

Optimizing Noisy Functions with Machine Learning

Jordan R. Hall

Department of Mathematical and Statistical Sciences
University of Colorado Denver

Friday, February 1, 2019



Department of Mathematical
& Statistical Sciences

UNIVERSITY OF COLORADO **DENVER**

Introduction

- Dimension Reduction

- Derivative-Free Optimization (DFO)

Preliminary Results

- Learning from Sampling

- Using the Active Subspace

- Software Dissemination

References



Notation

- » We define a model parameter space Λ with dimension N and a data space \mathcal{D} with dimension M .
 - In most settings, Λ will be of higher dimension than \mathcal{D} .
- » We define a parameter-to-data map $f : \Lambda \rightarrow \mathcal{D}$,
 - f may be polluted by noise.
 - ∇f may be inaccessible.
- » We write $d = f(\lambda) \in \mathcal{D}$ to denote a particular datum corresponding to the evaluation of a point $\lambda \in \Lambda$.



- » We consider functions $f : \Lambda \rightarrow \mathcal{D}$ where $\dim(\Lambda) = N$ is large and $\dim(\mathcal{D}) = M$ is such that $M < N$ or $M \ll N$.
 - Functions of interest may represent postprocessed quantities from the solution of complex physical models.
- » It is not often that every parameter has equal impact on function values – usually some parameters matter more than others.
- » The dimension reduction techniques considered seek to explain outputs $f(\Lambda)$ in an *active subspace* $\mathcal{A} \subset \Lambda$ for which $\dim(\mathcal{A}) < N$.
 - Many common uses (e.g., statistics) of f involve integration over Λ and are subject to the *curse of dimensionality*.
 - This forces the use of Monte Carlo or Quasi-Monte Carlo methods.
 - A lower-dimensional representation of f may enable faster methods.



- » $\nabla f(\lambda) \in \Lambda$ is a column vector containing the N partial derivatives of f , which for this discussion we assume exist, and are square integrable in Λ equipped with some probability density that is positive everywhere in Λ and 0 otherwise.
 - We consider $\pi_{\Lambda}^{\text{prior}}(\lambda)$, the density describing our prior state of knowledge, which we abbreviate as π_{Λ} .
- » For convenience, one transforms inputs λ to the origin with some fixed variance, typically so that $\lambda \in [-1, 1]^N$. We define

Covariance in Gradient Space ¹

$$W = \int_{\Lambda} \nabla f(\lambda) \nabla f(\lambda)^{\top} \pi_{\Lambda}(\lambda) d\lambda, \quad (1)$$

which is an $N \times N$ symmetric positive semi-definite matrix.

¹Constantine, *Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies*

- » Interpreting W as a certain covariance structure over Λ leads one to the idea of computing the Singular Value Decomposition of W ,

Singular Value Decomposition (SVD) of W

$$W = U\Sigma V^*, \quad (2)$$

where U is $N \times N$ unitary, Σ is $N \times N$ diagonal with the singular values of W along its diagonal, and V^* is $N \times N$ unitary.

- » We plot the singular values, $\{\sigma_i\}_{i=1}^n$ and seek a drop-off in magnitude between some pair of singular values, σ_j and σ_{j+1} . The active subspace is the span of u_1, \dots, u_j , which are the first j columns of U , the left singular vectors of W .



» For a point $\lambda \in \Lambda$, we define

Projection into \mathcal{A} , *active variables*

$$\mathcal{P}_{\mathcal{A}}(\lambda) = \sum_{i=1}^j (u_i^T \lambda) u_i, \quad (3)$$

which is the projection of λ in the active directions of f .

» We have arrived at the property that

Resolution of f in \mathcal{A}

$$f(\mathcal{P}_{\mathcal{A}}(\lambda)) \approx f(\lambda). \quad (4)$$

- » Finding an active subspace requires forming an approximation to W via Monte Carlo. Here we consider techniques in ¹ ².
- » We let $D_S = \{(\lambda_i, f(\lambda_i))\}_{i=1}^S$, which is a set of S pairs of samples $\lambda_i \in \Lambda$ and their function values.
- » One may use D_S to approximate ∇f . We denote each estimation to $\nabla f(\lambda_i) \approx \widehat{\nabla f}(\lambda_i)$.
- » We form the $N \times S$ matrix \tilde{W} (which we present as \tilde{W}^\top)

