SIAM/ASA J. UNCERTAINTY QUANTIFICATION
Vol. 9, No. 2, pp. 708–730

Efficient Estimation of the ANOVA Mean Dimension, with an Application to Neural Net Classification*

Art B. Owen† and Christopher Hoyt†

Abstract. The mean dimension of a black box function of d variables is a convenient way to summarize the extent to which it is dominated by high or low order interactions. It is expressed in terms of $2^d - 1$ variance components, but it can be written as the sum of d Sobol' indices that can be estimated by leave one out methods. We compare the variance of these leave one out methods: a Gibbs sampler called winding stairs, a radial sampler that changes each variable one at a time from a baseline, and a naive sampler that never reuses function evaluations and so costs about double the other methods. For an additive function the radial and winding stairs are most efficient. For a multiplicative function the naive method can easily be most efficient if the factors have high kurtosis. As an illustration we consider the mean dimension of a neural network classifier of digits from the MNIST data set. The classifier is a function of 784 pixels. For that problem, winding stairs is the best algorithm. We find that inputs to the final softmax layer have mean dimensions ranging from 1.35 to 2.0.

Key words. chaining, explainable AI, global sensitivity analysis, pick-freeze, Sobol' indices, winding stairs

AMS subject classifications. 62K99, 65D99, 68T07

DOI. 10.1137/20M1350236

1. Introduction.

The mean dimension of a square integrable function quantifies the extent to which higher order interactions among its d input variables are important. At one extreme, an additive function has mean dimension one which makes numerical tasks such as optimization and integration much simpler. It can also make it easier to compare the importance of the inputs to a function, and it simplifies some visualizations. At the other extreme, a function that equals a d-fold interaction has mean dimension d and can be much more difficult to study.

The mean dimension of a function can be expressed as a certain sum of Sobol' indices which we introduce below. There is extensive literature on efficiently estimating Sobol' indices (Homma and Saltelli, 1996; Jansen, 1999; Saltelli, 2002; Monod, Naud, and Makowki, 2006; Glen and Isaacs, 2012; Janon et al., 2014; Saltelli et al., 2010), and there are additional references in Puy et al. (2020) which has a thorough empirical comparison of methods for the total index, which is the one we use below. In the case of mean dimension, the necessary indices can be estimated numerically by algorithms that change just one input variable at a time. Two prominent strategies for this case are the winding stairs estimator of Jansen,

*Received by the editors July 6, 2020; accepted for publication (in revised form) February 16, 2021; published electronically June 1, 2021.
https://doi.org/10.1137/20M1350236
Funding: This work was supported by a grant from Hitachi Limited and by the U.S. National Science Foundation under grant IIS-1837931.

†Department of Statistics, Stanford University, Stanford, CA 94305 USA (owen@stanford.edu, crhoyt@stanford.edu).

Rossing, and Daamen (1994) which runs a Gibbs sampler over the input space and a radial strategy of Campolongo, Saltelli, and Cariboni (2011).

When estimating the mean dimension, a special consideration arises. Since it is a sum of d Sobol' indices, there are $O(d^2)$ covariances to consider which can greatly affect the efficiency of the estimation strategy. Sometimes a naive approach that uses roughly twice as many function evaluations can be more efficient than winding stairs because it eliminates all of those covariances.

The outline of this paper is as follows. Section 2 introduces some notation and defines the ANOVA decomposition, Sobol' indices, and the mean dimension. Section 3 presents three strategies for sampling pairs of input points that differ in just one component. A naive method takes $2Nd$ function evaluations to get N such pairs of points for each of d input variables. It never reuses any function values. A radial strategy (Campolongo, Saltelli, and Cariboni, 2011) uses $N(d+1)$ function evaluations in which N baseline points each get paired with d other points that change one of the inputs. The third strategy is winding stairs (Jansen, Rossing, and Daamen, 1994) which uses $Nd+1$ function evaluations. Section 4 compares the variances of mean dimension estimates based on these strategies. Those variances involve fourth moments of the original function. We consider additive and multiplicative functions. For additive functions all three methods have the same variance making the naive method inefficient by a factor of about 2 for large d. For some functions, methods that save function evaluations by reusing some of them can introduce positive correlations yielding a less efficient estimate. We find that the presence of factors with high kurtoses can decrease the value of reusing evaluations. Section 5 presents an example where we measure the mean dimension of a neural network classifier designed to predict a digit 0 through 9 based on 784 pixels. It was interesting to see the mean dimensions fall in the range from 1.35 to 2.0 for the penultimate layer of the network, suggesting that the information from those pixels is being used mostly one or two or three at a time. For instance, there cannot be any meaningfully large interactions of 100 or more inputs. Section 6 makes some concluding remarks. Notably, the circumstances that make the radial method inferior to the naive method or winding stairs for computing mean dimension serve to make it superior to them for some other uncertainty quantification tasks. We also discuss randomized quasi-Monte Carlo sampling alternatives and make brief comments about dependent inputs. Finally, there is an appendix in which we provide a more detailed analysis of winding stairs.

2. Notation.

We begin with the analysis of variance (ANOVA) decomposition for a function $f : \mathcal{X} \to \mathbb{R}$, where $\mathcal{X} = \prod_{j=1}^{d} \mathcal{X}_j$. We let $\boldsymbol{x} = (x_1, \dots, x_d)$, where $x_j \in \mathcal{X}_j$. The ANOVA is defined in terms of a distribution on $\mathcal{X}$ for which the x_j are independent and for which $\mathbb{E}(f(\boldsymbol{x})^2) < \infty$. The $\mathcal{X}_j$ are ordinarily subsets of $\mathbb{R}$, but the ANOVA is well defined for more general domains. We let P denote the distribution of $\boldsymbol{x}$ and let P_j denote the distribution of x_j. The ANOVA of $[0,1]^d$ was proposed by Hoeffding (1948) for U-statistics, and by Sobol' (1969) for numerical integration. It is well known in statistics following Efron and Stein (1981) where the ANOVA underlies the Efron–Stein inequality for the jackknife.

We will use $1{:}d$ as a short form for $\{1, 2, \dots, d\}$. For sets $u \subseteq 1{:}d$, their cardinality is $|u|$ and their complement $1{:}d \setminus u$ is denoted by $-u$. The components x_j for $j \in u$ are collectively denoted by $\boldsymbol{x}_u$. We will use hybrid points that merge components from two other points. The

point $\boldsymbol{y} = \boldsymbol{x}_u{:}\boldsymbol{z}_{-u}$ has $y_j = x_j$ for $j \in u$ and $y_j = z_j$ for $j \notin u$. It is typographically convenient to replace singletons $\{j\}$ by j, especially within subscripts.

The ANOVA decomposition writes $f(\boldsymbol{x}) = \sum_{u \subseteq 1:d} f_u(\boldsymbol{x})$, where the "effect" f_u depends on $\boldsymbol{x}$ only through $\boldsymbol{x}_u$. The first term is $f_\varnothing(\boldsymbol{x}) = \mathbb{E}(f(\boldsymbol{x}))$, and the others are defined recursively via

$$f_u(\boldsymbol{x}) = \mathbb{E}\Big(f(\boldsymbol{x}) - \sum_{v \subsetneq u} f_v(\boldsymbol{x}) \,\Big|\, \boldsymbol{x}_u\Big).$$

The variance component for u is

$$\sigma_u^2 \equiv \mathrm{Var}(f_u(\boldsymbol{x})) = \begin{cases} \mathbb{E}(f_u(\boldsymbol{x})^2), & u \neq \varnothing, \\ 0, & u = \varnothing. \end{cases}$$

The effects are orthogonal under P and $\sigma^2 = \mathrm{Var}(f(\boldsymbol{x})) = \sum_u \sigma_u^2$. We will assume that $\sigma^2 > 0$ in order to make some quantities well defined.

