linear least squares. **MCMC**: Markov chain Monte Carlo. **SINDy**: Sparse Identification of Non-linear Dynamics. **INLSQ**: individual non-linear least squares. **FFNN**: feed-forward neural network. **LGYO**: leave-glaciers-and-years-out. **LGO**: leave-glaciers-out. **LYO**: leave-years-out.

The different basal stress estimation results are summarized in Table 5, where the re-

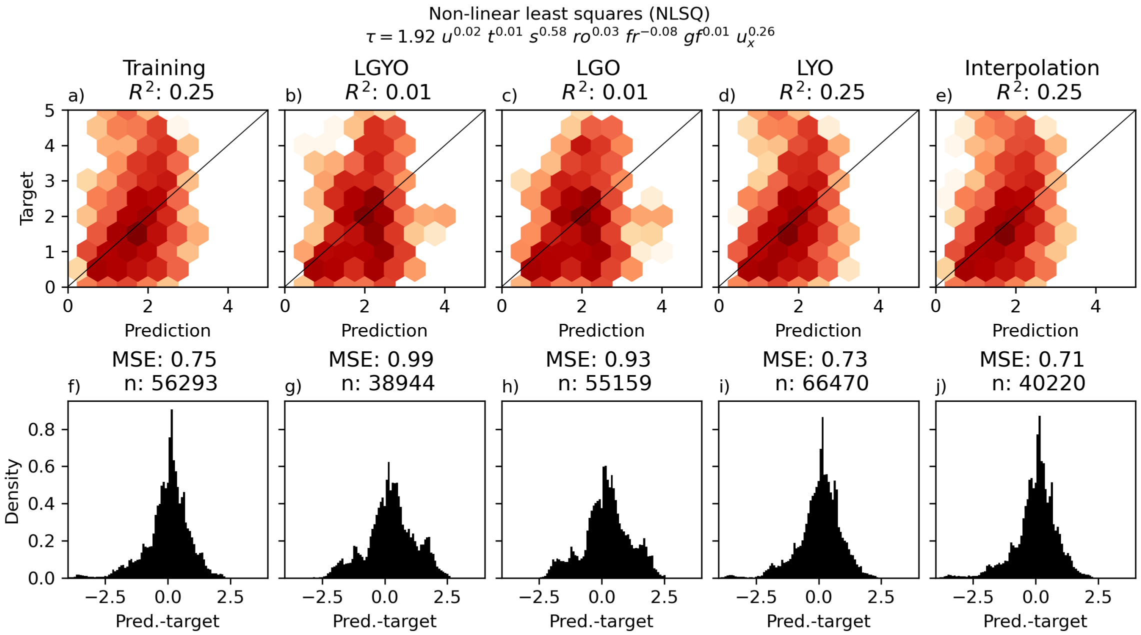

Figure 9: Results on training and test data for NLSQ. **a)-e):** density plots of target-prediction for training and test data sets. The diagonal line indicates a perfect fit. R^2 for each data set is in the subplot title. **f)-j):** error histograms for training and test data sets. MSE and the number of data points for each data set in the subplot title. LGYO (leave-glaciers-and-years-out), LGO (leave-glaciers-out) and LYO (leave-years-out) are three different aspects of extrapolation, which are divided to yield insight into the generalizability of a given model.

trieved coefficients (where applicable) and the training and test R^2 scores are shown. Visualizations of results are shown in the figures referenced in Table 5 (some figures have been left out for brevity). The figures show a density scatter plot of how predictions relate to targets and error histograms for all five training and test data sets. The figure style used for results will be explained in detail in Figure 9 and left out for the remainder of the figures in the same style.

The retrieved coefficients are similar across the different methods in Group A. For NTLSQ and NTNLSQ, where only two features are used in inversion yield slightly different models than the rest. The general picture is that the major active features are s, fr and u_x, with the rest of the exponents fluctuating around 0. The exponent for s spans 0.45 to 0.63, the exponent for fr spans -0.11 to -0.08, and the exponent for u_x spans 0.26 to 0.31. The R^2 vary across the different test data sets. Training R^2 range from 0.16 to 0.28, LGYO R^2 range from -0.12 to 0.09, LGO R^2 range from

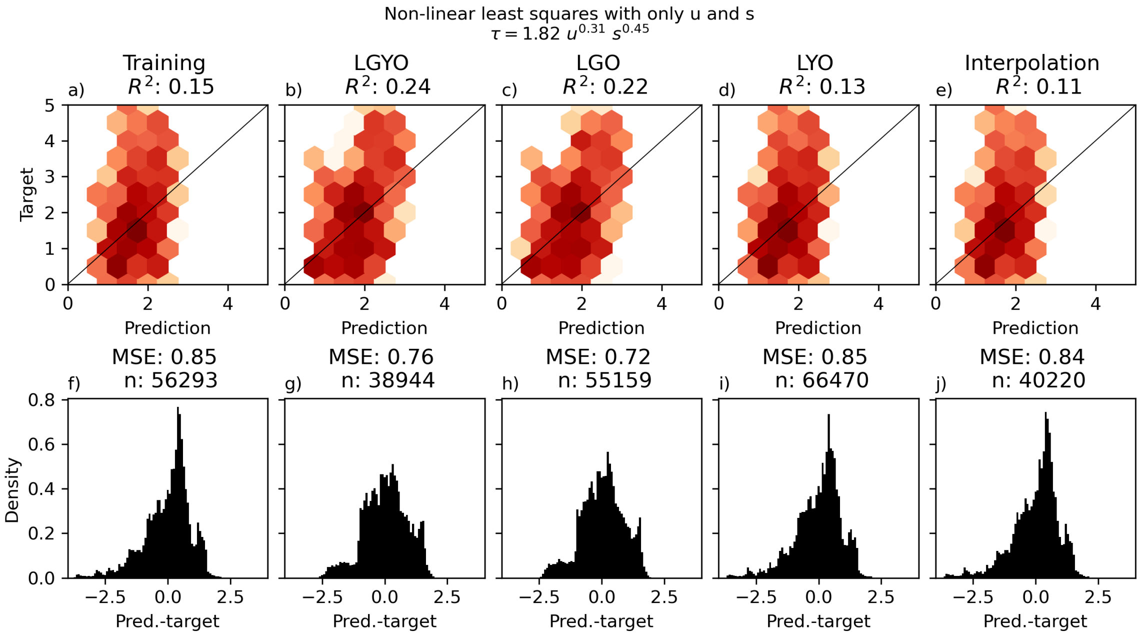

Figure 10: Results for NLSQ with only u and s as features. Explanation of figure in caption of Figure 9.

−0.12 to 0.09, LYO R^2 range from 0.14 to 0.31 and interpolation R^2 range from 0.14 to 0.28.

The MCMC 90% confidence interval spans a narrow for all model parameters, with deviations from the median of ±0.03 for the scale, and roughly ±0.01 for all exponents, yielding roughly the same R^2 scores. The MCMC samples are illustrated in Figure 16. Most pairs of parameters are uncorrelated. The exponent for u_x is negatively correlated with the exponent for u with a correlation coefficient of −0.74. Further, the exponents for gf and fr, and exponents for u and t are negatively correlated. Scale is positively correlated with the exponents for ro and fr. The exponents for ro and s, as well as the exponents for t and u_x are positively correlated. The estimated noise in MCMC, σ, has a value around 0.86-0.87.

The learned filter from prodCNN is seen in Figure 15. The lower right corner is the current grid point, with upstream and previous grid points in the rest of the filter. The filter is used in summing the runoff from previous and upstream points for inversion in each point. As such, the filter is to be interpreted as giving insight into which relative points in time and space have an impact on calculating basal

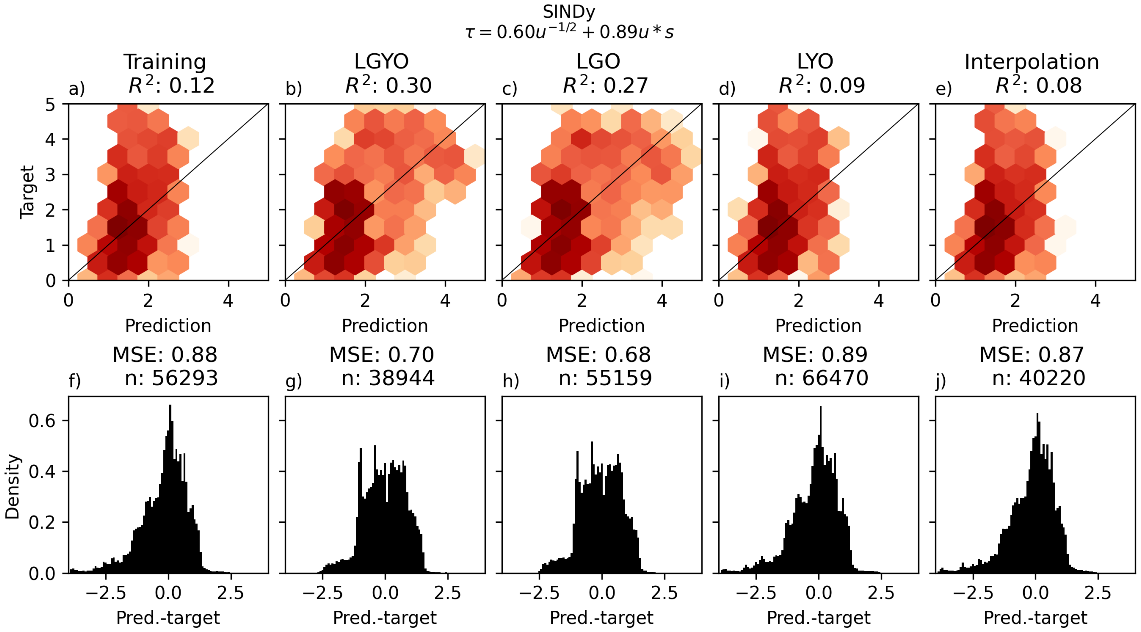

Figure 11: Results on training and test data for SINDy. Explanation of figure in caption of Figure 9.

stress at a given point. Clear spatial lines are present at temporal -1, -4, -10, -17 and -20. A cluster of points is seen from spatial -10 and upwards, around -8 to -13 temporal. The learned exponents are close to those of NLSQ, with an ro exponent of -0.01, putting very little weight on the filtered runoff.

The results from SINDy are shown in Figure 11. Here, the proposed model is a sum of two terms that are functions of features. The training, LYO and interpolation R^2 scores are around 0.10, with LGYO and LYO R^2 scores of 0.30 and 0.27.

For INLSQ, the R^2 scores are higher than all other parameter estimation methods. The parameters shown in Table 5 are the median of each parameter from the 34 glaciers, with the distributions and correlations across shown in Figure 17. The losses are calculated on the predictions with individual parameters for each glacier. Spatial test losses are unavailable, as each glacier was used in its individual inversion, and thus the learned models are not meant to extrapolate to other glaciers. Interpolation losses were left out to maximize the amount of training data available, as certain glaciers suffer from small amounts of data. The R^2 scores are 0.78 for training data and 0.77 for test data. In Figure 13 it is visually confirmed that this method yields

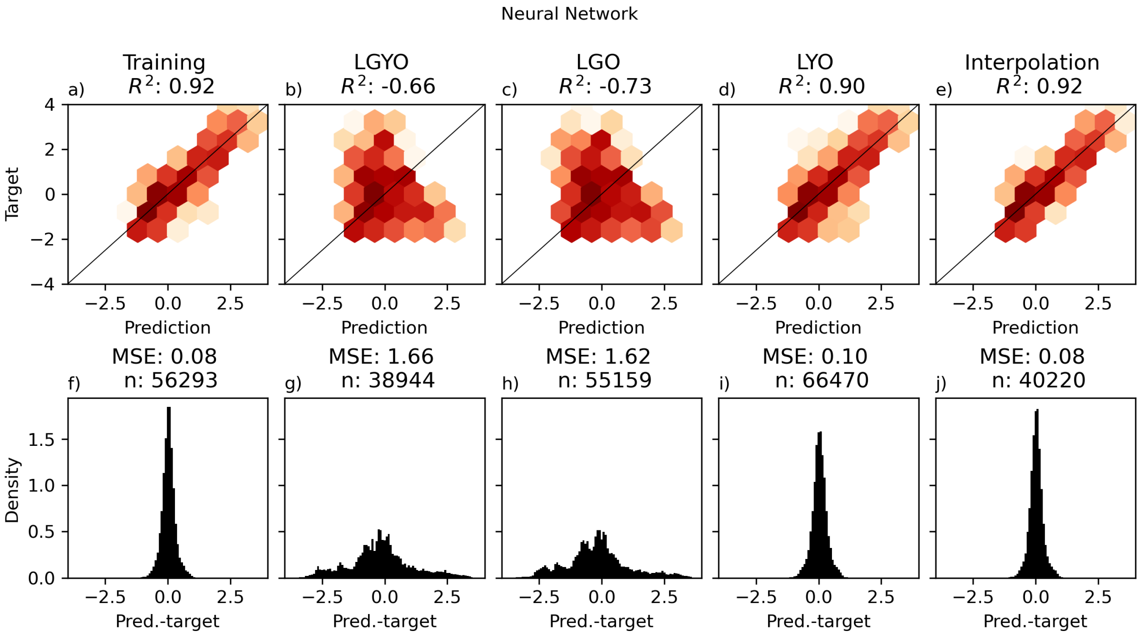

Figure 12: Results on training and test data for FFNN with 5 epochs. Explanation of figure in caption of Figure 9.

a better fit to observations than the other parameter estimation methods, where predictions are highly correlated with targets, and the losses are low compared to other methods.