Monte Carlo Approximation to W ¹

$$\tilde{W}^\top := \begin{bmatrix} \widehat{\nabla f}(\lambda_1) & \cdots & \widehat{\nabla f}(\lambda_S) \end{bmatrix}. \quad (5)$$

¹Russi, *UQ with Experimental Data and Complex System Models*

²Constantine et al, *Computing Active Subspaces Efficiently with Gradient Sketching*

- » Forming the SVD of \tilde{W} , $\tilde{W} = \tilde{U}\tilde{\Sigma}\tilde{V}^*$, we search for a drop off in the magnitude of the singular values $\{\tilde{\sigma}_i\}_{i=1}^S$. Assuming such a drop off occurs for an index $j : 1 < j < S$, we have the j corresponding left singular vectors, $\tilde{u}_1, \dots, \tilde{u}_j$.
- » Then we define

Monte Carlo approximation to \mathcal{A}

$$\mathcal{A}(f; D_S) := \text{span}\{\tilde{u}_1, \dots, \tilde{u}_j\},$$

the active subspace of f with respect to the samples D_S .

- » For low dimensional \mathcal{A} , we may check $f(\mathcal{P}_{\mathcal{A}}(\lambda)) \approx f(\lambda)$ in a *sufficient summary plot*, where we plot active variables against function values.



- » Many important physical systems possess turbulent or chaotic behavior.
- » The physical state of the system $u(x, \lambda)$ and the corresponding parameter to observable map $f(u(x, \lambda))$ may be modeled as a stochastic process, or as a deterministic function with additive or multiplicative noise.
 - In this setting, the efficient extraction of accurate gradients of f in parameter space is a challenging undertaking, as popular techniques based on linearization, including adjoint methods, are inaccurate^{1 2}.
 - The finite-difference approximation of ∇f_{Λ} involve $N = \dim \Lambda$ additional, usually nonlinear model solves for the physical system state $u(x, \lambda_i + \delta \lambda_i)$, and may be greatly polluted by the noise in f .

¹Lea, *Sensitivity analysis of the climate of a chaotic system*

²Qiqi, *Least Squares Shadowing sensitivity analysis of chaotic limit cycle oscillations*



- » We are interested in derivative-free optimization (DFO) algorithms suited for additive and multiplicative noise. These techniques only require evaluations of the noisy model and random draws from a normal distribution.
 - In particular, we consider the Step-size Approximation in Randomized Search (STARS) algorithm ¹.
- » The smoothing factor and step size in STARS depend on scale factors of the L_1 Lipschitz constant of f . It is of interest to obtain estimates of L_1 , which is not straightforward in a gradient-free setting. We refer to ^{2 3} for Lipschitz constant learning.
- » We note that **the convergence of STARS is dimension dependent.**

¹Chen and Wild, *Randomized DFO of Noisy Convex Functions*

²Jan-Peter Calliess, *Lipschitz optimisation for Lipschitz interpolation*

³Kvasov and Sergeyev, *Lipschitz gradients for global optimization in a one-point-based partitioning scheme*



» We consider the problem

Optimization Under Additive Uncertainty ¹

$$\min_{\lambda \in \mathbb{R}^N} \mathbb{E} [f(\lambda) + \nu(\lambda; \epsilon)], \quad (6)$$

» where:

- (i.) $f : \mathbb{R}^N \rightarrow \mathbb{R}$ is convex;
- (ii.) ϵ is a random variable with probability density $P(\epsilon)$;
- (iii.) for all λ the additive noise model ν is independent and identically distributed, has bounded variance σ_a^2 , and is unbiased; i.e., $\mathbb{E}_\epsilon(\nu(\lambda; \epsilon)) = 0$.

¹Chen and Wild, *Randomized DFO of Noisy Convex Functions*

Stepsize h

$$h = (4L_1(N + 4))^{-1}$$

where N is the dimension and L_1 is the global Lipschitz constant for $\|\nabla f\|$.

Smoothing factor μ^*

$$\mu^* = \left[\frac{8\sigma_a^2 N}{L_1^2 (N + 6)^3} \right]^{\frac{1}{4}}$$

where σ_a^2 is the variance of the additive noise.

Algorithm 1: *Minimization of f via STARS*¹.