Sobol' indices quantify importance of subsets of input variables on f. They are a primary method in global sensitivity analysis (Saltelli et al., 2008; Iooss and Lemaître, 2015; Borgonovo and Plischke, 2016). Lower and upper Sobol' indices are

$$\underline{\tau}_u^2 = \sum_{v \subseteq u} \sigma_v^2 \quad \text{and} \quad \bar{\tau}_u^2 = \sum_{v \cap u \neq \varnothing} \sigma_v^2,$$

respectively. The lower index is from Sobol' (1990, 1993), while the upper index was first used by Homma and Saltelli (1996). Both indices are commonly normalized, with $\underline{\tau}_u^2/\sigma^2$ known as the closed index and $\bar{\tau}_u^2/\sigma^2$ called the total index. Normalized indices are between 0 and 1 giving them interpretations as a proportion of variance explained, similar to R^2 from regression models. The Sobol' indices $\underline{\tau}_j^2$ and $\bar{\tau}_j^2$ for singletons $\{j\}$ are of special interest.

Sobol' indices satisfy the identities

$$\begin{aligned} \underline{\tau}_u^2 &= \mathbb{E}\big(f(\boldsymbol{x}) f(\boldsymbol{x}_u{:}\boldsymbol{z}_{-u})\big) - \mu^2 \\ &= \mathbb{E}\big(f(\boldsymbol{x})(f(\boldsymbol{x}_u{:}\boldsymbol{z}_{-u}) - f(\boldsymbol{z}))\big) \quad \text{and} \\ \bar{\tau}_u^2 &= \frac{1}{2}\mathbb{E}\big((f(\boldsymbol{x}) - f(\boldsymbol{x}_{-u}{:}\boldsymbol{z}_u))^2\big), \end{aligned}$$

when $\boldsymbol{z}$ is an independent copy of $\boldsymbol{x}$. Those identities make it possible to estimate $\underline{\tau}_u^2$ and $\bar{\tau}_u^2$ by Monte Carlo or quasi-Monte Carlo sampling without explicitly computing estimates of any of the effects f_v. The first identity is due to Sobol' (1993). The second was proposed independently by Saltelli (2002) and Mauntz (2002). The third identity underlies an estimator of the total index from Jansen (1999). The numerator in the estimate of $\bar{\tau}_j^2$ from Homma and Saltelli (1996) is based on the identity of Sobol' along with $\bar{\tau}_j^2 = \sigma^2 - \underline{\tau}_{-j}^2$.

The mean dimension of f is

$$\nu(f) = \sum_{u \subseteq 1:d} \frac{|u|\sigma_u^2}{\sigma^2}.$$

It satisfies $1 \leqslant \nu(f) \leqslant d$. A low mean dimension indicates that f is dominated by low order ANOVA terms, a favorable property for some numerical problems.

An easy identity from Liu and Owen (2006) shows that $\sum_{u\subseteq 1:d}|u|\sigma_u^2 = \sum_{j=1}^d \bar\tau_j^2$. Then the mean dimension of f is

$$\nu(f) \equiv \frac{1}{\sigma^2}\sum_{j=1}^d \bar\tau_j^2 \quad \text{for} \quad \bar\tau_j^2 = \frac{1}{2}\mathbb{E}\big(f(\boldsymbol{x}) - f(\boldsymbol{x}_{-j}{:}\boldsymbol{z}_j)\big)^2.$$

Although the mean dimension combines 2^d-1 nonzero variances, it can be computed from d Sobol' indices (and the total variance σ^2).

We can get a Monte Carlo estimate of the numerator of $\nu(f)$ by summing estimates of $\bar\tau_j^2$ such as

$$\frac{1}{2N}\sum_{i=1}^N \big(f(\boldsymbol{x}_i) - f(\boldsymbol{x}_{i,-j}{:}\boldsymbol{z}_{i,j})\big)^2 \tag{1}$$

for independent random points $\boldsymbol{x}_i, \boldsymbol{z}_i \sim P$. Equation (1) corresponds to the strategy that Jansen (1999) uses to estimate the numerator of the normalized total sensitivity index for x_j. The Jansen estimator was one of the best performers in Puy et al. (2020).

There is more than one way to arrange the computation that sums (1) over $j = 1,\dots,d$. When computing a list of Sobol' indices it is advantageous to reuse many of the function values. See Saltelli (2002) for some strategies. When there are $O(d)$ total indices to sum, there are $O(d^2)$ covariances to consider, which is the issue we focus on here.

3. Estimation strategies. Equation (1) gives an estimate of $\bar\tau_j^2$ evaluating f at pairs of points that differ only in their jth coordinate. An estimate for the numerator of $\nu(f)$ sums these estimates. We have found, empirically and somewhat surprisingly, that different sample methods for computing the numerator $\sum_j \bar\tau_j^2$ can have markedly different variances, even when they are all of Jansen (1999) type.

A naive implementation uses $2Nd$ function evaluations taking $\boldsymbol{x}_i, \boldsymbol{z}_i$ independent for $i = 1,\dots,N$ for each of $j = 1,\dots,d$. In that strategy, the point $\boldsymbol{x}_i$ in (1) is actually different for each j. Such a naive implementation is wasteful. We could instead use the same $\boldsymbol{x}_i$ and $\boldsymbol{z}_i$ for all $j = 1,\dots,d$ in the radial method of Campolongo, Saltelli, and Cariboni (2011). This takes $N(d+1)$ evaluations of f. A third strategy is known as "winding stairs" (Jansen, Rossing, and Daamen, 1994). The data come from a Gibbs sampler that in its most basic form changes inputs to f one at a time changing indices in this order: $j = 1,\dots,d,1,\dots,d,\cdots,1,\dots,d$. It uses only $Nd+1$ evaluations of f. These three approaches are illustrated in Figure 1. We will also consider a variant of winding stairs that randomly refreshes after every block of $d+1$ evaluations.

First, we compare the naive to the radial strategy. For $\nu = \sum_j \bar\tau_j^2/\sigma^2$ we concentrate on estimation strategies for the numerator

$$\delta = \sigma^2\nu = \sum_{j=1}^d \bar\tau_j^2.$$

This quantity is much more challenging to estimate than the denominator σ^2, especially for large d, as it involves d^2 covariances.

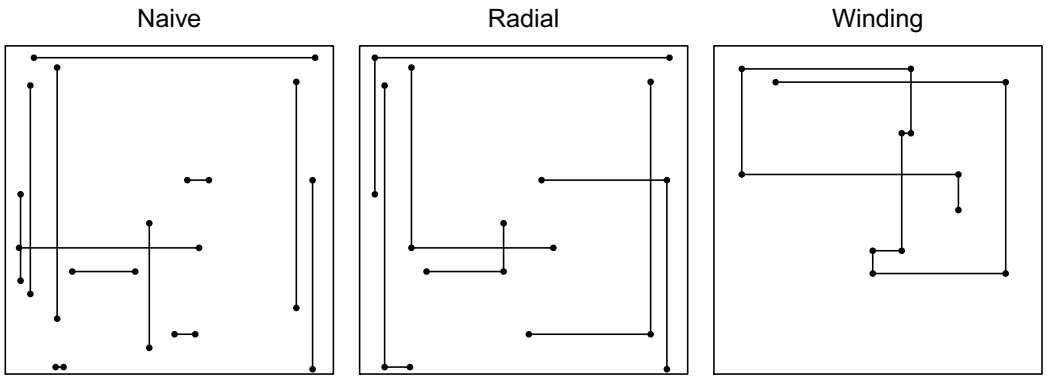

Figure 1. *Examples of three input sets to compute $\delta = \sum_{j=1}^d \bar\tau_j^2$ when $d = 2$. The naive estimate uses dN pairs of points of which there are N pairs for each of the d variables. Each edge connects a pair of points used in the estimate. The radial estimate uses N baseline points and d comparison points for each of them. The winding stairs estimates sequentially change one input at a time.*

The naive sampler takes

$$\hat\delta = \sum_{j=1}^d \widehat{\bar\tau}_j^2, \quad \text{where} \quad \widehat{\bar\tau}_j^2 = \frac{1}{2N}\sum_{i=1}^N \bigl(f(\boldsymbol{x}_i^{(j)}) - f(\boldsymbol{x}_{i,-j}^{(j)}{:}\boldsymbol{z}_{i,j})\bigr)^2 \tag{2}$$

with independent $\boldsymbol{z}_i, \boldsymbol{x}_i^{(j)} \sim P$ for $i = 1,\dots,N$ and $j = 1,\dots,d$. It takes $N(d+1)$ input vectors and $2Nd$ evaluations of f.