For FFNN, the R^2 scores for training, LYO and interpolation loss are higher lower than for parameter estimation methods, at 0.92, 0.90 and 0.92, respectively. LGYO and LGO R^2 scores are among the lowest at -0.66 and -0.73. In Figure 12, it is clear that the learned model generalizes well to interior test points and temporal extrapolation, while it does not generalize to other glaciers as tested by LGO and LGYO. Overfitting is limited by early stopping, with only five training epochs. For 1000 epochs, the training, interpolation, and LYO fits get marginally better, while the LGYO and LGO fits get worse. Figure A.30 shows the evolution of losses with epochs. Applying FFNN to the residual when using NLSQ with only features u and s, the image is similar to other FFNN results. In Figure A.31, it is found that the training, LYO and interpolation R^2 scores are high, while LGYO and LGO are lower than in all other methods.

For NLSQ+NN, where only features u and s are used in NLSQ, the results are similar

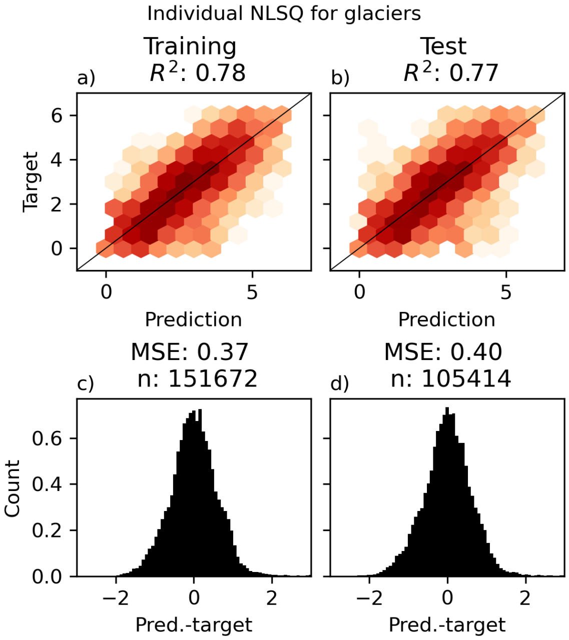

Figure 13: Results on training and test for INLSQ. **a)-b):** density plots of target-prediction for training and test data sets. The diagonal line indicates a perfect fit. R^2 for each data set is in the subplot title. **c)-d):** error histograms for training and test data sets. MSE and the number of data points for each data set in the subplot title. LYO (leave-years-out) is used for test data.

to FFNN, although with even lower R^2 scores for extrapolation.

Based on the correlation between observations and exponents of u and u_x as seen in Figures 3, 16 and 17, and the classic power-law suggesting a functional form with $u^{1/3}$, a row of different experiments based on NLSQ was carried out. The results are shown under group C in Table 5. In particular, the results for NLSQ with features u and s are shown in Figure 10. The exponents for u and s are 0.31 and 0.45, with extrapolation R^2 scores LGYO= 0.24, LGO=0.22 and LYO=0.13.

The optimal basal stress model determined by NLSQ is

$$\tau = 1.92 \cdot u^{0.02} \cdot t^{0.01} \cdot s^{0.58} \cdot ro^{0.03} \cdot fr^{-0.08} \cdot gf^{0.01} \cdot u_x^{0.26}. \tag{26}$$

Formulating the problem differently in SINDy, yields the basal stress model

$$\tau = 0.60 \cdot u^{-1/2} + 0.89 \cdot u \cdot s. \tag{27}$$

Finally, for the NLSQ method with only u and s as features, the basal stress model is

$$\tau = 1.82 \cdot u^{0.31} \cdot s^{0.45}. \tag{28}$$

Spatiotemporal distribution of errors for NLSQ is shown in Figure 14. The error is the MSE of all data (training and test) corresponding to the appropriate time and glacier. The MSE for different time points varies between ∼0.7 and ∼1.5. Generally, errors are highest in the middle of the year, where velocities are typically highest. The spatial distribution of errors varies between ~ 0.2 and ~ 6.5, with most glaciers below ∼1.0. The matrix plot of spatio-temporal errors shows for what glaciers and time points there are no data. Most data are missing during summer, where velocities are typically highest. There are a few glaciers that have full time-lines and a few that have a low temporal resolution. With the highest errors in the middle of the year, the results for NLSQ training without summer data (defined here as 1/5-1/10) are shown in Figure A.26.

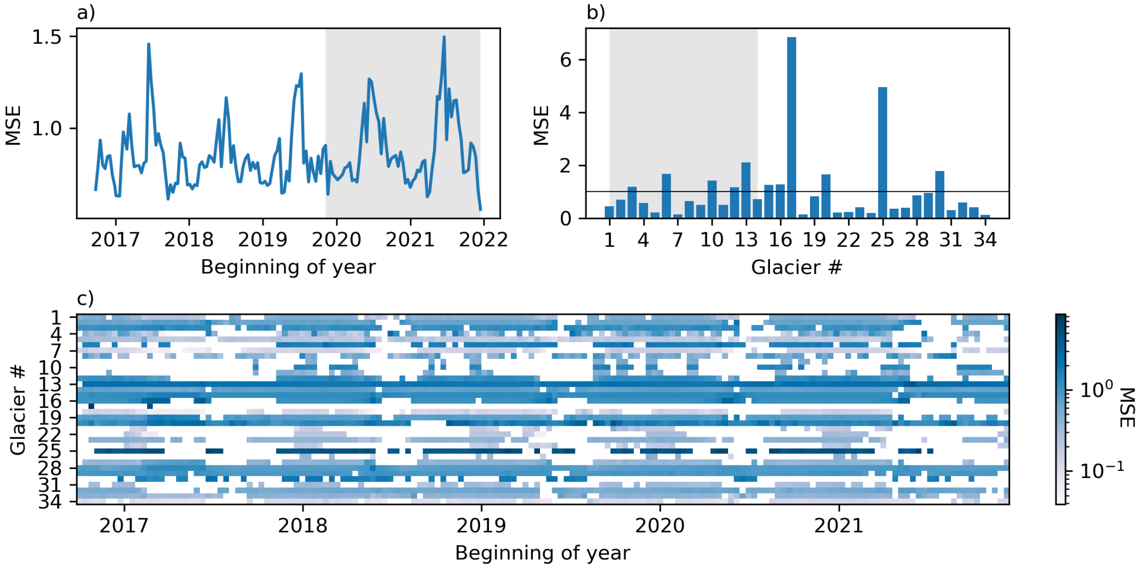

Figure 14: Spatiotemporal distribution of error for NLSQ. **a)** shows the MSE at each timestep. **b)** MSE for each glacier, with the horizontal line at MSE= 1. **c)** MSE at different glaciers at different times. The gaps in the dataset are due to the filtering explained in Section 2. The grey shaded areas in the upper panels illustrate data for extrapolation test losses.

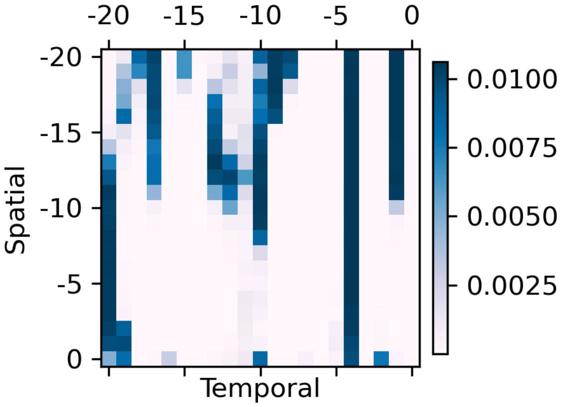

Figure 15: Learned filter from prodCNN. The spatial axis refers to grid points. The temporal axis refers to time points. Negative indices thus refer to upstream and earlier. The bottom right corner (temporal and spatial = 0) refers to the current point in time and space. The upper left corner (temporal and spatial = -20) refers to a point 5 km upstream (spatial grid spacing is 250 m) and 240 days earlier (temporal grid spacing is 12 d).

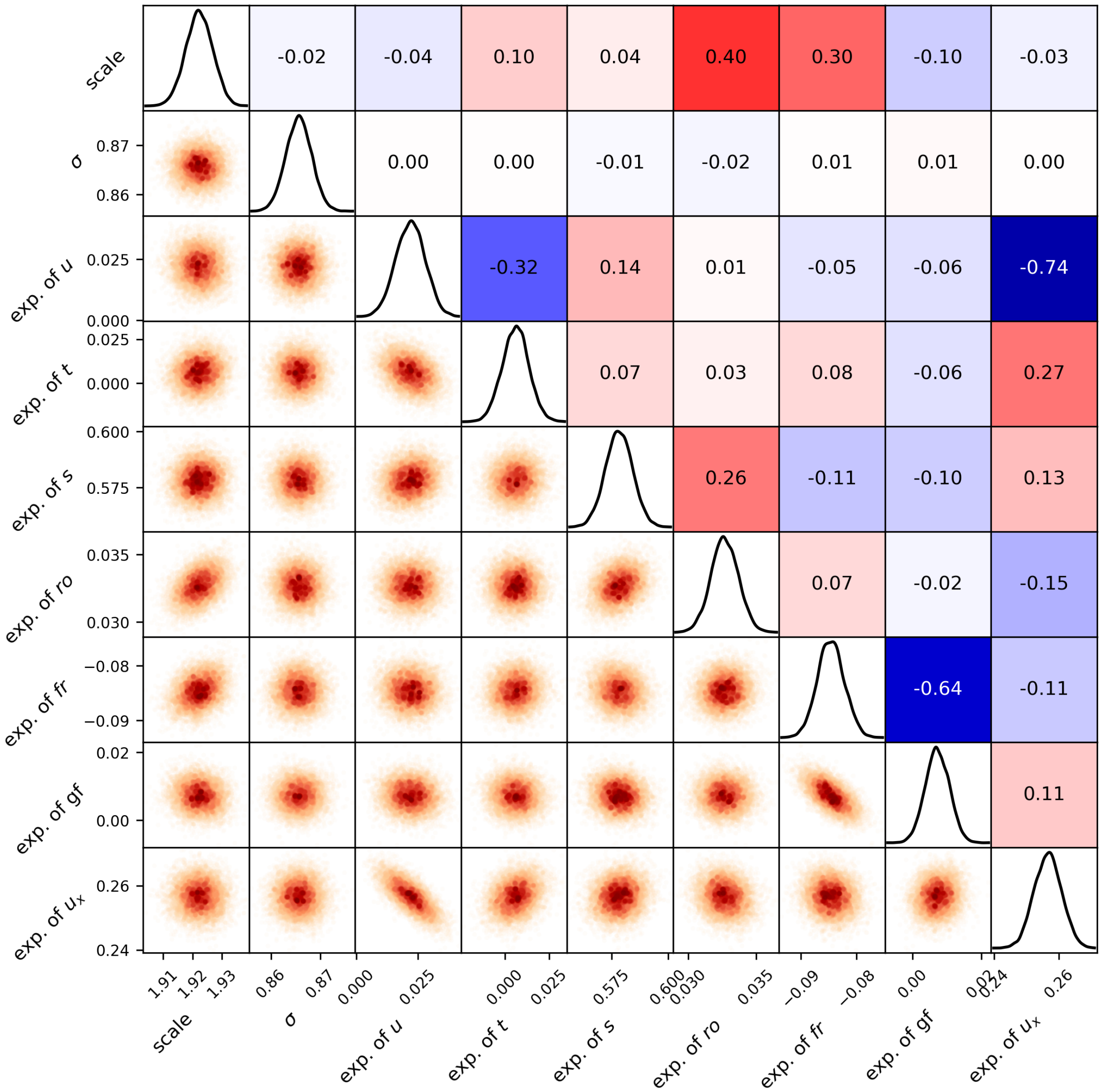

Figure 16: Samples of MCMC of the parameters. Aggregation of 5 chains, each with 2000 burn-in samples and 10000 samples. Diagonal shows distributions as KDE. Above the diagonal, the correlation between parameters is shown, with blue colours signifying negative correlation and red colours positive correlation. Below the diagonal, each pair of parameter samples are plotted against each other, where darker colours indicate higher densities.

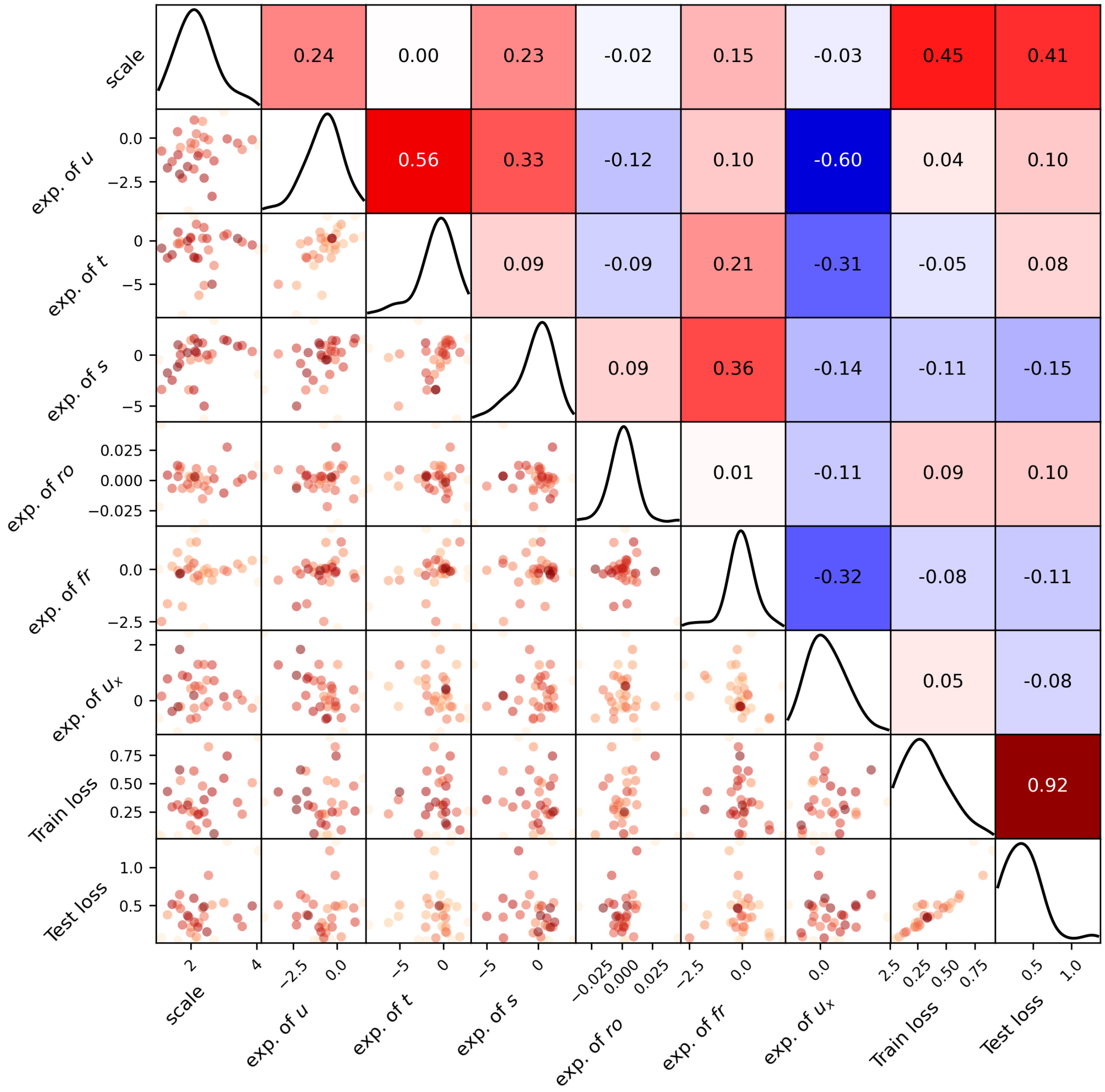

Figure 17: Distribution of parameters for individual NLSQ, along with the distributions of training and test loss.