- 1: Define: maxit ; $\lambda^{(0)}$; $f_0 := f(\lambda^{(0)})$; μ^* ; h . Set $i=1$.
- 2: Draw a random $N \times 1$ vector $r^{(i)}$, where $r_j^{(i)} \sim N(0, 1)$ for $j = 1, \dots, N$.
- 3: Evaluate $g_i := f(\lambda^{(i-1)} + \mu^* r^{(i)})$.
- 4: Set $d^{(i)} := \frac{g_i - f_{i-1}}{\mu^*} r^{(i)}$.
- 5: Set $\lambda^{(i)} = \lambda^{(i-1)} - h \cdot d^{(i)}$.
- 6: Evaluate $f_i := f(\lambda^{(i)})$; set $i=i+1$; return to 2.
- 7: Terminate when $i=\text{maxit}$.

¹Chen and Wild, *Randomized DFO of Noisy Convex Functions*

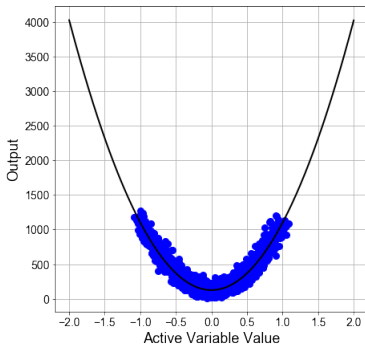
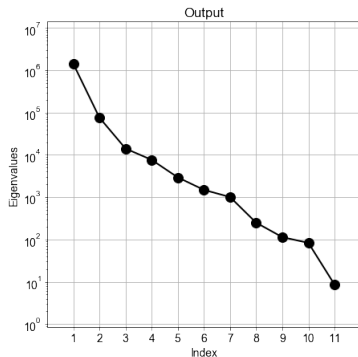
Example 3.

Let $\Lambda = [-1, 1]^{11}$ and define

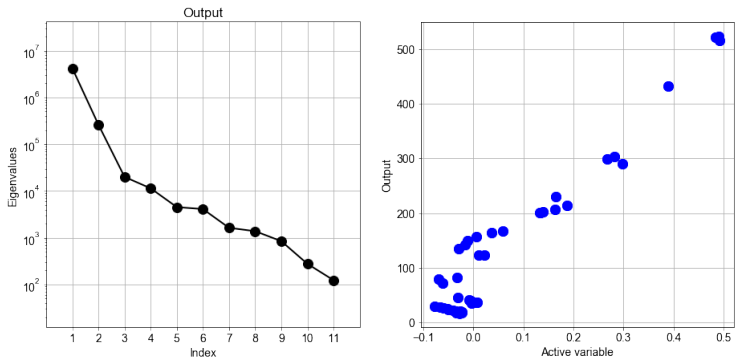
$$f(\lambda) = \sum_{i=0}^{10} 2^{(-1)^i i} \lambda_i^2 + \epsilon(\lambda),$$

where $\epsilon(\lambda)$ is a draw of additive noise corresponding to the input λ ; here, we take draws of ϵ of order 10^{-4} . We see that $\mathcal{D} = [0, 2^{10}]$ and $N = 11$, $M = 1$. Note that the minimum of f is given by $0 \in \Lambda$. Here, as i increases, terms in f become either more important or less important, depending on whether i is even or odd.

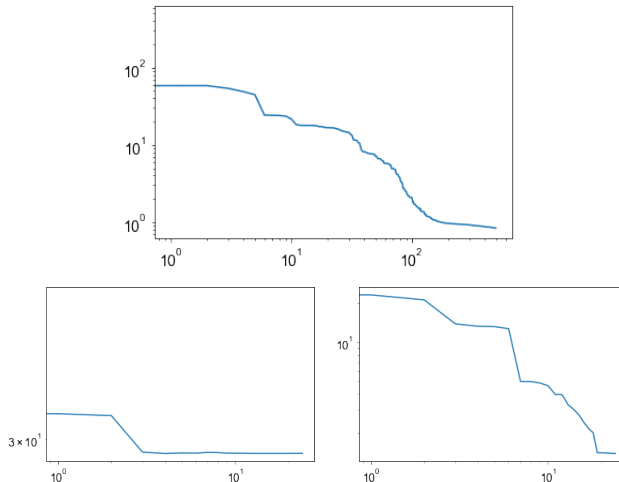




Left: A plot of the eigenvalues of the matrix \hat{W} formed from 1000 Monte Carlo samples in Λ . We see one dominant eigenvalue on the order of 10^6 . Right: A sufficient summary plot where all 1000 samples are projected into \mathcal{A} and plotted against their function values; a quadratic surrogate fits the projected data with $R^2 \approx 0.9$.