The radial sampler takes

$$\tilde\delta = \sum_{j=1}^d \widetilde{\bar\tau}_j^2, \quad \text{where} \quad \widetilde{\bar\tau}_j^2 = \frac{1}{2N}\sum_{i=1}^N \bigl(f(\boldsymbol{x}_i) - f(\boldsymbol{x}_{i,-j}{:}\boldsymbol{z}_{i,j})\bigr)^2 \tag{3}$$

for independent $\boldsymbol{x}_i, \boldsymbol{z}_i \sim P$, $i = 1,\dots,N$.

For $f \in L^2(P)$ both $\tilde\delta$ and $\hat\delta$ converge to $\delta = \nu\sigma^2$ as $N \to \infty$ by the law of large numbers. To compare accuracy of these estimates we assume also that $f \in L^4(P)$. Then $\mathbb{E}(f(\boldsymbol{x})^4) < \infty$ and both estimates have variances that are $O(1/N)$.

A first comparison is that

$$\begin{aligned}
\operatorname{Var}(\tilde\delta) &= \sum_{j=1}^d \operatorname{Var}(\widetilde{\bar\tau}_j^2) + 2\sum_{1\leqslant j<k\leqslant d} \operatorname{Cov}(\widetilde{\bar\tau}_j^2, \widetilde{\bar\tau}_k^2), \quad \text{while} \\
\operatorname{Var}(\hat\delta) &= \sum_{j=1}^d \operatorname{Var}(\widehat{\bar\tau}_j^2) + 2\sum_{1\leqslant j<k\leqslant d} \operatorname{Cov}(\widehat{\bar\tau}_j^2, \widehat{\bar\tau}_k^2) \\
&= \sum_{j=1}^d \operatorname{Var}(\widehat{\bar\tau}_j^2)
\end{aligned} \tag{4}$$

by independence of $(\boldsymbol{x}_i^{(j)}, \boldsymbol{z}_{i,j})$ from $(\boldsymbol{x}_i^{(k)}, \boldsymbol{z}_{i,k})$ for $j \neq k$. What we see from (4) is that while the naive estimate uses about twice as many function evaluations, the radial estimate sums d times as many terms. The off diagonal covariances do not have to be very large for us to have $\mathrm{Var}(\tilde{\delta}) > 2\mathrm{Var}(\hat{\delta})$, in which case $\hat{\delta}$ becomes the more efficient estimate despite using more function evaluations. Intuitively, each time $f(\boldsymbol{x}_i)$ takes an unusually large or small value it could make a large contribution to all d of $\widetilde{\tau}_j^2$, which can result in $O(d^2)$ positive covariances. We study this effect more precisely below giving additional assumptions under which $\mathrm{Cov}(\widetilde{\tau}_j^2, \widetilde{\tau}_k^2) > 0$. We also have a numerical counterexample at the end of this section, and so this positive covariance does not hold for all $f \in L^4(P)$.

The winding stairs algorithm starts at $\boldsymbol{x}_0 \sim P$ and then makes a sequence of single variable changes to generate $\boldsymbol{x}_i$ for $i > 0$. We let $\ell(i) \in 1{:}d$ be the index of the component that is changed at step i. The new values are independent samples $z_i \sim P_{\ell(i)}$. That is, for $i > 0$

$$\boldsymbol{x}_{i,j} = \begin{cases} z_i, & j = \ell(i), \\ \boldsymbol{x}_{i-1,j}, & j \neq \ell(i). \end{cases}$$

We have a special interest in the case where $P = \mathcal{N}(0, I)$ for which each P_j is $\mathcal{N}(0, 1)$.

The indices $\ell(i)$ can be either deterministic or random. We let $\mathcal{L}$ be the entire collection of $\ell(i)$. We assume that the entire collection of z_i are independent of $\mathcal{L}$. The most simple deterministic update has $\ell(i) = 1 + (i - 1 \mod d)$ and cycles through all indices $j \in 1{:}d$ in order. The simplest random update has $\ell(i) \overset{\text{iid}}{\sim} \mathbf{U}(1{:}d)$. In usual Gibbs sampling it would be better to take $\ell(i) \overset{\text{iid}}{\sim} \mathbf{U}(1{:}d \setminus \{\ell(i-1)\})$ for $i \geqslant 2$. Here, because we are accumulating squared differences, it is not very harmful to have $\ell(i) = \ell(i-1)$. The vector $\boldsymbol{x}_i$ contains d independently sampled Gaussian random variables. Which ones those are, depends on $\mathcal{L}$. Because $\boldsymbol{x} \sim \mathcal{N}(0, I)$ conditionally on $\mathcal{L}$, it also has that distribution unconditionally.

Letting e_j be the jth unit vector in $\mathbb{R}^d$ we can write

$$\boldsymbol{x}_i = \boldsymbol{x}_{i-1} + (z_i - x_{i-1,\ell(i)})e_{\ell(i)}.$$

If $\ell(i) \sim \mathbf{U}(1{:}d)$, then the distribution of $\boldsymbol{x}_i$ given $\boldsymbol{x}_{i-1}$ is a mixture of d different Gaussian distributions, one for each value of $\ell(i)$. As a result $\boldsymbol{y}_i = (\boldsymbol{x}_i^{\mathsf{T}}, \boldsymbol{x}_{i-1}^{\mathsf{T}})^{\mathsf{T}}$ does not then have a multivariate Gaussian distribution and is harder to study. For this reason, we focus on the deterministic update.

In the deterministic update we find that any finite set of $\boldsymbol{x}_i$ or $\boldsymbol{y}_i$ has a multivariate Gaussian distribution. We also know that $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+k}$ are independent for $k \geqslant d$ because after k steps all components of $\boldsymbol{x}_i$ have been replaced by new z_i values. It remains to consider the correlations among a block of $d + 1$ consecutive vectors. Those depend on the pattern of

shared components within different observations as illustrated in the following diagram:

$$\begin{array}{ccccccc} \boldsymbol{x}_d & \boldsymbol{x}_{d+1} & \boldsymbol{x}_{d+2} & \cdots & \boldsymbol{x}_{2d-1} & & \boldsymbol{x}_{2d} \\ \| & \| & \| & & \| & & \| \\ \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_{d-1} \\ z_d \end{pmatrix} & \begin{pmatrix} z_{d+1} \\ z_2 \\ z_3 \\ \vdots \\ z_{d-1} \\ z_d \end{pmatrix} & \begin{pmatrix} z_{d+1} \\ z_{d+2} \\ z_3 \\ \vdots \\ z_{d-1} \\ z_d \end{pmatrix} & \cdots & \begin{pmatrix} z_{d+1} \\ z_{d+2} \\ z_{d+3} \\ \vdots \\ z_{2d-1} \\ z_d \end{pmatrix} & & \begin{pmatrix} z_{d+1} \\ z_{d+2} \\ \vdots \\ z_{2d-1} \\ z_{2d} \end{pmatrix}. \end{array} \tag{5}$$

For $i \geqslant d$ and $j = 1, \dots, d$ we can write

$$\boldsymbol{x}_{i,j} = z_{r(i,j)}, \quad \text{where} \quad r(i,j) = d\left\lfloor \frac{i-j}{d} \right\rfloor + j. \tag{6}$$

It is convenient to use (6) for all $i \geqslant 0$ which is equivalent to initializing the sampler at $\boldsymbol{x}_0 = (z_{-(d-1)}, z_{-(d-2)}, \dots, z_{-1}, z_0)^{\mathsf{T}}$. Equation (6) holds for any independent $z_i \sim P_{\ell(i)}$ and does not depend on our choice of $P_j = \mathcal{N}(0,1)$.