6 Discussion

6.1 Interpretation of results

This study aims to learn a model for basal stress based on input features. From the proposed models summarized in Table 5, the significant features relevant are surface elevation s and strain rate u_x, which consistently have the largest magnitudes of exponents across the different methods in Group A (all features used).

The commonly assumed basal stress power-law (Eq. 2) with $m = 3$ is not found for any method with the full set of features. The exponent for u in the basal stress model ranges from -0.08 to 0.03 in Group A (see Table 5). However, as is seen in Figure 3 and Figure 16, both observations and exponents for u are negatively correlated with u_x. This is clear for NLSQ $m = 3$, where the exponent for u_x drops to 0.10 from 0.26-0.33 in the other methods.

NTLSQ found the optimal parsimonious model with two terms, s and u_x, as shown in Figure A.24, where the LGYO loss is lowest with two active terms. u and u_x, as well as their exponents in MCMC samples, are correlated as seen in Figure 3 and 16. To investigate this effect, an attempt was made with NLSQ with only u and s as features. The results are shown in Figure 10. The training R^2 is on par with the lowest in the Group A results, while the spatial extrapolation R^2 scores (LGO and LGYO) are the highest across all results in Group A. Further, the exponent of u is found to be 0.31, which is very close to the commonly assumed $1/m = 1/3$ in Eq. (2).

Test data for the synthetic test of NLSQ with a scale spanning 2-4 and an exponent spanning 1/4-1/2 yielded an R^2 score of 0.53 (see Table 4, experiment 2). In that case, the method recovered a model that could be interpreted as a mean model of the generating data. The LGYO test R^2 for NLSQ with features u, s was 0.24. The results from synthetic testing thus indicate that more is missing from the dynamics learned from NLSQ than just the correct exponents and scales - or that the underlying values span more extensive ranges than in synthetic testing.

Basal melt from friction is persistent across all models, with a slightly negative exponent. However, comparing NLSQ $f(u, s, fr)$ and NLSQ $f(u, s)$, it is seen that including fr in inversion yields a worse extrapolation loss, indicating that the result is not generalizable to all glaciers. Models are generally not dependent on geothermal basal melt and surface runoff; the learned exponents for these quantities are close to 0 for all methods. The conclusion is that no dependency was found between basal stress and meltwater availability.

INLSQ, where NLSQ is applied individually to each glacier, offers a different look at how exponents are distributed and related. Where MCMC results fundamentally give the NLSQ solution, with distributions for each parameter yielding a confidence interval, INLSQ results give insight into how different parameters relate to each other across different glaciers. Interestingly, the exponents of u and t are positively correlated, with a correlation coefficient of 0.56. This is the opposite of the result of MCMC, where the two parameters are negatively correlated.

A major difference between INLSQ and NLSQ is the exponent for u. Generally, the exponent in the power-law formulation Eq. (2), is assumed positive and around $1/m = 1/3$ (e.g. Cuffey and Paterson, 2010; Schoof, 2007). While the results for NLSQ with features u or u, s agree with this, the results for INLSQ contradict this, with the median exponents retrieved being negative and of higher magnitudes, at -0.66. A negative exponent for u in a power-law rheology implies that higher velocities are found with lower basal stresses, which counters the general interpretation that the bed of a glacier provides resistance to flow.

Figure A.36 presents a visualization of how this contradiction occurs. The overall trend between velocity and basal stress is positive (shown in log-space), while for most glaciers, the trend is negative. The interpretation of this is that on an ice-sheet level, the general relationship between velocity and basal stress is positive, while on an individual glacier level, the general relationship is negative. A histogram of exponents is shown in Figure A.37.

The results from prodCNN show interesting patterns, seen in the filter in Figure 15. It seems that a region around 8-20 grid points (corresponding to 2-5 km upstream) and 8-13 spatial points (corresponding to 96-156 days earlier) is consistently present in the filter. This is interpreted as runoff at this point in space and time is part of the basal stress function in this method, in line with general assumptions that water availability upstream and earlier could impact the basal stress in a given point (Jay-Allemand et al., 2011). However, it is noted that the exponent for runoff in this method is small in magnitude at a value of -0.01. This is comparable to other methods, with the filter designed to sum to 1 to facilitate the comparison of the exponent. The low value of the exponent makes the filter poorly constrained by data, as very little weight is given to variations in the filter. However, on different runs, the filter would converge to the same features as seen in Figure 15. This indicates that even if the signal is insignificant, it is persistent and could be used as a starting point for further studies of the spatio-temporal dependence between basal stress and runoff.

The fundamental assumption of prodCNN is that the relation between basal stress to runoff at upstream and earlier times is the same across all glaciers at all time points. This is highly unlikely, given the different glaciers' varying topographies and dynamics. Alternatively, the spatial aspect could be considered by routing the runoff along topography. One approach is to assume that runoff sinks vertically and is then routed via the basal topography. A different approach is to assume that water is routed via the surface topography and then sinks vertically. Either of these two could be combined with learning a temporal filter, individual for each glacier, that calculated basal stress based on the temporally filtered data.

A different approach would be to apply prodCNN individually for each glacier, learning filters that are potentially different for different glaciers. This would allow insight into the dynamics of specific glaciers, dropping the assumption that the runoff has the same spatio-temporal pattern of impact on basal stress across all glaciers.

The highest training, LYO and interpolation R^2 scores are found for methods with NNs, while these also yield the lowest LGYO and LGO R^2 scores. Even for just a few training epochs, the spatial extrapolation R^2 scores are lower than any other method. This does not, however, allow denying the fact that there exists a basal stress model that extrapolates to other glaciers; all basal stress was generated from a function, as shown in Eq. (9), where the target basal stress is calculated as the sum of extensional and driving stresses. The fact that the NN methods have not found this relation could be interpreted as an expression of NNs flexibility - the NNs have prescribed another function that fits very well within the training glaciers (training, LYO and interpolation) but does not extrapolate to the test glaciers (LGYO and LGO). Another interpretation is that the NNs are not flexible enough to represent the generating basal stress model in Eq. (9). Combining NLSQ and FFNN did not change results significantly. The interpretability of neural networks limits the application of FFNNs for discovering dynamics.

The results from SINDy are from the parsimonious model determined by Figure A.27, where the losses have been plotted as a function of regularization. The suggested model in Eq. (27) is among the simplest models discovered in this study and has the highest extrapolation R^2 scores of any model in this study. Due to its functional form, a sum of two functions of features, it cannot be directly compared to the extended power-law formulations; however, its dependencies are similar to that of the highest scoring NLSQ model, with features u and s. To compare the two models, an illustration is provided in Figure 18. The two proposed models have a similar functional form for most observations, as indicated by the histogram. Where NLSQ, following the power-law model, has a fixed sensitivity to u, the SINDy model implies that very low velocities yield a high resistance, with the basal stress decreasing as velocity reaches ~ 0.3, after which it approaches linearly increasing with u. This implies a more complex relationship between basal stress and u, and thus hydrological and bed properties, than typically assumed under the power-law rheologies. However, it is noted that the majority of observations constraining the functional form are for velocities above the low velocities, for which this more complex relationship between u and τ is found. The discovered models are most constrained where the histogram is dense.

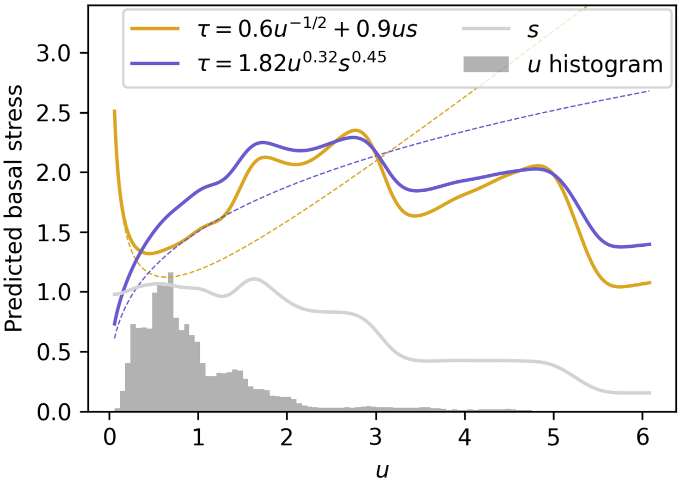

Figure 18: Function illustration of the two best extrapolating models discovered, using SINDy and NLSQ. The predicted basal stress is plotted for velocities. The surface topography is provided for corresponding values to velocities, smoothed with a Gaussian filter. A histogram of observed velocities is provided in grey, with the density values following the y-axis. Functions for a constant s are provided as dashed lines.

The amount of regularization determined by Figure A.27 shows that the test loss is lower than the training loss, where the test loss is at its minimum. The training loss is comparable to a baseline model with an MSE of around 1.00. It is seen for other results in Table 5 that the test losses are lower than training losses. This would not have been possible if the features and targets came from similar distributions. However, this further indicates that the dynamics of the glaciers across Greenland vary and that prescribing a unified basal stress law is difficult without tuning it to local conditions.

6.2 Spatial and temporal errors

The abilities of the methods to extrapolate spatially, seen in particular in the LGYO and LGO columns of Table 5, prove against the hypothesis that a single basal stress function of observations could be learned that generalizes across space and time. From Figure 14, it is seen that there is a seasonal signal to the error in time, with mid-year errors higher than the rest of the year. This is likely related to the combination of higher velocities and thus higher stresses in summer, along with more noise in data and thus more data gaps in summer, leading to an imbalanced data set where the non-summer dynamics are overrepresented. A winter analysis was carried out, using only winter data, with results shown in Figure A.26. The retrieved basal stress function is close to that of NLSQ, with decreased losses, reflecting that the under-represented summer data are left out.

From Figure 14, it is further seen that there is a large variety in how the NLSQ model fits to different glaciers. To get insight into how the loss varies across the different glaciers, the predictions and targets for the four glaciers with the highest and lowest MSEs are plotted in Figure A.35. The general picture is that there seems to be low to no correlation between target and prediction across the glaciers, even for those that have low MSEs. The seemingly good results seem to be related to the scale of targets rather than the quality of fits. This was further confirmed when recreating Figure 14, where the loss was normalized with the standard deviation of targets, with the result that the bar plot of glacial errors was much more even across glaciers (not shown).

The spatio-temporal errors of INLSQ are shown in Figure A.33, where almost all glaciers have MSEs below 0.5. Plotting the four highest and lowest MSE glaciers in Figure A.34, it is clear that the low MSE glaciers show good fits that are not just due to small values but with clear correlations between target and prediction. The variety in the learned functions for basal stress makes it clear that there are very different dynamics at play for the different glaciers. Distributions of the various parameters and losses across glaciers are shown in Figure 17.

6.3 Relation to other results

Maier et al. (2021) use three different complexities of ice-flow models to investigate the relationship between velocity and basal stress of Greenland glaciers. The models are run in different catchments, dividing the ice sheet into eight separate regions, with individual inversions in each region. The results are interpreted in terms of bed strength, based on the exponent in the power-law relationship that this study is based upon. Generally, the results are consistent across the different model complexities (in order of decreasing complexity: Full-Stokes, SSA and shallow ice approximation).

They find that the average value across the eight catchments is $1/m = 0.34$ (denoted $1/p$ in Maier et al. (2021)). For the best extrapolating model in this study (NLSQ with features u and s), it was found that $1/m = 0.31$. However, INLSQ's median exponent for u was $1/m = -0.65$. Running INLSQ with only features u and s yielded a median of $1/m = -0.76$, with a similar result for running it with just u. Maier et al. (2021) finds positive exponents in all catchments.

In Maier et al. (2021), the value of $1/m$ is interpreted in terms of bed strength and hydrologic properties, with $m = 3$ for hard-bed sliding and higher values for sliding over deformable beds. These studies give no analysis of negative exponents for u in the basal stress law, and more analysis is necessary to understand the implications of this result.

Winton et al. (2022) use an inverse approach to infer the basal stress coefficient (c in Eq. (4)), as a spatio-temporal variable, during a surge of Hagen Bræ. While there are theories that surges of glaciers are related to meltwater (Sevestre and Benn, 2015; Benn et al., 2019, e.g.), they find no relationship between the basal stress coefficient and the amount of surface meltwater for Hagen Bræ during the years 2015 to 2019. This is in line with this study, where none of the methods applied yield a strong dependency between runoff or basal melt features and basal stress, with the weak dependency on friction melt yielding poor extrapolation results. Many other studies (e.g. Jay-Allemand et al., 2011; Stevens et al., 2022), however, do find relations between basal stress and meltwater. This underlines that more complex ice-flow models and data methods might be needed to uncover this relation on an ice-sheet scale.