Left: A plot of the eigenvalues of the matrix \hat{W} formed from using 100 DFO iterates as samples in Λ . We again see one dominant eigenvalue between orders 10^6 and 10^7 . Right: A sufficient summary plot where all 100 samples are projected into \mathcal{A} and plotted against their function values.












Top: 500 iterations of standard DFO; Bottom Left: 25 iterations of minimizing f with DFO along the λ_8 axis; Bottom Right: 25 iterations of minimizing f with DFO along the λ_6 axis.










- » Currently, only some of the software used to produce results in this paper are publicly available on GitHub.com.
- » Some of the algorithms used here and other algorithms that are of interest are within open-source packages available online ^{1 2}.
- » Special thanks to Michael Pilosov for bringing Active Subspace package into Python 3
- » Other schemes considered here make modifications to given algorithms and remain under development.
- » A major goal of this thesis proposal will be producing well-documented, open-source software complete with python Jupyter Notebooks containing illustrative, replicable examples.

¹T. Butler et al

²Constantine et al



- 
- Battaglia, D. J. and Burrell, K. H. and Chang, C. S. and Ku, S. and deGrassie, J. S. and Grierson, B. A. "Kinetic neoclassical transport in the H-mode pedestal." *Physics of Plasmas*, Volume 21, No. 7. 2014.
- 
- T. Butler and J. Jakeman and T. Wildey. "Combining Push-Forward Measures and Bayes' Rule to Construct Consistent Solutions to Stochastic Inverse Problems." *SIAM Journal on Scientific Computing*, Volume 40, No. 2, pp. A984-A1011, 2018.
- 
- Jan-Peter Calliess. "Lipschitz optimisation for Lipschitz interpolation." In 2017 American Control Conference (ACC 2017), Seattle, WA, USA, May 2017.
- 
- Chen and Wild. "Randomized Derivative-Free Optimization of Noisy Convex Functions." Funded by the Department of Energy. 2015.
- 
- Constantine, Paul G. "Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies." *SIAM*, 2015.
- 
- Constantine, Eftekhari, Wakin. "Computing Active Subspaces Efficiently with Gradient Sketching." Conference paper, 2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP).
- 
- Kvasov and Sergeyev. "Lipschitz gradients for global optimization in a one-point-based partitioning scheme." *Journal of Computational and Applied Mathematics*. Volume 236, Issue 16, pp. 4042-4054. 2012.
- 
- S. Ku and R. Hager and C.S. Chang and J.M. Kwon and S.E. Parker. "A new hybrid-Lagrangian numerical scheme for gyrokinetic simulation of tokamak edge plasma." *Journal of Computational Physics*, Volume 315, pp. 467-475. 2016.
- 
- Lao, L.L, St. John, H, R.D. Stambaugh, A.G. Kellman, and Pfeiffer, W., "Reconstruction of current profile parameters and plasma shapes in tokamaks", *Nuclear Fusion*, Volume 25, No. 11, pp. 1611, 1985.

- 
- Lea, Daniel J. and Allen, Myles R. and Haine, Thomas W. N. "Sensitivity analysis of the climate of a chaotic system." *Tellus A*, Volume 52, No. 5, pp. 523-532. 2000.
- 
- Russi, Trent M. "Uncertainty Quantification with Experimental Data and Complex System Models." Dissertation, University of California Berkeley. 2010.
- 
- Smith, Ralph. "Uncertainty Quantification: Theory, Implementation, and Applications." SIAM, 2013.
- 
- Stuart, Andrew. "Inverse problems: A Bayesian perspective." *Acta Numerica*, volume 19, pp. 451-559. 2010.
- 
- Tarantola, Albert. "Inverse Problem Theory and Methods for Model Parameter Estimation." SIAM. 2005.
- 
- Takeda, T. and Tokuda S., "Computation of MHD equilibrium of tokamak plasma", *Journal of Computational Physics*, 93, 1, 1 - 107, 1991.
- 
- Qiqi Wang and Rui Hu and Patrick Blonigan. "Least Squares Shadowing sensitivity analysis of chaotic limit cycle oscillations." *Journal of Computational Physics*, Volume 267, pp. 210-224. 2014.
- 
- Wesson, J. "Tokamaks." "Oxford University Press, 4th edition." 2011.
- 
- Willey, T., Butler, T., Jakeman, J., Walsh, S. "A Consistent Bayesian Approach for Stochastic Inverse Problems Based on Push-forward Measures." SAND2017-3436PE. 2017.