The winding stairs estimate of δ is

$$\check{\delta} = \sum_{j=1}^{d} \check{\tau}_j^2 \quad \text{for} \quad \check{\tau}_j^2 = \frac{1}{2N} \sum_{i=1}^{N} \Delta_{d(i-1)+j}^2, \tag{7}$$

where $\Delta_r = f(\boldsymbol{x}_r) - f(\boldsymbol{x}_{r-1})$. We will see that the covariances of $\check{\tau}_j^2$ and $\check{\tau}_k^2$ depend on the pattern of common components among the $\boldsymbol{x}_i$. In our special case functions certain kurtoses have an impact on the variance of winding stairs estimates.

A useful variant of winding stairs simply makes N independent replicates of the $d+1$ vectors shown in (5), which raises the number of function evaluations from $Nd+1$ to $N(d+1)$. It uses N independent Markov chains of length $d+1$. For large d the increased computation is negligible. In original winding stairs, each squared difference $\Delta_i^2 = (f(\boldsymbol{x}_i) - f(\boldsymbol{x}_{i-1}))^2$ can be correlated with up to $2(d-1)$ other squared differences. In truncated winding stairs, it can only be correlated with $d-1$ other squared differences. We denote the resulting estimate by $\ddot{\delta}$ which is a sum of $\ddot{\tau}_j^2$.

For $d = 2$ this truncated winding stairs method is the same as radial sampling. For $d \geqslant 3$ they are different. For instance the value of f at the radial point is compared to f at d other points in the radial method while no function value is compared to more than two others in the variant of winding stairs. See Figure 2 for an illustration when $d = 3$.

In section 4.2 we present some multiplicative functions where the naive estimator of δ has much less than half of the variance of the radial estimator. To complete this section we exhibit a numerical example where the naive estimator has increased variance which must mean that the correlations induced by the radial and winding estimators are at least slightly negative. The integrand is simply $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_2$ for $\boldsymbol{x} \sim \mathcal{N}(0, I)$ in d dimensions. Figure 3 shows results. We used $N = 10^6$ evaluations to show that (truncated) winding stairs and radial sampling both have smaller variance than the naive algorithm for estimating δ. We

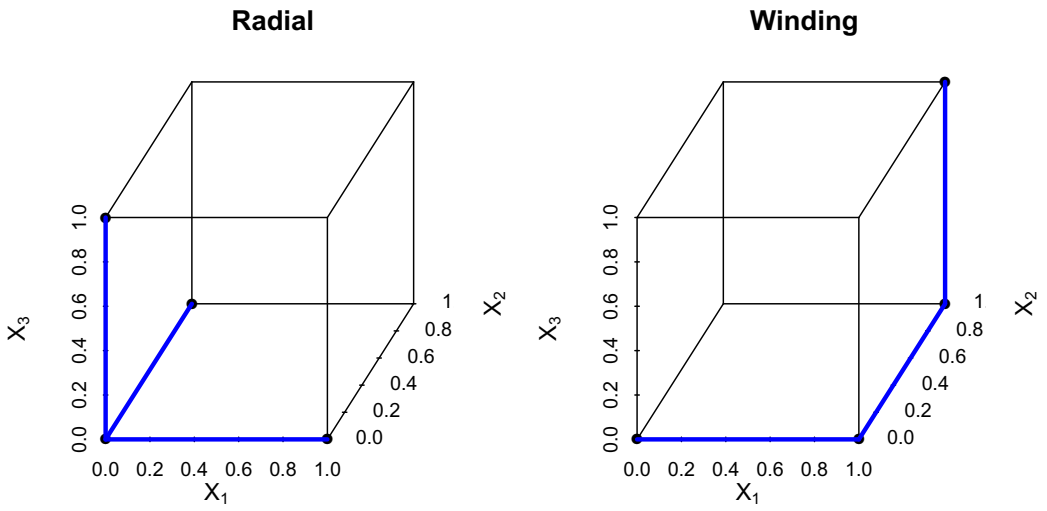

Figure 2. *The left figure shows example input points used by the radial method for $d = 3$, with thick edges connecting input points used to form differences in f. The right figure shows the same for the truncated variant of winding stairs.*

also see extremely small mean dimensions for $f(\boldsymbol{x})$ that decrease as d increases. It relates to some work in progress studying mean dimension of radial basis functions as a counterpart to Hoyt and Owen (2020) on mean dimension of ridge functions. The visible noise in that figure stems from the mean dimensions all being so very close to 1 that the vertical range is quite small. The estimate for $d = 1$ is roughly 0.9983 where the true value must be 1.

4. Additive and multiplicative functions. The variances of quadratic functions of the $f(\boldsymbol{x}_i)$ values such as $\hat{\delta}$, $\tilde{\delta}$, and $\check{\delta}$ involve fourth moments of the original function. Whereas 2^d variance components are sufficient to define Sobol' indices and numerous generalizations, fourth moments do not simplify nearly as much from orthogonality and involve considerably more quantities. While distinct pairs of ANOVA effects are orthogonal, we find for nonempty $u, v, w \subset 1{:}d$ that

$$\mathbb{E}\big(f_u(\boldsymbol{x})f_v(\boldsymbol{x})f_w(\boldsymbol{x})\big)$$

does not in general vanish when $u \subset v \cup w$, $v \subset u \cup w$, and $w \subset u \cup v$ all hold. This "chaining phenomenon" is worse for products of four effects: the number of nonvanishing combinations rises even more quickly with d. The chaining problem also comes up if we expand f in an orthonormal basis for $L^2(P)$ and then look at fourth moments.

In this section we investigate some special functional forms. The first is an additive model

$$f_A(\boldsymbol{x}) = \mu + \sum_{j=1}^{d} g_j(x_j), \tag{8}$$

where $\mathbb{E}(g_j(x_j)) = 0$. An additive model with finite variance has mean dimension $\nu(f_A) = 1$. It represents one extreme in terms of mean dimension. The second function we consider is a

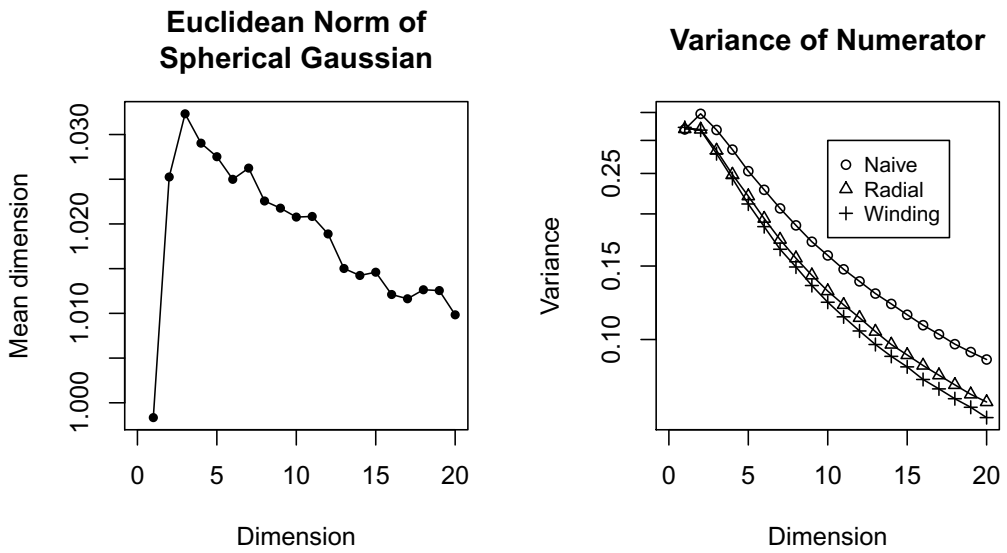

Figure 3. *The left panel shows low and mostly decreasing estimates of $\nu(f)$ versus dimension for $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_2$ when $\boldsymbol{x} \sim \mathcal{N}(0, I)$. The right panel shows variances of estimates of δ for this function.*