6.4 Data quality

Within machine learning, the quality of data is of high importance. In this study, the goal is to find a constant function that fits to a spatiotemporally varying target, where the predictive temporal variation comes from temporal variation in the used features. However, of the data sets used, only the velocity and regional climate model (RCM) data have a temporal component. With the RCM data reduced to one variable due to correlation, this leaves two temporally variable data sources, yielding three features (with strain rates a derived feature from velocity.)

The data used for basal melt from Karlsson et al. (2021) is a first constraint rather than a finely tuned model. Thus, the data provided are not well-constrained by observations and are an attempt to quantify magnitudes rather than specific spatially variable actual melt. Its use in this study has been as the latter, which might be the reason why minimal dependency was found between basal stress and basal melt, which is commonly assumed to be related (Cuffey and Paterson, 2010).

6.5 Suggestions for improvements and further work

During this study, many considerations have been made about the limitations and shortcomings of the applied methods and assumptions. Here, some critical points will be discussed, along with thoughts about how to develop the results obtained in this study. Finally, a very different view of the problem is presented, which constitutes the essence of SciML.

6.5.1 The proposed basal stress model

In this study, the assumed functional form of the basal stress model is similar to the power-law in Eq. (2) where other features are included in a similar form as u. While the power law is based on physical considerations and assumptions, the proposed functional form is not. A more rigorous approach to deriving a physics-based basal stress law with more observables than u could shed more light on how the observed features affect the basal stress. Alternatively, a less restricted functional form might reveal other relationships between variables at the cost of interpretability.

A further restriction of the assumed basal stress model is that it requires positive features. This eliminated the feature related to basal melt from surface water percolation and refreezing, which has variability in sign across the domain. Further, it was necessary to remove all points with negative strain rates. Finally, as much of the runoff is 0 through most of the year, it required the temporal summing of the feature to have data for a more representative part of the year. Other more flexible functional forms could come about these issues, allowing for a greater variety in features.

One approach to extending the functional form could be using other non-linear functions in a similar product-form as in Eq (4), that do not restrict the input to positive values. One such idea could be learning FFNNs for each variable in a formulation like

$$\tau = m_0 \cdot NN(u) \cdot NN(t) \cdot NN(s) \cdot NN(ro) \cdot NN(fr) \cdot NN(gf) \cdot NN(u_x), \quad (29)$$

where NNs are feed-forward neural networks that are independent of each other. Each NN would be more easily interpreted as they are just functions of one variable. Thus, it would be easier to identify the dependence of specific features than was the case in the FFNN method presented in this study.

Finally, the proposed functional form could be extended by using more data sets as features or creating physical features from other observations. As described earlier, the effective water pressure N is commonly seen in applied basal stress laws. Under assumptions of the bed being hydrostatically connected to the ocean, N could have been included as a feature. Other features could have been a measure of bed roughness on different scales, the slope of the bed topography, and a measure of bed till strength. In particular, more temporally variable features could help uncover the temporal variation in basal stress.

6.5.2 Ice-flow model

The applied ice-flow model (SSA) could be extended with lateral drag, making the basal stress more well-constrained (Veen et al., 2011). Further, it could be an idea to tune the rheological temperature-dependent coefficient A to better approximate local conditions. This tuning could be done deterministically, with data on surface temperatures, thicknesses and temperature profiles where available. A was assumed constant for all glaciers in this study, which is a grave assumption, given the geographical and topographical variety of the glaciers.

One way to incorporate the variability of glacier dynamics across Greenland is to do the analysis done in this study on a catchment scale, as seen in Maier et al. (2021), or by classification of seasonal velocity patterns as in Vijay et al. (2021). This would allow discovering different dynamics for different groups of glaciers while still allowing testing of how well a model generalizes to other glaciers.

6.5.3 Synthetic testing

The synthetic testing of FFNN could be improved by increasing the number of synthetic data to the complete data set size, allowing better comparison between synthetic and data applications. With the current setup, where the size of the data set is around ten times larger than the synthetic data set, it was found that fewer epochs were needed to converge for the data application than synthetic application. This is related to the batch training, where 1024 data points were used in each batch, with a full epoch being counted when all data has been through training once. Further, the size of the network, in terms of nodes and layers, was tuned based on the synthetic data set, which further argues for having similar amounts of data in synthetic and real data applications.

The synthetic tests of SINDy were carried out to test its ability to discover the functional form of basal stress in Eq. (4). In order to get a more tailored test of SINDy and its sensitivities to model and data noise, it would have been beneficial to create a similar array of experiments as presented in Section 4, but where the functional form was a linear combination of the library of candidate functions. Thus, the synthetic testing here does not generalize to other uses of SINDy but was used merely to test SINDy under the assumed functional form of basal stress used in this study.

6.5.4 Train and test data

In general, LYO and interpolation R^2 scores are on par with training R^2. This indicates that the models do not overfit to the training data but generalize to interpolation and temporal extrapolation. The temporal extrapolation success across all methods indicates that the dynamics learned from training are similar in the extrapolation period, seen in Figure 14 to be roughly the two years 2021-2022.

The results were somewhat sensitive to which glaciers were left out for testing. For some methods in Table 5, the R^2 scores were higher for LGYO and LGO than for training. In testing the methods, a random split was used in each iteration, with the consequence that the retrieved models and R^2 varied. Subsequently, and for all results presented in this report, a fixed bias-free split was determined by holding out the first 14 glaciers (alphabetically) for testing. It could be argued that more data should be used for training, but due to the variety in dynamics and scales of stresses across the different glaciers, it was chosen to retain a rather large set of glaciers for testing in order to test against a variety of glacier dynamics.

Another type of test data set could be temporal and spatial interpolation data sets, where, e.g. the middle 20% of grid points for each glacier, or the middle 20% of time points across all glaciers, could be held out for testing. This would further evaluate the models' ability to predict on unseen data and might reveal strengths or shortcomings of the models not discovered so far.

In this study, a relatively simple approach has been taken to split the data, with one training data set and four test data sets, where the splits have been fixed. A more comprehensive approach might yield better models and better estimations of errors. One such approach is k-fold cross-validation, where rather than having static divisions, different splits are made, training and evaluating the models at the different splits, and taking the average of training and test errors across the different splits.

Finally, it could be argued that a final test data set should be held out for the final evaluation, after which no further analysis is done. The different methods applied have been guided by changes in the test losses, and thus the test losses shown are not unbiased but have been used for hyperparameter tuning.

6.5.5 Combining deterministic models and neural networks

This study suggested different methods combining deterministic models and neural networks. While the synthetic tests proved promising, application to real data did not turn out to generalize. This section presents a different approach to learning about the underlying model. Previously, the stress balance in Eq. (1) was used to determine the basal stress term from data using finite differences. The equation is now considered a differential equation with dependent and independent variables u and x.

First, the stress balance is nondimensionalized, almost mirroring that of Tsai et al. (2015) and Schoof (2007). Assuming positive strain rates, the stress balance is

$$\varepsilon \left(\hat{h}\hat{u}_{\hat{x}}^{1/n}\right)_{\hat{x}} - \hat{\tau} - \hat{h}\left(\hat{h}-\hat{b}\right)_{\hat{x}} = 0, \quad u(0) = u_0, \quad u_x(0) = u_{x,0}, \tag{30}$$

where hats denote scaled variables and the dimensionless parameter ε is defined

$$\varepsilon = \frac{\left(2[u][x]^{-1}A^{-1}\right)^{1/n}}{\rho g[h]}, \tag{31}$$

with the scalings $[u], [x]$ and $[h]$ are determined from data. Two further simplifications are made to make the ODE easier to solve, allowing focus on the model discovery aspect. First, it is assumed that the relation between the dominant shear stress and the corresponding shear strain rate is linear by setting the rheological exponent $n = 1$. Second, it is assumed that $h_x u_x \ll h u_{xx}$, allowing $(hu_x)_x \simeq hu_{xx}$.

$$\varepsilon h u_{xx} - \tau - h\left(h-b\right)_x = 0, \tag{32}$$

where hats denoting scalings have been dropped. Rearranging yields the non-linear (assuming that τ is a non-linear function of u) ODE in u on a form suitable for standard ODE solvers

$$u_{xx} = \frac{\tau + h(h-b)_x}{\varepsilon h}. \tag{33}$$

The rhs of this equation can then be learned fully or partially from data in a framework presented by Rackauckas (2019) as Universal Differential Equations (UDE). The problem of fully learning the operator could be posed as

$$u_{xx} = NN\left(u, t, s, ro, fr, gf, u_x\right), \tag{34}$$

where the NN attempts to learn the rhs of Eq. (33).

Alternatively, the dynamics can be learned partially from data, where the known dynamics are kept in the problem, and unknown dynamics are replaced by a NN

$$u_{xx} = NN\left(u, t, s, ro, fr, gf, u_x\right) + \frac{h(h-b)_x}{\varepsilon h}, \tag{35}$$

from which the basal stress dynamics can be found as

$$\tau = NN\left(u, t, s, ro, fr, gf, u_x\right)\varepsilon h. \tag{36}$$

This change in perspective compared to earlier allows for learning the dynamics of the system differently. Rather than fitting to the basal stress determined by a model that does not capture all dynamics (SSA), the basal stress and the system's unknown dynamics could be learned. Applying embedded NNs within differential equation solvers could yield insight into ice flow dynamics that have thus far been undiscovered. Preliminary testing of the method to learn synthetic dynamics in both formulations shown above showed promising results, as long as the system was noise-free. Adding noise to the synthetic observations of velocity leads to convergence issues. Further testing of the method would give insight into its prospects and possibly its application to the questions and data presented in this study.

7 Conclusions

A range of methods was applied to discover the relationship between basal stress and physical features, provided by observational records and models. Topographies, surface velocity and quantities related to basal melt and surface runoff were used as features in frameworks inspired by the development of scientific machine learning, where physical models and data are combined in methods for system identification and parameter estimation. Models were trained on a subset of data, with four different test data sets held out for testing various aspects of generalizability.

Synthetic testing of methods showed promising results when subject to model and data noise. Even under high model and data uncertainty, recovering the correct functional form and dependency on features was possible.

For the extended power-law formulation, the best results on extrapolation data sets used only velocity u and surface topography s, with the model $\tau \propto u^{0.31}s^{0.45}$ explaining 24% variance as measured by R^2. This model is in line with the commonly assumed hard-bed rheology of the power-law. Improvements were made when fitting models to glaciers individually, where the models yielded an explained variance of 77% in temporal extrapolation. This indicates that the glaciers have different dependencies on the features and that a general basal stress law cannot be determined without spatio-temporal tuning to reflect local dynamics. Other dynamics were uncovered using SINDy, where the proposed model $\tau \propto u^{-1/2} + 1.5us$, yielded the highest R^2 scores for spatio-temporal extrapolation, explaining 30% variance.

No dependency on meltwater was uncovered with any method, as validated by extrapolation fit, and more sophisticated methods might be necessary to uncover the relationship between basal stress and meltwater. While a feed-forward neural network could learn the dynamics of training glaciers with good results on temporal extrapolation, it could not extrapolate the learned dynamics to other glaciers in Greenland.

The applied methods in this study uncovered some of the basal stress dynamics observed. Further application of scientific machine learning could uncover missing ice-flow dynamics and allow for further discovery of basal dynamics.

References

J. K. Andersen, R. S. Fausto, K. Hansen, J. E. Box, S. B. Andersen, A. P. Ahlstrøm, D. van As, M. Citterio, W. Colgan, N. B. Karlsson, K. K. Kjeldsen, N. J. Korsgaard, S. H. Larsen, K. D. Mankoff, A. Pedersen, C. L. Shields, A. Solgaard, and B. Vandecrux. Update of annual calving front lines for 47 marine terminating outlet glaciers in Greenland (1999-2018). *Geological Survey of Denmark and Greenland Bulletin*, 43, 7 2019. ISSN 19044666. doi: 10.34194/GEUSB-201943-02-02.

R. C. Aster, B. Borchers, and C. H. Thurber. *Parameter Estimation and Inverse Problems*. Academic Press, second edition edition, 2013. ISBN 978-0-12-385048-5. doi: 10.1016/B978-0-12-385048-5.00010-0. URL http://www.sciencedirect.com/science/article/pii/B9780123850485000100.

D. Benn, A. Fowler, I. Hewitt, and H. Sevestre. A general theory of glacier surges. *Journal of Glaciology*, 65:1–16, 4 2019. doi: 10.1017/jog.2019.62.

H. B. Bingham, P. S. Larsen, and V. A. Barker. *Computational Fluid Dynamics*. Polyteknisk Forlag, 2020.

C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, 2006. ISBN 0387310738.

J. Bolibar, A. Rabatel, I. Gouttevin, C. Galiez, T. Condom, and E. Sauquet. Deep learning applied to glacier evolution modelling. *Cryosphere*, 14:565–584, 2 2020. ISSN 19940424. doi: 10.5194/tc-14-565-2020.

D. Brinkerhoff, A. Aschwanden, and M. Fahnestock. Constraining subglacial processes from surface velocity observations using surrogate-based Bayesian inference. 6 2020. doi: 10.1017/jog.2020.112. URL http://arxiv.org/abs/2006.12422http://dx.doi.org/10.1017/jog.2020.112.

S. L. Brunton, J. L. Proctor, J. N. Kutz, and W. Bialek. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences of the United States of America*, 113:3932–3937, 2016. ISSN 10916490. doi: 10.1073/pnas.1517384113.

E. Bueler. Numerical Modelling of Ice Sheets, Streams, and Shelves, 2021. URL https://doi.org/10.1007/978-3-030-42584-5_8.