product model

$$f_P(\boldsymbol{x}) = \prod_{j=1}^{d} g_j(x_j), \tag{9}$$

where $\mathbb{E}(g_j(x_j)) = \mu_j$ and $\mathrm{Var}(g_j(x_j)) = \sigma_j^2$. Product functions are frequently used as test functions. For instance, Sobol's g-function (Saltelli and Sobol', 1995) is the product $\prod_{j=1}^{d}(|4x_j - 2| + a_j)/(1 + a_j)$ in which later authors make various choices for the constants a_j.

If all $\mu_j = 0$, then $\nu(f_P) = d$. In general, the mean dimension of a product function is

$$\nu(f_P) = \frac{\sum_{j=1}^{d} \sigma_j^2/(\mu_j^2 + \sigma_j^2)}{1 - \prod_{j=1}^{d} \mu_j^2/(\mu_j^2 + \sigma_j^2)}.$$

See Owen (2003).

Additive and multiplicative functions comprise two extremes in mean dimension. Additive functions always have mean dimension 1. While multiplicative functions can have any mean dimension in the interval $(1, d]$ they are easily engineered to provide functions with mean dimension d by setting all $\mu_j = 0$.

4.1. Additive functions. We will use Lemma 1 below to compare the variances of our mean dimension estimators for additive functions. For these, we need the kurtosis of some random variables. Recall that the kurtosis of a random variable Y with variance $\sigma^2 > 0$ is $\kappa = \mathbb{E}((Y - \mu)^4)/\sigma^4 - 3$ which can be infinite. As points of reference, if Y is Gaussian, then

$\kappa = 0$ and if Y has a uniform distribution, then $\kappa = -6/5$ and the smallest possible kurtosis is -2.

Lemma 1. *Let Y_1, Y_2, Y_3, Y_4 be independent identically distributed random variables with variance σ^2 and kurtosis κ. Then*

$$\begin{aligned}\mathbb{E}\big((Y_1 - Y_2)^4\big) &= (12 + 2\kappa)\sigma^4,\\ \mathrm{Var}\big((Y_1 - Y_2)^2\big) &= (8 + 2\kappa)\sigma^4,\\ \mathbb{E}\big((Y_1 - Y_2)^2(Y_3 - Y_4)^2\big) &= 4\sigma^4,\\ \mathbb{E}\big((Y_1 - Y_2)^2(Y_1 - Y_3)^2\big) &= (6 + \kappa)\sigma^4.\end{aligned}$$

Proof. These follow directly from independence of the Y_j and the definitions of variance and kurtosis. ∎

Theorem 1. *For the additive function f_A of (8),*

$$\mathrm{Var}(\tilde{\delta}) = \mathrm{Var}(\hat{\delta}) = \mathrm{Var}(\ddot{\delta}) = \frac{1}{N}\sum_{j=1}^{d}\Big(2 + \frac{\kappa_j}{2}\Big)\sigma_j^4 \tag{10}$$

and

$$\mathrm{Var}(\check{\delta}) = \mathrm{Var}(\ddot{\delta}) + \frac{N-1}{2N^2}\sum_{j=1}^{d}(\kappa_j + 2)\sigma_j^4. \tag{11}$$

Proof. The winding stairs results for $\check{\delta}$ and $\ddot{\delta}$ quoted above are proved in Theorem 3 of the appendix. For the naive estimate, $\widehat{\tau}_j^2$ is independent of $\widehat{\tau}_k^2$ when $j \neq k$ as remarked upon at (4). For an additive function

$$f_A(\boldsymbol{x}_i) - f_A(\boldsymbol{x}_{i,-j}{:}\boldsymbol{z}_{i,j}) = g_j(x_{ij}) - g_j(z_{ij})$$

is independent of $g_k(x_{ik}) - g_k(z_{ik})$ for $j \neq k$ and so the radial estimate has the same independence property as the naive estimate. Therefore,

$$\mathrm{Var}(\widehat{\tau}_j^2) = \mathrm{Var}(\widetilde{\tau}_j^2) = \frac{1}{4N}\mathrm{Var}\big((g_j(x_{1j}) - g_j(z_{1j}))^2\big),$$

and using Lemma 1, $\mathrm{Var}((g_j(x_{1j}) - g_j(z_{1j}))^2) = (8 + 2\kappa_j)\sigma_j^4$. ∎

If $f(\boldsymbol{x})$ is additive, then Theorem 1 shows that the radial method is better than the naive one. They have the same variance, but the naive method uses roughly twice as many function evaluations. If the function is nearly additive, then it is reasonable to expect the variances to be nearly equal and the radial method to be superior. Because $\kappa_j \geqslant -2$ always holds, the theorem shows an advantage to truncated winding stairs over plain winding stairs.

4.2. Multiplicative functions. We turn next to functions of product form. For the factors $g_j(x_j)$ defining f in (9), we let $\mu_{2j} = \mathbb{E}(g_j(x_j)^2)$, $\mu_{3j} = \mathbb{E}(g_j(x_j)^3)$, and $\mu_{4j} = \mathbb{E}(g_j(x_j)^4)$. To simplify some expressions for winding stairs we adopt the conventions that for $1 \leqslant j < k \leqslant d$ and quantities q_ℓ, $\prod_{\ell\in(j,k)} q_\ell$ means $\prod_{\ell=j+1}^{k-1} q_\ell$ and $\prod_{\ell\notin[j,k]} q_\ell$ means $\prod_{\ell=1}^{j-1} q_\ell \times \prod_{\ell=k+1}^{d} q_\ell$, with products over empty index sets equal to one.

Theorem 2. *For the product function f_P of (9),*

$$\mathrm{Var}(\hat\delta) = \frac{1}{N}\sum_{j=1}^{d}\sigma_j^4\left(\Big(3+\frac{\kappa_j}{2}\Big)\prod_{\ell\neq j}\mu_{4\ell} - \prod_{\ell\neq j}\mu_{2\ell}^2\right) \quad \textit{and} \tag{12}$$

$$\mathrm{Var}(\tilde\delta) = \mathrm{Var}(\hat\delta) + \frac{2}{N}\sum_{j<k}\Big(\frac{\eta_j\eta_k}{4} - \sigma_j^2\sigma_k^2\mu_{2j}\mu_{2k}\Big)\prod_{\ell\notin\{j,k\}}\mu_{4\ell}, \tag{13}$$

where $\eta_j = \mathbb{E}(g_j(x_j)^2(g_j(x_j)-g_j(z_j))^2) = \mu_{4j} - 2\mu_j\mu_{3j} + \mu_{2j}^2$ for independent $x_j, z_j \sim P_j$. The winding stairs estimates satisfy

$$\mathrm{Var}(\ddot\delta) = \mathrm{Var}(\hat\delta) + \frac{2}{N}\sum_{j<k}\left(\frac{\eta_j\eta_k}{4}\prod_{\ell\in(j,k)}\mu_{2\ell}^2\prod_{\ell\notin[j,k]}\mu_{4\ell} - \sigma_j^2\sigma_k^2\mu_{2j}\mu_{2k}\prod_{\ell\notin j:k}\mu_{2\ell}^2\right) \tag{14}$$

and

$$\mathrm{Var}(\check\delta) = \mathrm{Var}(\ddot\delta) + \frac{2}{N}\sum_{j<k}\left(\frac{\eta_j\eta_k}{4}\prod_{\ell\notin j:k}\mu_{4\ell} - \sigma_j^2\sigma_k^2\prod_{\ell\notin j:k}\mu_{2\ell}^2\right)\prod_{\ell\in(j,k)}\mu_{2\ell}^2. \tag{15}$$