E. Bueler and J. Brown. Shallow shelf approximation as a ”sliding law” in a thermomechanically coupled ice sheet model. *Journal of Geophysical Research: Solid Earth*, 114, 3 2009. ISSN 21699356. doi: 10.1029/2008JF001179.

K. Champion, B. Lusch, J. N. Kutz, and S. L. Brunton. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences of the United States of America*, 116:22445–22451, 11 2019. ISSN 10916490. doi: 10.1073/pnas.1906995116.

J. A. Church and N. J. White. Sea-Level Rise from the Late 19th to the Early 21st Century. *Surveys in Geophysics*, 32:585–602, 9 2011. ISSN 01693298. doi: 10.1007/s10712-011-9119-1.

K. M. Cuffey and W. S. B. Paterson. *The Physics of Glaciers*. Elsevier Science, 4th edition, 2010. ISBN 9780080919126.

G. Cybenkot. Mathematics of Control, Signals, and Systems Approximation by Superpositions of a Sigmoidal Function*, 1989.

D. Donoho. High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality Theories of Deep Learning View project BeamLab View project, 2014. URL https://www.researchgate.net/publication/220049061.

F. Gillet-Chaulet, G. Durand, O. Gagliardini, C. Mosbeux, J. Mouginot, F. Rémy, and C. Ritz. Assimilation of surface velocities acquired between 1996 and 2010 to constrain the form of the basal friction law under Pine Island Glacier. *Geophysical Research Letters*, 43:10,311–10,321, 10 2016. ISSN 19448007. doi: 10.1002/2016GL069937.

S. Glantz and B. Slinker. *Primer of Applied Regression and Analysis of Variance*. McGraw-Hill Education, 2000. ISBN 9780071360869. URL https://books.google.dk/books?id=fzV2QgAACAAJ.

J. W. Glen. The flow law of ice: A discussion of the assumptions made in glacier theory, their experimental foundations and consequences. *IASH Publ*, 47:e183, 1958.

M. Habermann, D. Maxwell, and M. Truffer. Reconstruction of basal properties in ice sheets using iterative inverse methods. *Journal of Glaciology*, 58:795–808, 2012. doi: 10.3189/2012JoG11J168.

M. Habermann, M. Truffer, and D. Maxwell. Error sources in basal yield stress inversions for Jakobshavn Isbræ, Greenland, derived from residual patterns of misfit to observations. *Journal of Glaciology*, 63:999–1011, 2017. doi: 10.1017/jog.2017.61.

C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. Array programming with NumPy, 9 2020. ISSN 14764687.

M. D. Hoffman and A. Gelman. The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo, 2014. URL http://mcmc-jags.sourceforge.net.

I. M. Howat, A. Negrete, and B. E. Smith. The Greenland Ice Mapping Project (GIMP) land classification and surface elevation data sets. *Cryosphere*, 8:1509–1518, 8 2014. ISSN 19940424. doi: 10.5194/tc-8-1509-2014.

E. Hüllermeier and W. Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, 110:457–506, 3 2021. ISSN 15730565. doi: 10.1007/s10994-021-05946-3.

M. Jay-Allemand, F. Gillet-Chaulet, O. Gagliardini, and M. Nodet. Investigating changes in basal conditions of Variegated Glacier prior to and during its 1982-1983 surge. *The Cryosphere*, 5:659–672, 2011. doi: 10.5194/tc-5-659-2011. URL https://www.the-cryosphere.net/5/659/2011/.

W. F. Jenkins, P. Gerstoft, M. J. Bianco, and P. D. Bromirski. Unsupervised Deep Clustering of Seismic Data: Monitoring the Ross Ice Shelf, Antarctica. *Journal of Geophysical Research: Solid Earth*, 126, 9 2021. ISSN 21699356. doi: 10.1029/2021JB021716.

I. Joughin and R. B. Alley. Stability of the West Antarctic ice sheet in a warming world. *Nature Geoscience*, 4:506–513, 2011. doi: 10.1038/NGEO1194.

I. Joughin, B. E. Smith, and I. M. Howat. A complete map of Greenland ice velocity derived from satellite data collected over 20 years, 2 2018. ISSN 00221430.

G. Jouvet, G. Cordonnier, B. Kim, M. Lüthi, A. Vieli, and A. Aschwanden. Deep learning speeds up ice flow modelling by several orders of magnitude. *Journal of Glaciology*, 2021. ISSN 00221430. doi: 10.1017/jog.2021.120.

A. A. Kaptanoglu, B. M. de Silva, U. Fasel, K. Kaheman, A. J. Goldschmidt, J. L. Callaham, C. B. Delahunt, Z. G. Nicolaou, K. Champion, J.-C. Loiseau, J. N. Kutz, and S. L. Brunton. PySINDy: A comprehensive Python package for robust sparse system identification. 11 2021. doi: 10.21105/joss.03994. URL `http://arxiv.org/abs/2111.08481http://dx.doi.org/10.21105/joss.03994`.

N. B. Karlsson, A. M. Solgaard, K. D. Mankoff, F. Gillet-Chaulet, J. A. MacGregor, J. E. Box, M. Citterio, W. T. Colgan, S. H. Larsen, K. K. Kjeldsen, N. J. Korsgaard, D. I. Benn, I. J. Hewitt, and R. S. Fausto. A first constraint on basal melt-water production of the Greenland ice sheet. *Nature Communications*, 12, 12 2021. ISSN 20411723. doi: 10.1038/s41467-021-23739-z.

D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. 12 2014. URL `http://arxiv.org/abs/1412.6980`.

Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Fourier Neural Operator for Parametric Partial Differential Equations. 10 2020. URL `http://arxiv.org/abs/2010.08895`.

D. R. MacAyeal. Large-scale ice flow over a viscous basal sediment: theory and application to ice stream B, Antarctica. *Journal of Geophysical Research*, 94: 4071–4087, 1989. ISSN 01480227. doi: 10.1029/jb094ib04p04071.

N. Maier, F. Gimbert, F. Gillet-Chaulet, and A. Gilbert. Basal traction mainly dictated by hard-bed physics over grounded regions of Greenland. *Cryosphere*, 15: 1435–1451, 3 2021. ISSN 19940424. doi: 10.5194/tc-15-1435-2021.

N. M. Mangan, J. N. Kutz, S. L. Brunton, and J. L. Proctor. Model selection for dynamical systems via sparse regression and information criteria. 1 2017. doi: 10.1098/rspa.2017.0009. URL `http://arxiv.org/abs/1701.01773http://dx.doi.org/10.1098/rspa.2017.0009`.

K. D. Mankoff, X. Fettweis, P. L. Langen, M. Stendel, K. K. Kjeldsen, N. B. Karlsson, B. Noël, M. R. V. D. Broeke, A. Solgaard, W. Colgan, J. E. Box, S. B. Simonsen, M. D. King, A. P. Ahlstrøm, S. B. Andersen, and R. S. Fausto. Greenland ice sheet mass balance from 1840 through next week. *Earth System Science Data*, 13: 5001–5025, 10 2021. ISSN 18663516. doi: 10.5194/essd-13-5001-2021.

F. S. McCormack, J. L. Roberts, L. M. Jong, D. A. Young, and L. H. Beem. A note on digital elevation model smoothing and driving stresses. *Polar Research*, 38, 2019. ISSN 17518369. doi: 10.33265/polar.v38.3498.

L. W. Morland, C. J. der Veen, and J. Oerlemans. Dynamics of the West Antarctic ice sheet. pages 99–116, 1987.

M. Morlighem, E. Rignot, H. Seroussi, E. Larour, H. B. Dhia, and D. Aubry. A mass conservation approach for mapping glacier ice thickness. *Geophysical Research Letters*, 38, 10 2011. ISSN 00948276. doi: 10.1029/2011GL048659.

M. Morlighem, C. N. Williams, E. Rignot, L. An, J. E. Arndt, J. L. Bamber, G. Catania, N. Chauché, J. A. Dowdeswell, B. Dorschel, I. Fenty, K. Hogan, I. Howat, A. Hubbard, M. Jakobsson, T. M. Jordan, K. K. Kjeldsen, R. Millan, L. Mayer, J. Mouginot, B. P. Noël, C. O'Cofaigh, S. Palmer, S. Rysgaard, H. Seroussi, M. J. Siegert, P. Slabon, F. Straneo, M. R. van den Broeke, W. Weinrebe, M. Wood, and K. B. Zinglersen. BedMachine v3: Complete Bed Topography and Ocean Bathymetry Mapping of Greenland From Multibeam Echo Sounding Combined With Mass Conservation. *Geophysical Research Letters*, 44:11,051–11,061, 11 2017. ISSN 19448007. doi: 10.1002/2017GL074954.

R. M. Neal. MCMC using Hamiltonian dynamics. 6 2012. doi: 10.1201/b10905. URL http://arxiv.org/abs/1206.1901http://dx.doi.org/10.1201/b10905.

B. Noël, W. J. V. D. Berg, S. Lhermitte, and M. R. V. D. Broeke. Rapid ablation zone expansion amplifies north Greenland mass loss, 2019. URL https://www.science.org.

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, A. Müller, J. Nothman, G. Louppe, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Édouard Duchesnay. Scikit-learn: Machine Learning in Python. 1 2012. URL http://arxiv.org/abs/1201.0490.

D. Phan, N. Pradhan, and M. Jankowiak. Composable Effects for Flexible and Accelerated Probabilistic Programming in NumPyro. 12 2019. URL http://arxiv.org/abs/1912.11554.

C. Rackauckas. The Essential Tools of Scientific Machine Learning. 2019. doi: 10.15200/winn.156631.13064.

C. Rackauckas, Y. Ma, J. Martensen, C. Warner, K. Zubov, R. Supekar, D. Skinner, A. Ramadhan, and A. Edelman. Universal Differential Equations for Scientific Machine Learning. 1 2020. URL http://arxiv.org/abs/2001.04385.

J. Schmidhuber. Deep Learning in Neural Networks: An Overview. 4 2014. doi: 10.1016/j.neunet.2014.09.003. URL http://arxiv.org/abs/1404.7828http://dx.doi.org/10.1016/j.neunet.2014.09.003.

C. Schoof. Marine ice-sheet dynamics. Part 1. The case of rapid sliding. *Journal of Fluid Mechanics*, 573:27–55, 2 2007. ISSN 14697645. doi: 10.1017/S0022112006003570.

O. V. Sergienko, R. A. Bindschadler, P. L. Vornberger, and D. R. MacAyeal. Ice stream basal conditions from block-wise surface data inversion and simple regression models of ice stream flow: Application to Bindschadler Ice Stream. *Journal of Geophysical Research: Earth Surface*, 113, 2008. doi: 10.1029/2008JF001004. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2008JF001004.

H. Sevestre and D. I. Benn. Climatic and geometric controls on the global distribution of surge-type glaciers: implications for a unifying model of surging. *Journal of Glaciology*, 61:646–662, 2015. doi: 10.3189/2015JoG14J136.

A. Solgaard and A. Kusk. Greenland Ice Velocity from Sentinel-1 Edition 2, 2022. URL https://doi.org/10.22008/promice/data/sentinel1icevelocity/greenlandicesheet.

A. Solgaard, A. Kusk, J. P. M. Boncori, J. Dall, K. D. Mankoff, A. P. Ahlstrøm, S. B. Andersen, M. Citterio, N. B. Karlsson, K. K. Kjeldsen, N. J. Korsgaard, S. H. Larsen, and R. S. Fausto. Greenland ice velocity maps from the PROMICE project. *Earth System Science Data*, 13:3491–3512, 7 2021. ISSN 18663516. doi: 10.5194/essd-13-3491-2021.

L. A. Stearns and C. J. van der Veen. Friction at the bed does not control fast glacier flow, 2018. URL https://www.science.org.

L. A. Stevens, M. Nettles, J. L. Davis, T. T. Creyts, J. Kingslake, A. P. Ahlstrom, and T. B. Larsen. Helheim Glacier diurnal velocity fluctuations driven by surface melt forcing. *Journal of Glaciology*, 68:77–89, 2 2022. ISSN 00221430. doi: 10.1017/jog.2021.74.

V. C. Tsai, A. L. Stewart, and A. F. Thompson. Marine ice-sheet profiles and stability under Coulomb basal conditions. *Journal of Glaciology*, 61:205–215, 5 2015. ISSN 00221430. doi: 10.3189/2015JoG14J221.

C. J. V. D. Veen, J. C. Plummer, and L. A. Stearns. Controls on the recent speed-up of Jakobshavn Isbræ, West Greenland. *Journal of Glaciology*, 57:770–782, 9 2011. ISSN 00221430. doi: 10.3189/002214311797409776.

S. Vijay, M. D. King, I. M. Howat, A. M. Solgaard, S. A. Khan, and B. Noël. Greenland ice-sheet wide glacier classification based on two distinct seasonal ice velocity behaviors. *Journal of Glaciology*, 67:1241–1248, 12 2021. ISSN 00221430. doi: 10.1017/jog.2021.89.

P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İlhan Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, A. Vijaykumar, A. P. Bardelli, A. Rothberg, A. Hilboll, A. Kloeckner, A. Scopatz, A. Lee, A. Rokem, C. N. Woods, C. Fulton, C. Masson, C. Häggström, C. Fitzgerald, D. A. Nicholson, D. R. Hagen, D. V. Pasechnik, E. Olivetti, E. Martin, E. Wieser, F. Silva, F. Lenders, F. Wilhelm, G. Young, G. A. Price, G. L. Ingold, G. E. Allen, G. R. Lee, H. Audren, I. Probst, J. P. Dietrich, J. Silterra, J. T. Webber, J. Slavič, J. Nothman, J. Buchner, J. Kulick, J. L. Schönberger, J. V. de Miranda Cardoso, J. Reimer, J. Harrington, J. L. C. Rodríguez, J. Nunez-Iglesias, J. Kuczynski, K. Tritz, M. Thoma, M. Newville, M. Kümmerer, M. Bolingbroke, M. Tartre, M. Pak, N. J. Smith, N. Nowaczyk, N. Shebanov, O. Pavlyk, P. A. Brodtkorb, P. Lee, R. T. McGibbon, R. Feldbauer, S. Lewis, S. Tygier, S. Sievert, S. Vigna, S. Peterson, S. More, T. Pudlik, T. Oshima, T. J. Pingel, T. P. Robitaille, T. Spura, T. R. Jones, T. Cera, T. Leslie, T. Zito, T. Krauss, U. Upadhyay, Y. O. Halchenko, and Y. Vázquez-Baeza. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, 3 2020. ISSN 15487105. doi: 10.1038/s41592-019-0686-2.

J. Weertman. On the Sliding of Glaciers. *Journal of Glaciology*, 3:33–38, 1957. doi: 10.3189/S0022143000024709.

J. Weertman. Stability of the Junction of an Ice Sheet and an Ice Shelf. *Journal of Glaciology*, 13:3–11, 1974. doi: 10.3189/S0022143000023327.

Ø. A. Winton, S. B. Simonsen, A. M. Solgaard, R. McNabb, and N. B. Karlsson. Basal stress controls ice-flow variability during a surge cycle of Hagen Bræ, Greenland. *Journal of Glaciology*, 2022. ISSN 00221430. doi: 10.1017/jog.2021.111.

E. Zhang, L. Liu, L. Huang, and K. S. Ng. An automated, generalized, deep-learning-based method for delineating the calving fronts of Greenland glaciers from multi-sensor remote sensing imagery. *Remote Sensing of Environment*, 254:112265, 2021. ISSN 18790704, 00344257. doi: 10.1016/j.rse.2020.112265.

P. Zheng, T. Askham, S. L. Brunton, J. N. Kutz, and A. Y. Aravkin. A Unified Framework for Sparse Relaxed Regularized Regression: SR3. *IEEE Access*, 7: 1404–1423, 2019. ISSN 21693536. doi: 10.1109/ACCESS.2018.2886528.

A Appendices

A.1 Synthetic testing of methods

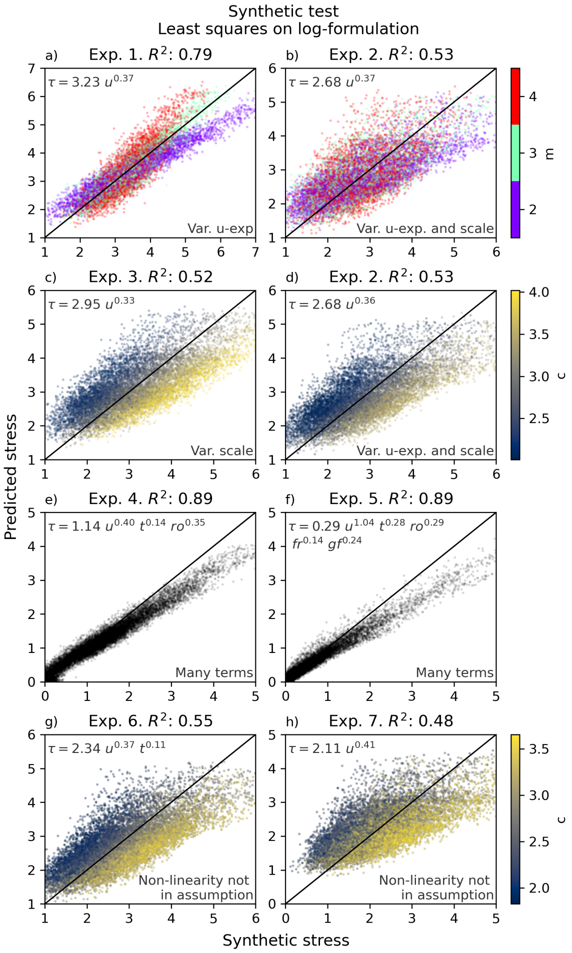

Figure A.19: Results of synthetic test on 7 experiments LSQ with 10% noise added to the target. The experiment numbers are detailed in Table 4, with a short explanation in the lower right of each panel of this figure. The resulting equations are in the upper left of each panel.

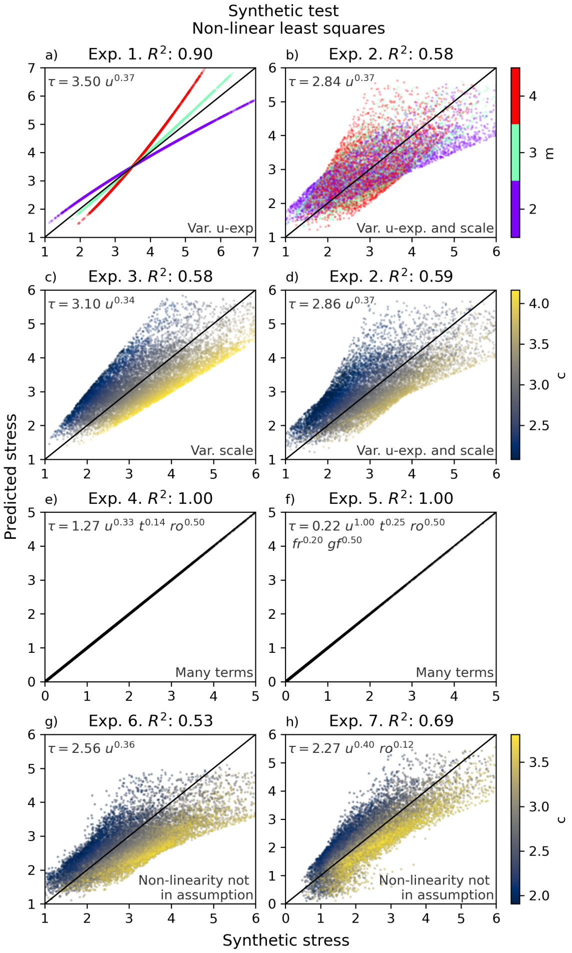

Figure A.20: Results of synthetic test on 7 experiments for NLSQ with no noise added to the target. The experiment numbers are detailed in Table 4, with a short explanation in the lower right of each panel of this figure. The resulting equations are in the upper left of each panel.

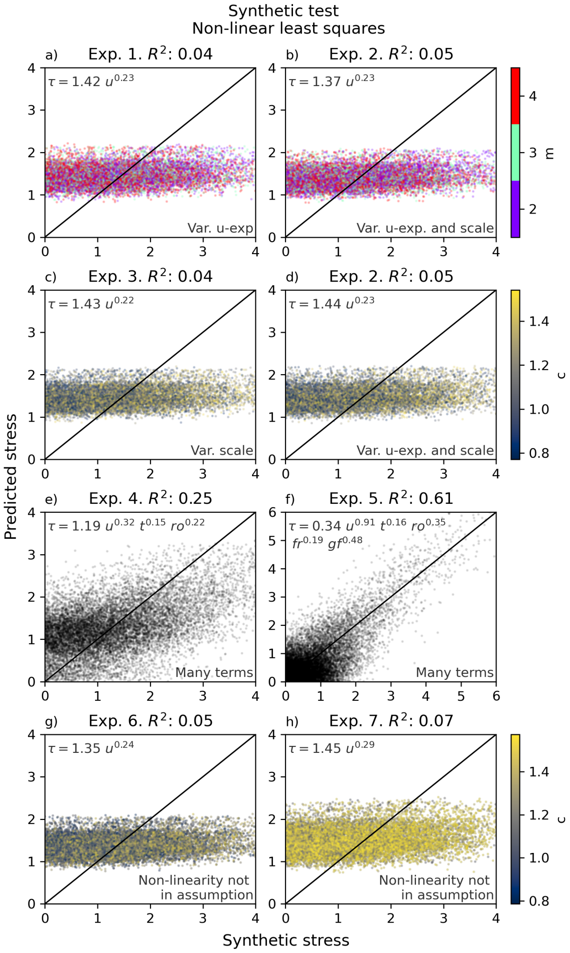

Figure A.21: Results of synthetic test on 7 experiments for NLSQ with 100% noise added to the target. The experiment numbers are detailed in Table 4, with a short explanation in the lower right of each panel of this figure. The resulting equations are in the upper left of each panel.

A.2 Results

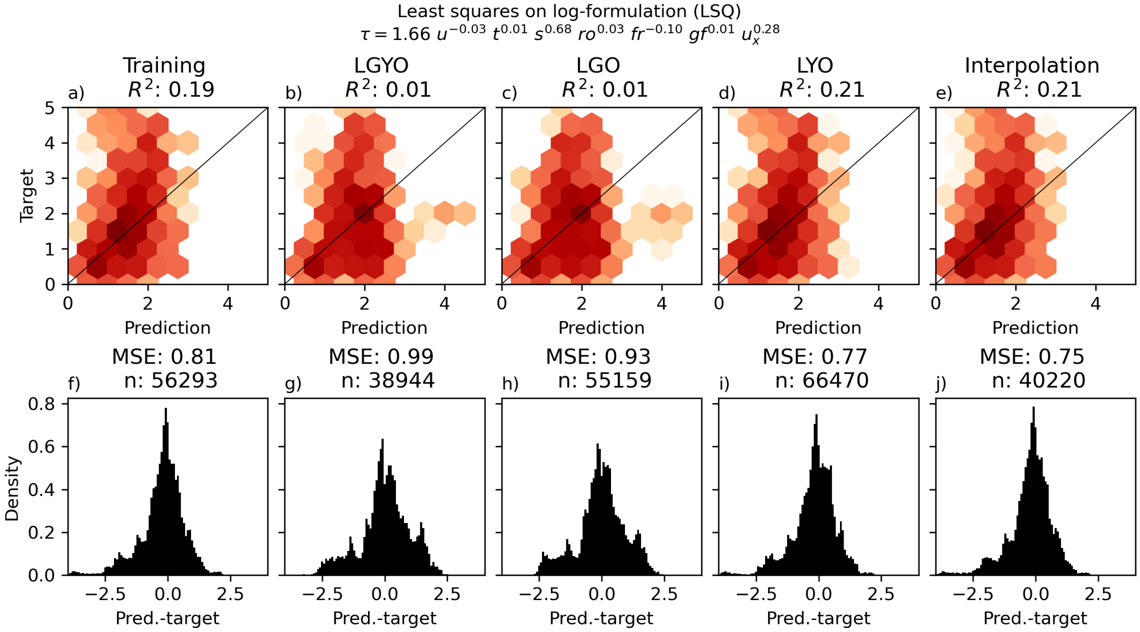

Figure A.22: Results on training and test data for LSQ. **a)-e):** density plots of target-prediction for training and test data sets. The diagonal line indicates a perfect fit. R^2 for each data set is in the subplot title. **f)-j):** error histograms for training and test data sets. MSE and the number of data points for each data set in the subplot title. LGYO (leave-glaciers-and-years-out), LGO (leave-glaciers-out) and LYO (leave-years-out) are three different aspects of extrapolation, which are divided to yield insight into the generalizability of a given model.

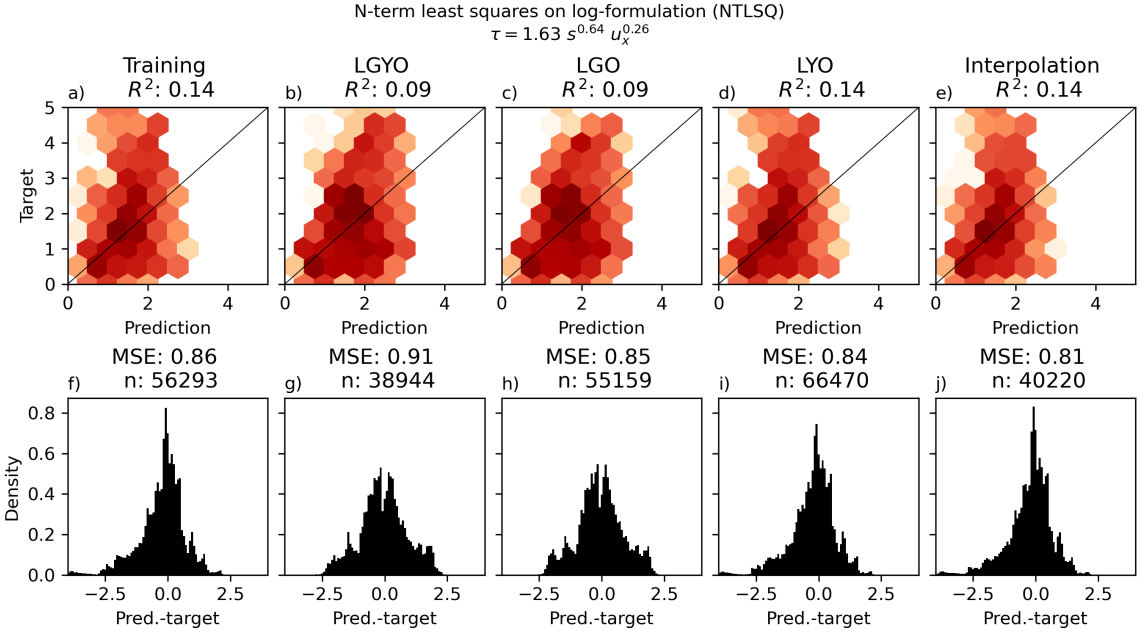

Figure A.23: Results on training and test data for NTLSQ. Explanation of figure is provided in caption of Figure A.22.

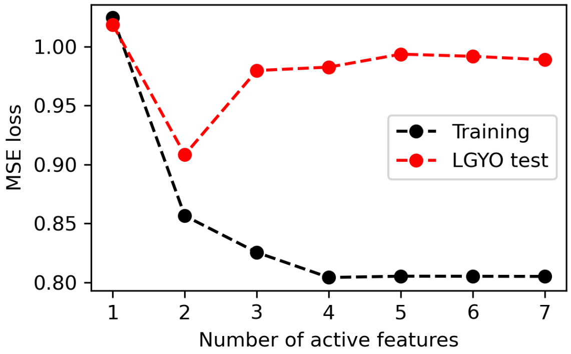

Figure A.24: Training and LGYO test loss as a function of active terms for NTLSQ, used to determine the parsimonious model that best fits the data. From this figure, the optimal number of active features is 2.