Proof. The winding stairs results are from Theorem 4 in the appendix. Next, we turn to the naive estimator. For $\boldsymbol{x}, \boldsymbol{z} \sim P$ independently, define $\Delta_j = \Delta_j(\boldsymbol{x}, \boldsymbol{z}) \equiv f_P(\boldsymbol{x}) - f_P(\boldsymbol{x}_{-j}{:}\boldsymbol{z}_j)$. Now

$$\Delta_j = (g_j(x_j) - g_j(z_j)) \times \prod_{\ell\neq j} g_\ell(x_\ell),$$

and so $\mathbb{E}(\Delta_j^2) = 2\sigma_j^2 \times \prod_{\ell\neq j}\mu_{2\ell}$ and $\mathbb{E}(\Delta_j^4) = (12+2\kappa_j)\sigma_j^4 \times \prod_{\ell\neq j}\mu_{4j}$, from Lemma 1. Therefore,

$$\mathrm{Var}(\Delta_j^2) = (12+2\kappa_j)\sigma_j^4 \times \prod_{\ell\neq j}\mu_{4j} - 4\sigma_j^4 \times \prod_{\ell\neq j}\mu_{2\ell}^2,$$

establishing (12).

In the radial estimate, Δ_j is as above and $\Delta_k = (g_k(x_k) - g_k(z_k)) \times \prod_{\ell\neq k} g_\ell(x_\ell)$. In this case, however, the same point $\boldsymbol{x}$ is used in both Δ_j and Δ_k so $\mathbb{E}(\Delta_j^2\Delta_k^2)$ equals

$$\mathbb{E}\left(g_j(x_j)^2 g_k(x_k)^2 (g_j(x_j)-g_j(z_j))^2 (g_k(x_k)-g_k(z_k))^2 \prod_{\ell\notin\{j,k\}} g_\ell(x_\ell)^4\right)$$
$$= \eta_j\eta_k \prod_{\ell\notin\{j,k\}}\mu_{4\ell}.$$

Then $\mathrm{Cov}(\Delta_j^2, \Delta_k^2) = \big(\eta_j\eta_k - 4\sigma_j^2\sigma_k^2\mu_{2j}\mu_{2k}\big)\prod_{\ell\notin\{j,k\}}\mu_{4\ell}$, establishing (13). ∎

We comment below on interpretations of the winding stairs quantities. First, we compare naive to radial sampling.

As an illustration, suppose that $g_j(x_j) \sim \mathcal{N}(0,1)$ for $j = 1, \dots, d$. Then

$$\mathrm{Var}(\hat{\delta}) = \frac{1}{N}\sum_{j=1}^{d}(3^d - 1) = \frac{d(3^d-1)}{N},$$

and since this example has $\eta_j = 4$,

$$\mathrm{Var}(\tilde{\delta}) = \frac{d(3^d-1)}{N} + \frac{2}{N}\sum_{j<k}\Big(\frac{16}{4} - 1\Big)3^{d-2} = \frac{d(3^d-1)}{N} + \frac{2d(d-1)3^{d-1}}{N}.$$

For large d the radial method has variance about $2d/3$ times as large as the naive method. Accounting for the reduced sample size of the radial method it has efficiency approximately $3/d$ compared to the naive method for this function.

A product of mean zero functions has mean dimension d making it an exceptionally hard case. More generally, if $\eta_j/2 - \sigma_j^2\mu_{2j} \geqslant \epsilon > 0$ for $j \in 1{:}d$, then $\mathrm{Var}(\hat{\delta}) = O(d/N)$ while $\mathrm{Var}(\tilde{\delta})$ is larger than a multiple of d^2/N.

Corollary 1. *For the product function f_P of (9), suppose that $\kappa_j \geqslant -5/16$ for $j = 1, \dots, d$. Then $\mathrm{Cov}(\widetilde{\tau}_j^2, \widetilde{\tau}_k^2) \geqslant 0$ for $1 \leqslant j < k \leqslant d$, and so $\mathrm{Var}(\tilde{\delta}) \geqslant \mathrm{Var}(\hat{\delta})$.*

Proof. It suffices to show that $\eta_j > 2\sigma_j^2\mu_{2j}$ for $j = 1, \dots, d$. Let $Y = g_j(x_j)$ for $x_j \sim P_j$ have mean μ, uncentered moments μ_{2y}, μ_{3y}, and μ_{4y} of orders 2, 3, and 4, respectively, variance σ^2, skewness γ, and kurtosis κ. Now let $\eta = \mu_{4y} - 2\mu\mu_{3y} + \mu_{2y}^2$. This simplifies to

$$\eta = (\kappa+2)\sigma^4 + 2\mu\sigma^3\gamma + 2\mu^2\sigma^2 + \sigma^4$$

and so

$$\eta - 2\sigma^2\mu_{2y} = (\kappa+2)\sigma^4 + 2\mu\sigma^3\gamma + \mu^2\sigma^2.$$

If $\sigma = 0$, then $\eta - 2\sigma^2\mu_{2y} = 0$ and so we suppose that $\sigma > 0$. Replacing Y by Y/σ does not change the sign of $\eta - 2\sigma^2\mu_{2y}$. It becomes $\kappa + 2 + 2\mu_*\gamma + \mu_*^4$ for $\mu_* = \mu/\sigma$. If γ and μ_* have equal signs, then $\kappa + 2 + 2\mu_*\gamma + \mu_*^4 \geqslant 0$, so we consider the case where they have opposite signs. Without loss of generality we take $\gamma < 0 < \mu_*$. An inequality of Rohatgi and Székely (1989) shows that $|\gamma| \leqslant \sqrt{\kappa+2}$, and so

$$\kappa + 2 + 2\mu_*\gamma + \mu_*^4 \geqslant \theta^2 - 2\mu_*\theta + \mu_*^4 \tag{16}$$

for $\theta = \sqrt{\kappa+2}$. Equation (16) is minimized over $\mu_* \geqslant 0$ at $\mu_* = (\theta/2)^{1/3}$ and so $\kappa + 2 + 2\mu_*\gamma + \mu_*^4 \geqslant \theta^2 + \big(2^{-4/3} - 2^{2/3}\big)\theta^{4/3}$. One last variable change to $\theta = (2\lambda)^3$ gives

$$\kappa + 2 + 2\mu_*\gamma + \mu_*^4 \geqslant \lambda^4(4\lambda^2 - 3).$$

This is nonnegative for $\lambda \geqslant (3/4)^{1/2}$, equivalently $\theta \geqslant 2(3/4)^{3/2}$, and finally for $\kappa \geqslant -5/16$. ∎

From the above discussion we can see that large kurtoses, and hence large values of $\mu_{4j} = \mathbb{E}(g_j(x_j)^4)$, create difficulties. In this light we can compare winding stairs to the radial sampler. The covariances in the radial sampler involve a product of $d-2$ of the μ_{4j}. The winding stairs estimates involve products of fewer of those quantities. For truncated winding stairs the j,k-covariance includes a product of only $d-k+j-1$ of them. The values $\mu_{4\ell}$ for ℓ nearest to 1 and d appear the most often and so the ordering of the variables makes a difference. For regular winding stairs some additional fourth moments appear in a second term.