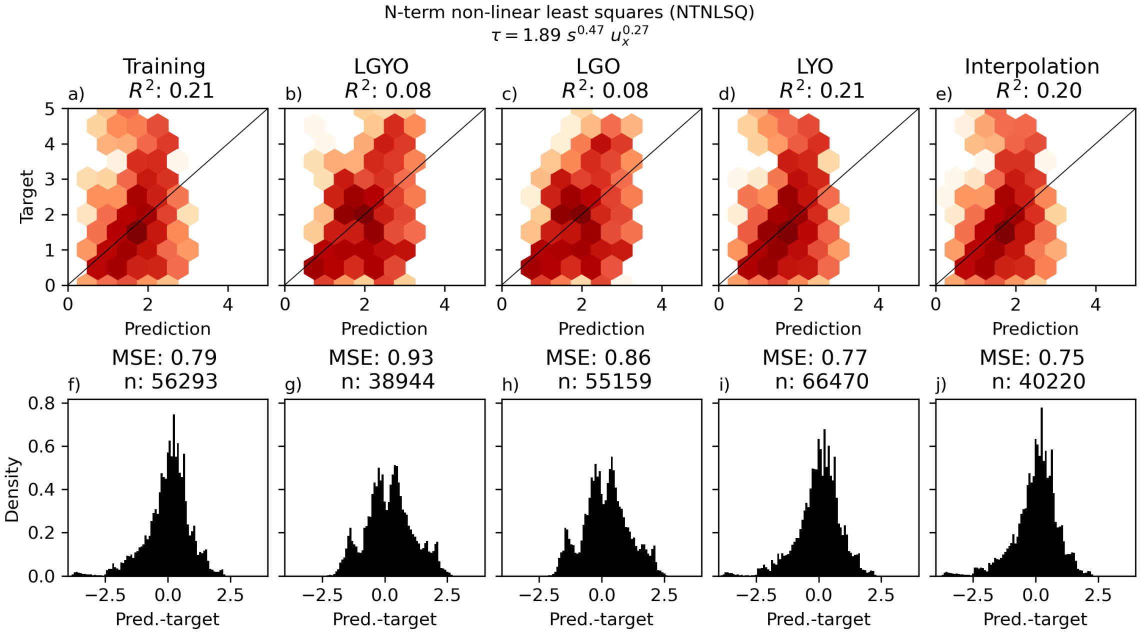

Figure A.25: Results on training and test data for NTNLSQ. Explanation of figure is provided in caption of Figure A.22.

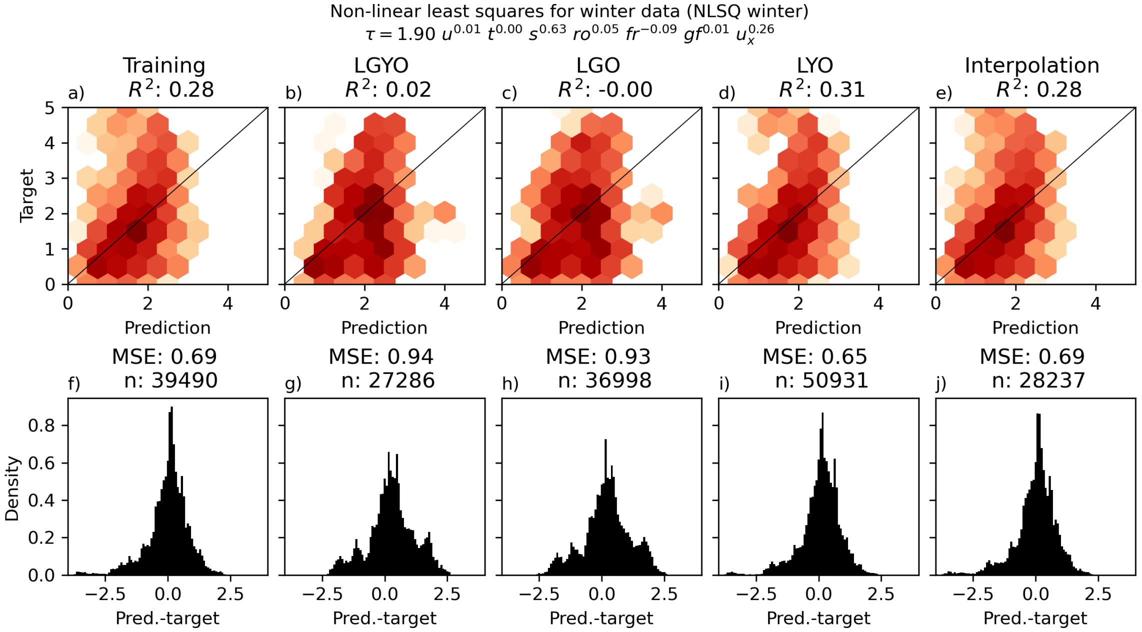

Figure A.26: Results on training and test data for NLSQ without summer data (1/5-1/10). Explanation of figure is provided in caption of Figure A.22.

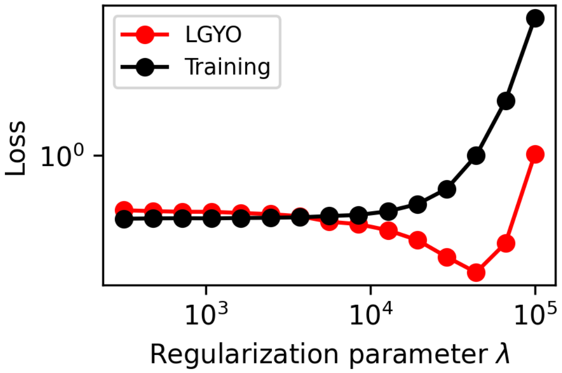

Figure A.27: Test and training loss as a function of regularization strength in the SR3 optimization algorithm used for SINDy, used to determine the regularization parameter in SR3 (Zheng et al., 2019). The x-axis is the regularization strength on a logarithmic scale, but the labels have been replaced with the number of active terms at each regularization level. The lowest LGYO value corresponds to $\lambda = 44.000$ in Eq. (19)

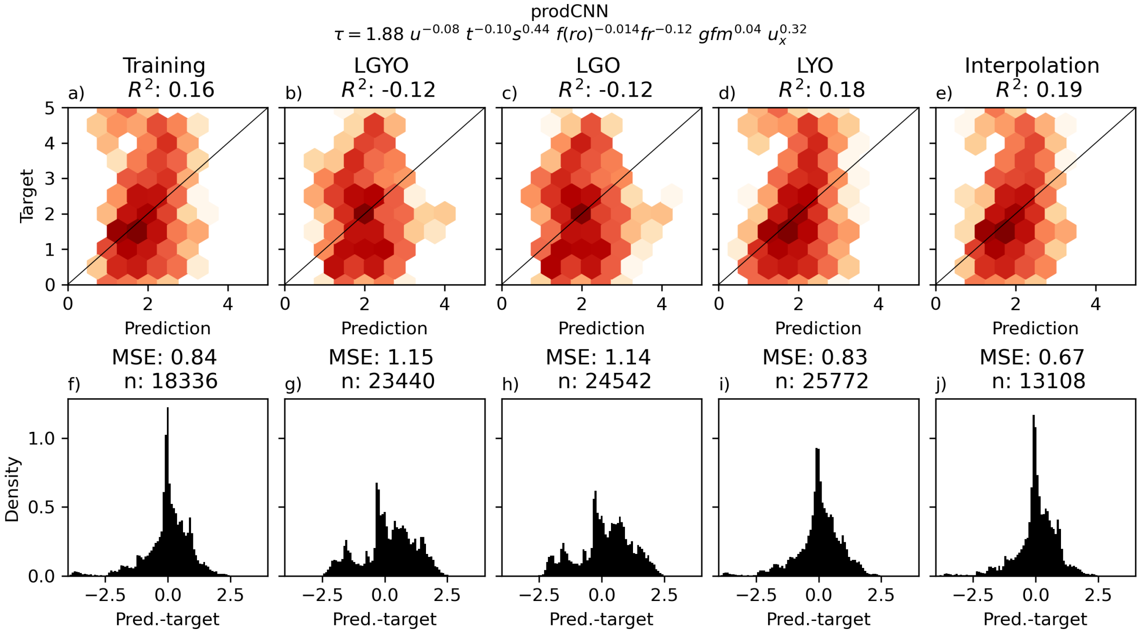

Figure A.28: Results on training and test data for prodCNN. Explanation of figure is provided in caption of Figure A.22.

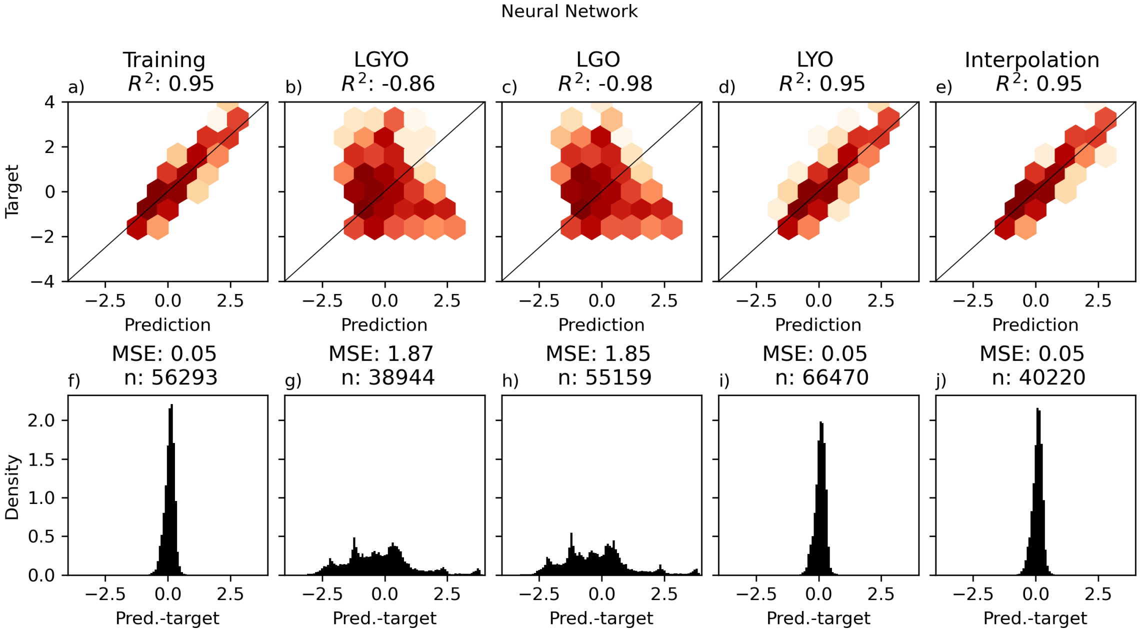

Figure A.29: Results on training and test data for FFNN with 1000 epochs. Explanation of figure is provided in caption of Figure A.22.

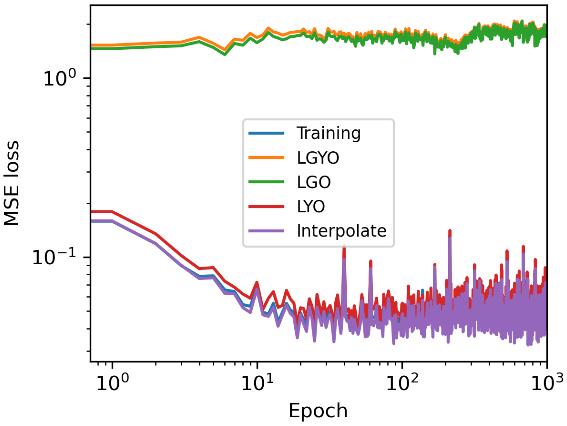

Figure A.30: Evolution of losses for FFNN.

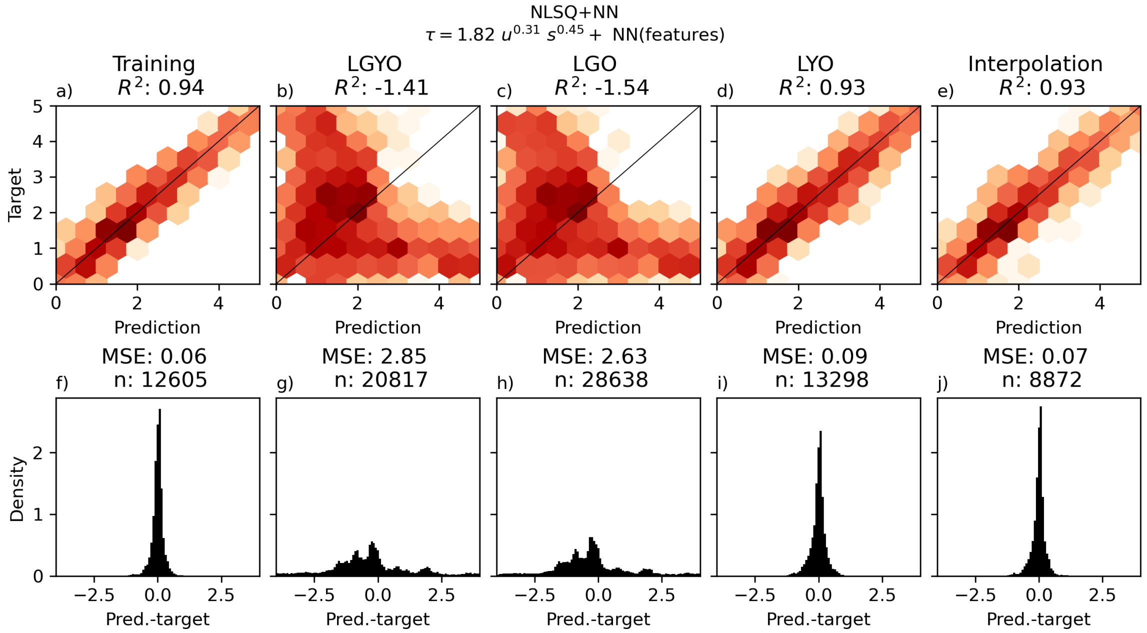

Figure A.31: Results on training and test data for NLSQ+NN. Explanation of figure is provided in caption of Figure A.22.