5. Example: MNIST classification. In this section, we investigate the mean dimension of a neural network classifier that predicts a digit in $\{0,1,\dots,9\}$ based on an image of 784 pixels. We compare algorithms for finding mean dimension, investigate some mean dimensions, and then plot some images of Sobol' indices.

The MNIST data set from http://yann.lecun.com/exdb/mnist/ is a very standard benchmark problem for neural networks. It consists of 70,000 images of handwritten digits that were size-normalized and centered within 28×28 pixel gray scale images. We normalize the image values to the unit interval, $[0,1]$. The prediction problem is to identify which of the ten digits "0," "1," ..., "9" is in one of the images based on $28^2 = 784$ pixel values. We are interested in the mean dimension of a fitted prediction model.

The model we used is a convolutional neural network fit via tensorflow (Abadi et al., 2016). The architecture applied the following steps to the input pixels in order:

(1) a convolutional layer (with 28 kernels, each of size 3×3),
(2) a max pooling layer (over 2×2 blocks),
(3) a flattening layer,
(4) a fully connected layer with 128 output neurons (ReLU activation),
(5) a dropout layer (node values were set to 0 with probability 0.2), and
(6) a final fully connected layer with 10 output neurons (softmax activation).

This model is from Yalcin (2018) who also defines those terms. The network was trained using 10 epochs of ADAM optimization, also described in Yalcin (2018), on 60,000 training images. For our purposes, it is enough to know that it is a complicated black box function of 784 inputs. The accuracy on 10,000 held out images was 98.5%. This is not necessarily the best accuracy attained for this problem, but we consider it good enough to make the prediction function worth investigating.

There are $2^{784} - 1 > 10^{236}$ nontrivial sets of pixels, each making their own contribution to the prediction functions, but the mean dimension can be estimated by summing only 784 Sobol' indices.

We view the neural network's prediction as a function on 784 input variables $\boldsymbol{x}$. For data $(\boldsymbol{x}, Y)$ where $Y \in \{0,1,\dots,9\}$ is the true digit of the image, the estimated probability that $Y = y$ is given by

$$f_y(\boldsymbol{x}) = \frac{\exp(g_y(\boldsymbol{x}))}{\sum_{\ell=0}^{9} \exp(g_\ell(\boldsymbol{x}))}$$

for functions g_y, $0 \leqslant y \leqslant 9$. This last step, called the softmax layer, exponentiates and normalizes functions g_y that implement the prior layers. We study the mean dimension of $g_0,\dots,g_9$ as well as the mean dimension of $f_0,\dots,f_9$. Studying the complexity of predictions

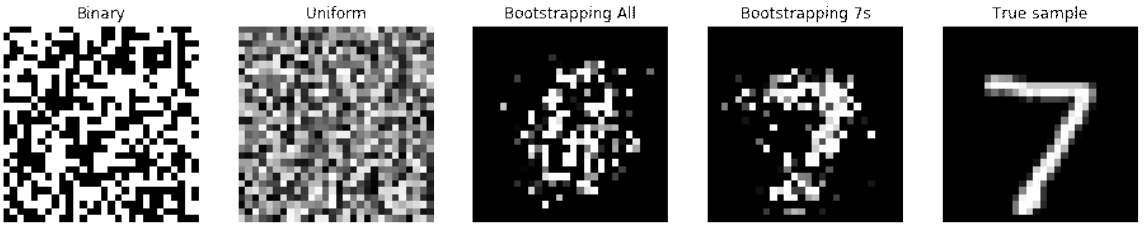

Figure 4. *From left to right: draws from* $\mathbf{U}\{0,1\}^{28\times 28}$, $\mathbf{U}[0,1]^{28\times 28}$, *margins of all images, margins of all* 7*s, an example* 7.

via the inputs to softmax has been done earlier by Yosinski et al. (2015).

To compute mean dimension we need to have a model for $\boldsymbol{x}$ with 784 independent components. Real images are only on or near a very small manifold within $\mathbb{R}^{784}$. We considered several distributions P_j for the value of pixel j: $\mathbf{U}\{0,1\}$ (salt and pepper), $\mathbf{U}[0,1]$ (random gray), independent resampling from per pixel histograms of all images, and independent resampling per pixel just from images with a given value of $y \in \{0,1,\dots,9\}$. The histogram of values for pixel j from those images is denoted by $h_y(j)$ with h_y representing all 784 of them. Figure 4 shows some sample draws along with one real image. We think that resampling pixels from images given y is the most relevant of these methods, though ways to get around the independence assumption would be valuable. We nonetheless include the other samplers in our computations.

Our main interest is in comparing the variance of estimates of δ. We compared the naive method $\hat{\delta}$, the radial method $\tilde{\delta}$, and truncated winding stairs $\ddot{\delta}$. For $\ddot{\delta}$ our winding stairs algorithm changed pixels in raster order, left to right within rows, taking rows of the image from top to bottom. We omit $\check{\delta}$ because we think there is no benefit from its more complicated model and additional correlations. Our variance comparisons are based on $N = 100{,}000$ samples.

Figure 5 shows the results for all 10 output values y, and all 11 different input histogram distributions. Ten of those histograms are from resampling pixel values within categories, and the eleventh is a pooled histogram. There are separate plots for functions f_y that include softmax and g_y that exclude it. The radial method always had greater variance than the naive method. For functions g_y it never had as much as twice the variance of the naive method, and so the radial method proves better for g_y. For f_y there were some exceptions where the naive method is more efficient. In all of our comparisons the winding stairs method had lower variance than the radial method, and so for these functions, (truncated) winding stairs is clearly the best choice.

Figure 5 is a summary of 660 different variance estimates. We inspected the variances and found two more things worth mentioning but not presenting. The variances were all far smaller using softmax than not, which is not surprising since softmax compresses the range of f_y to be within $[0,1]$ which will greatly affect the differences that go into estimates of δ. The variances did not greatly depend on the input distribution. While there were some statistically significant differences, which is almost inevitable for such large N, the main practical difference