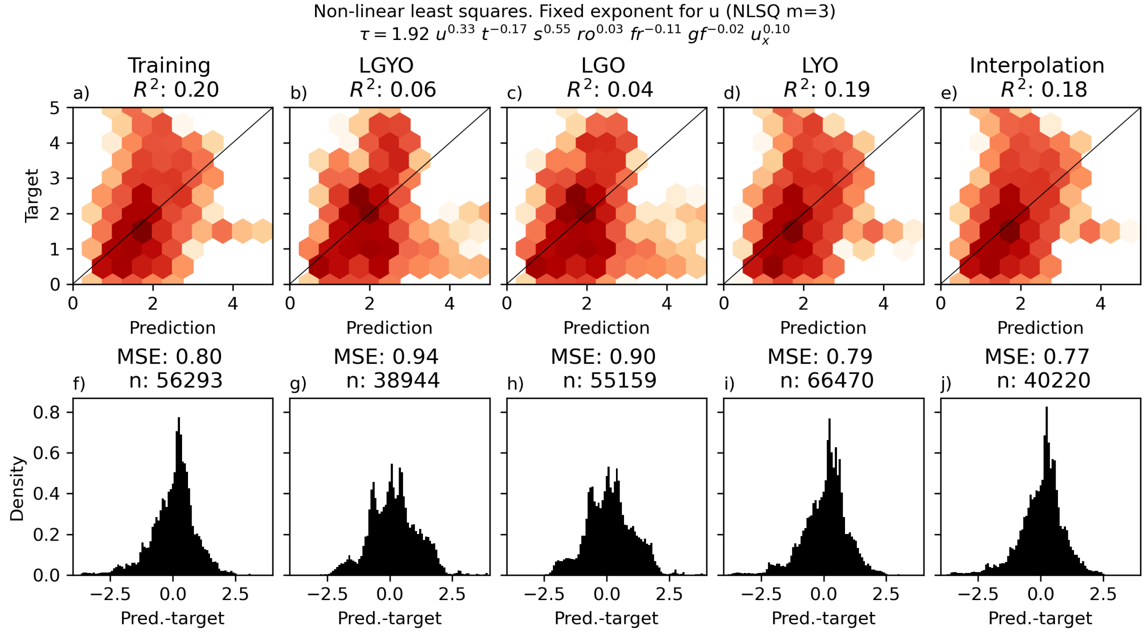

Figure A.32: Results on training and test data for NLSQ with the exponent of u fixed at 1/3. Explanation of figure is provided in caption of Figure A.22.

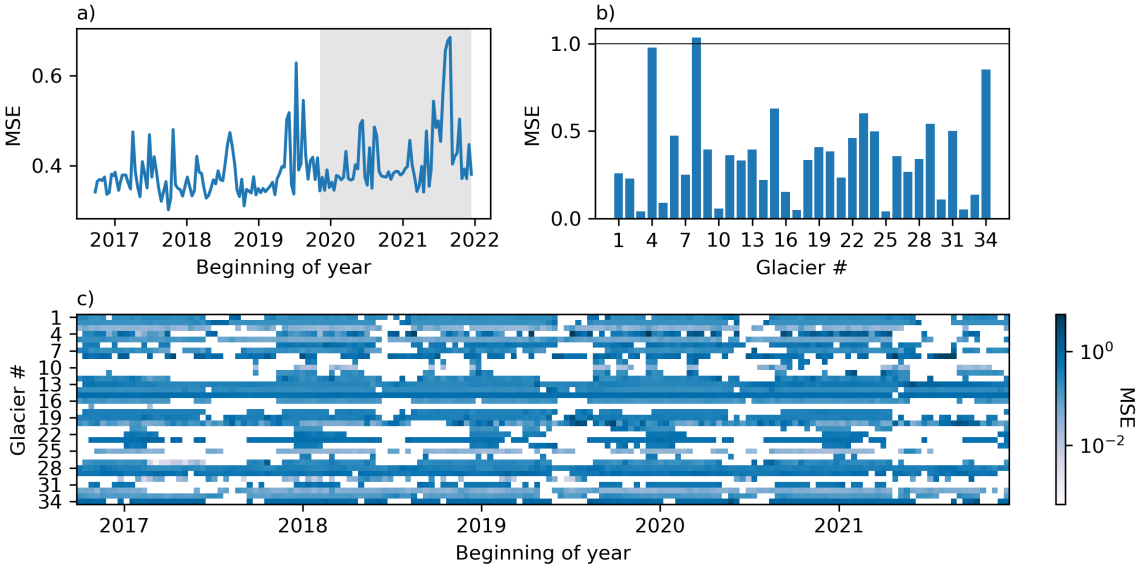

Figure A.33: Spatiotemporal error for INLSQ. Explanation of figure in Figure 14.

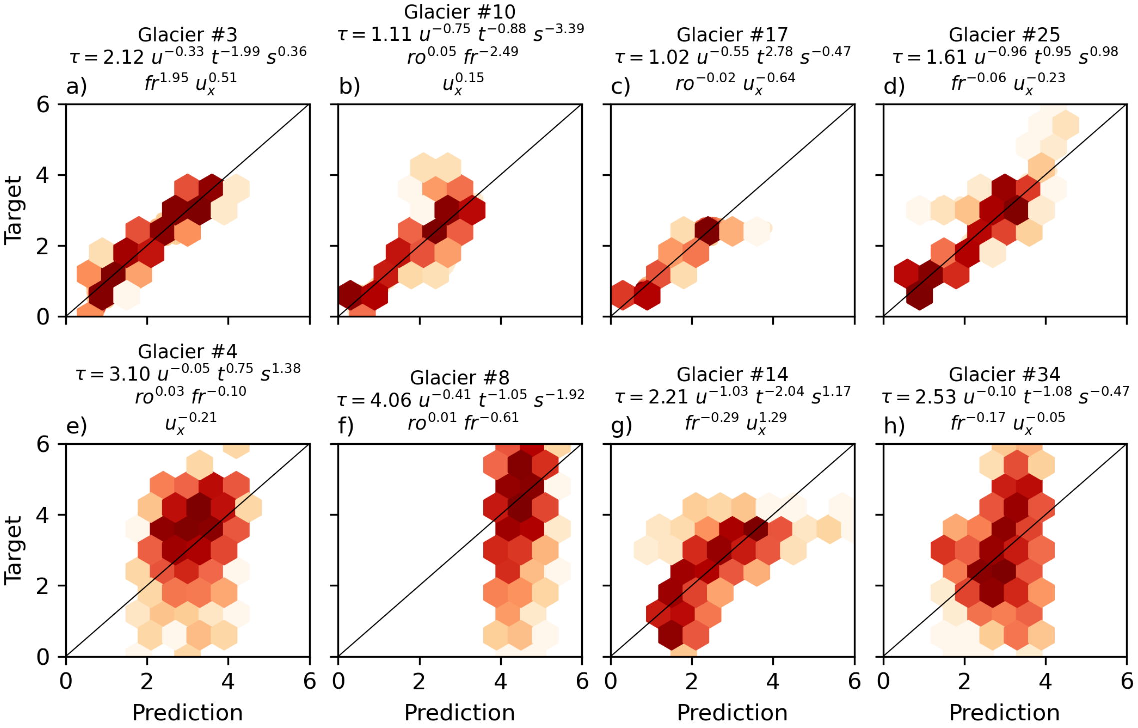

Figure A.34: Best and worst fits for INLSQ

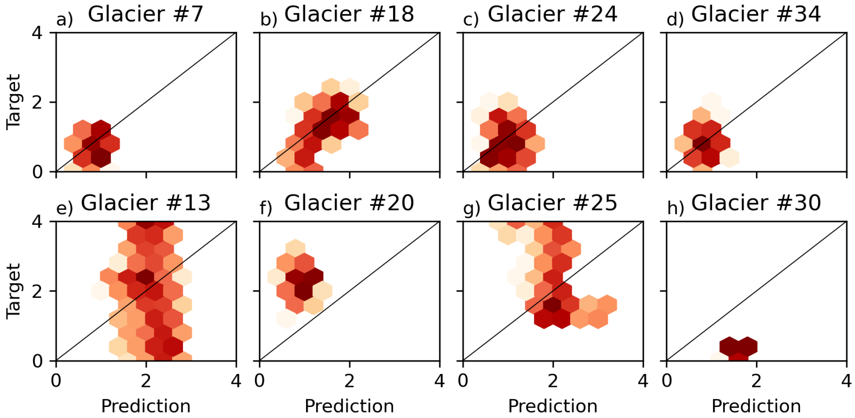

Figure A.35: Best and worst fits for NLSQ

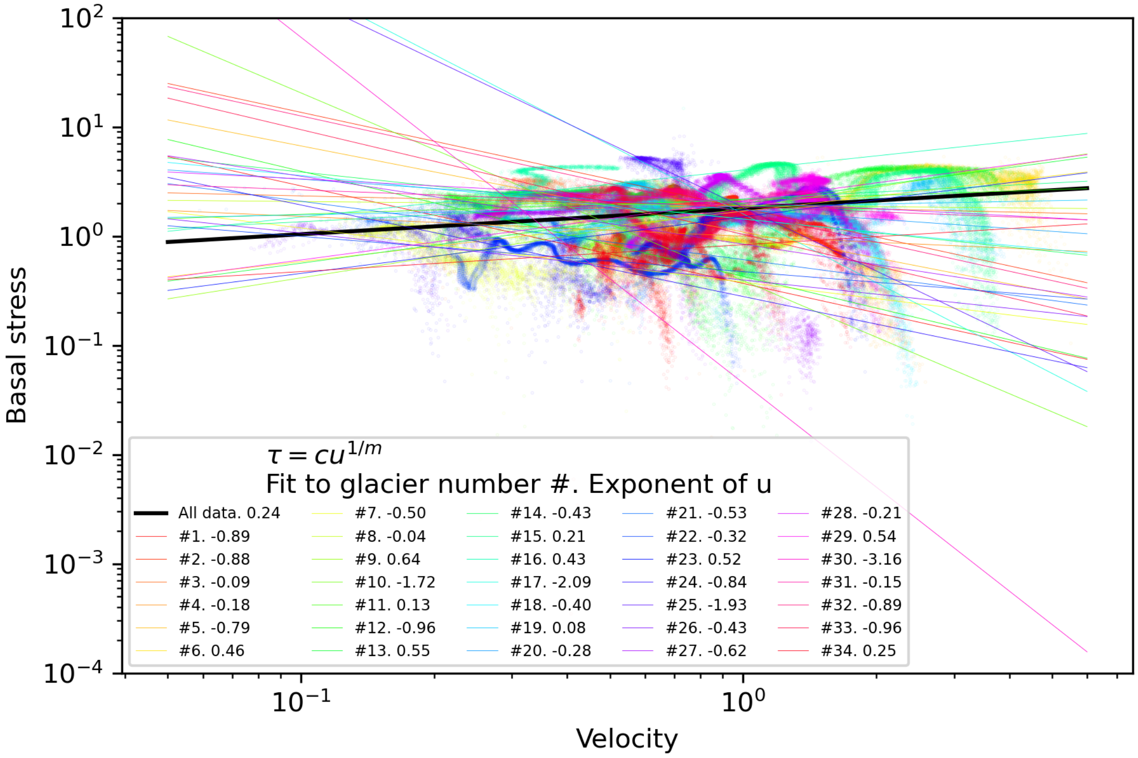

Figure A.36: Scatter plot of basal stress and velocities on log-scale. Plotted with results of INLSQ with only feature u. NLSQ with only u shown in black. Illustrates the effect of how fitting to all data yields a positive exponent, while fitting individually to glaciers generally yields negative exponents, in the power-law formulation $\tau = cu^{m}$.

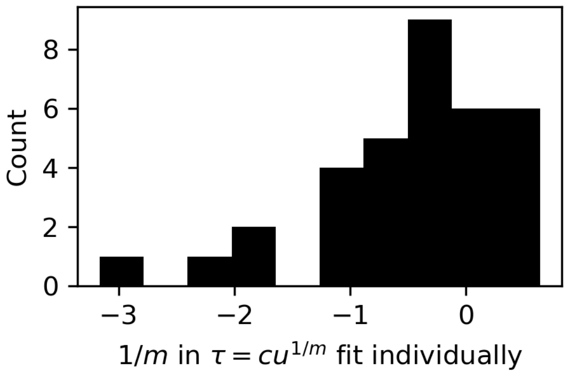

Figure A.37: Histogram of exponent for INLSQ with only feature u.

A.3 Glaciers

Glacier #	Glacier name	Number of data points
1	79 North Glacier	4682
2	DaugaardJensen	17203
3	Docker Smith	2250
4	Fenris	1128
5	Hayes	7285
6	Helheim	12741
7	Humboldt	7977
8	Ikertivaq A	2548
9	Ikertivaq B	2200
10	Ikertivaq C	2436
11	Ikertivaq D	2352
12	Ingia	4182
13	Jakobshavn	11972
14	KNS	15147
15	Kangerdlugssuaq	23027
16	Kangerdlugssup Sermerssua	5506
17	Kangigdleq	95
18	Kong Oscars	14228
19	Nunatakassaap Sermia	8250
20	Nunatakavsaup Sermia	1648
21	Ostenfeld	3752
22	Petermann	10258
23	Rink	18581
24	Ryder	5083
25	Sermeq Silardleq	4568
26	Steensby	960
27	Steenstrup	7962
28	Store	13782
29	Tingmjarmiut	9536
30	Umiamako	153
31	Upernavik A	7502
32	Upernavik B	7783
33	Upernavik C	10574
34	Zachariae	9735

Table A.6: Overview of the numbering of glaciers along with the total number of data points (in time and space) for each glacier.