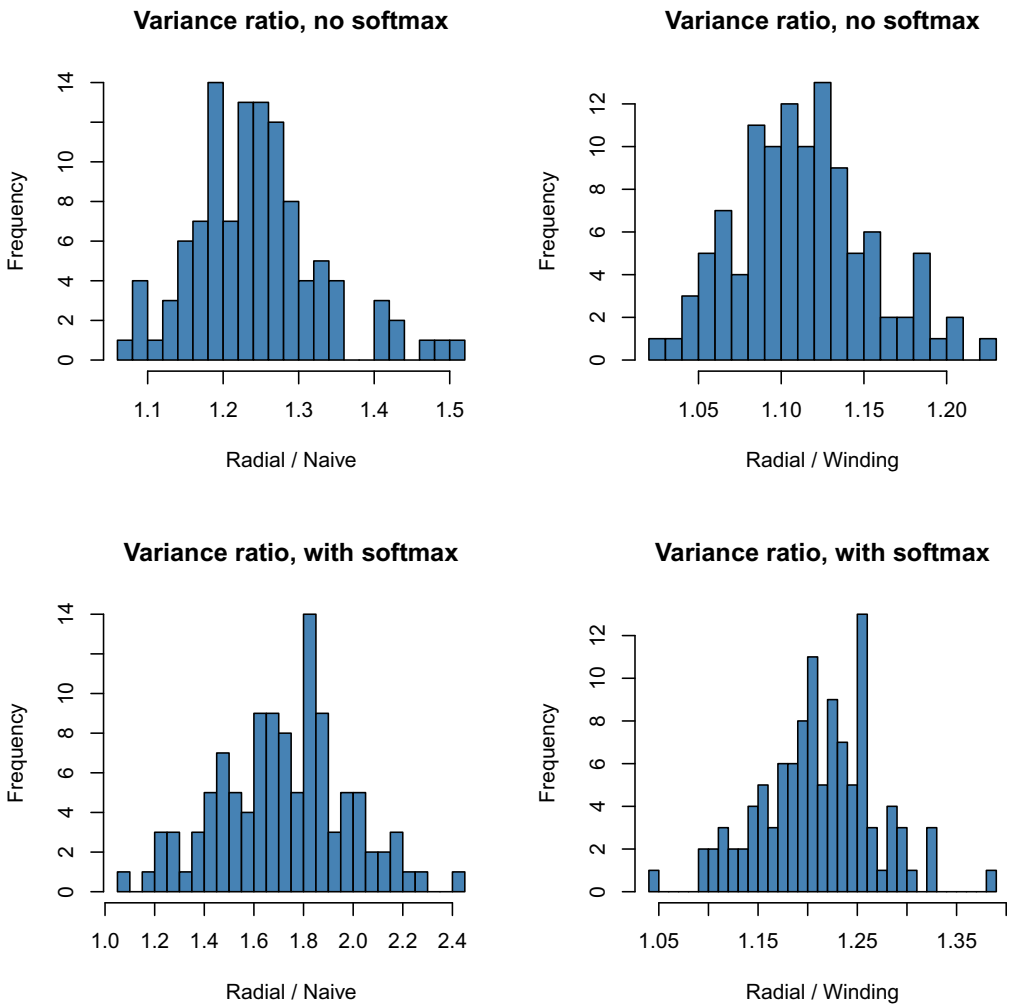

Figure 5. *The upper left histogram shows* $\mathrm{Var}(\tilde{\delta})/\mathrm{Var}(\hat{\delta})$ *for functions* g_y *that exclude softmax. The upper right histogram shows* $\mathrm{Var}(\tilde{\delta})/\mathrm{Var}(\ddot{\delta})$. *The bottom two show the same ratios for functions* f_y *that include softmax. The histograms include all* 10 *values of output* y, *and all* 10 y*-specific input histograms and the pooled input histogram.*

was that variances tended to be much smaller when sampling from h_1. We believe that this is because images for $y = 1$ have much less total illumination than the others.

While our main purpose is to compare estimation strategies for mean dimension, the mean dimensions for this problem are themselves of interest. Table 1 shows mean dimensions for functions f_y that include softmax as estimated via winding stairs. For this we used $N = 10^6$ when resampling from images $h_0, \ldots, h_9$ and $N = 2 \times 10^6$ otherwise. The first thing to note is an impossible estimate of $\nu(f_1)$ for binary and uniform sampling. The true $\nu(f_1)$ cannot be larger than 784. The function f_1 has tiny variance under those distributions, and we recall that $\nu = \delta/\sigma^2$. Next, we see that moving from binary to uniform to the combined histogram

Table 1
Estimated mean dimension of functions f_y using softmax.

Sampler	0	1	2	3	4	5	6	7	8	9
Binary	11.07	936.04	10.43	9.92	18.69	10.22	13.27	13.37	8.67	16.54
Uniform	6.92	4,108.99	7.28	6.60	9.90	7.03	6.92	8.03	5.61	9.48
Combined	8.77	4.68	4.06	3.95	4.56	5.11	7.62	4.62	3.43	7.39
0	3.52	6.81	3.48	7.20	6.56	5.78	7.54	4.67	4.04	9.08
1	36.12	2.88	6.00	3.43	7.75	3.76	8.74	7.60	2.83	5.58
2	10.03	3.86	3.68	4.70	8.23	12.27	12.57	7.20	4.31	17.23
3	23.20	4.69	5.95	4.10	6.96	6.72	13.63	7.10	4.42	9.00
4	7.42	8.39	7.59	9.96	3.81	7.63	8.57	5.35	3.86	6.82
5	8.12	4.77	5.72	4.82	5.60	3.48	7.61	7.28	3.54	7.87
6	9.22	5.65	4.36	6.52	4.31	6.67	3.57	6.43	4.28	11.99
7	8.57	5.85	4.42	4.09	4.66	5.09	3.59	3.59	4.29	5.58
8	19.58	6.06	4.54	4.77	8.21	6.28	13.15	6.72	4.20	10.11
9	7.47	7.00	5.25	4.96	3.15	4.52	7.34	3.74	2.92	3.48

generally lowers the mean dimension. Third, for the y-specific histograms h_y we typically see smaller mean dimensions for f_y with the same y that was used in sampling. That is, the diagonal of the lower block tends to have smaller values.

Table 2 shows mean dimensions for functions g_y that exclude softmax as estimated via winding stairs. They are all in the range from 1.35 to 1.92. We found no particular problem with the function g_1 like we saw for f_1. While the functions g_y that are sent into softmax were obtained by a very complicated process, they do not make much use of very high order interactions. There must be a significantly large component of additive functions and two factor interactions within them. There may be a small number of large high order interactions but they do not dominate any of the functions f_y under any of the sampling distributions we use. The softmax function begins by exponentiating f_y which we can think of as changing a function with a lot of additive structure into one with a lot of multiplicative structure. Multiplicative functions can have quite high mean dimension.

The measured mean dimensions of g_y are pretty stable as the sampling distribution changes. While the manifold of relevant images is likely to be quite small, it is reassuring that 13 different independent data distributions give largely consistent and small mean dimensions.

Figure 6 shows some Sobol' indices of f_y and g_y for $y \in \{0, 1, \dots, 9\}$ when sampling from h_0. In each set of 10 images, the gray scale goes from black for 0 to white for the largest intensity in any of those 10 images. As a consequence some of the images are almost entirely black.

The lower indices $\underline{\tau}_j^2$ depict the importance of inputs one at a time. This is similar to what one gets from a gradient (see, for instance, Grad-CAM (Selvaraju et al., 2017)), except that $\underline{\tau}_j^2$ is global over the whole range of the input instead of local like a gradient. Upper indices $\bar{\tau}_j^2$ depict the importance of each pixel combining all of the interactions to which it contributes, not just its main effect.

For the influence on f_0 when sampling from h_0, the difference between $\underline{\tau}_j^2$ and $\bar{\tau}_j^2$ is in that bright spot just left of the center of the image. That is the region of pixels involved in

Table 2
Estimated mean dimension of functions g_y without softmax.

Sampler	0	1	2	3	4	5	6	7	8	9
Binary	1.66	1.76	1.74	1.72	1.73	1.79	1.75	1.69	1.74	1.79
Uniform	1.65	1.62	1.66	1.66	1.67	1.71	1.71	1.61	1.68	1.70
Combined	1.79	1.77	1.70	1.73	1.73	1.90	1.88	1.78	1.90	1.89
0	1.92	1.65	1.68	1.69	1.65	1.80	1.86	1.56	1.68	1.81
1	1.48	1.56	1.35	1.61	1.62	1.57	1.49	1.42	1.56	1.50
2	1.55	1.66	1.62	1.74	1.57	1.72	1.67	1.61	1.78	1.59
3	1.56	1.65	1.59	1.58	1.63	1.85	1.59	1.64	1.67	1.66
4	1.87	1.62	1.61	1.55	1.70	1.75	1.76	1.66	1.57	1.78
5	1.71	1.60	1.59	1.63	1.72	1.78	1.74	1.62	1.76	1.90
6	1.65	1.60	1.60	1.66	1.68	1.70	1.65	1.60	1.54	1.63
7	1.73	1.59	1.61	1.63	1.60	1.62	1.65	1.57	1.59	1.63
8	1.73	1.65	1.60	1.64	1.66	1.78	1.75	1.64	1.84	1.75
9	1.86	1.68	1.61	1.63	1.73	1.80	1.86	1.67	1.69	1.82

the most interactions. It appears to be involved in distinguishing 0s from 2s and 8s because that region is also bright for functions f_2 and f_8. Without softmax that bright spot for $\bar{\tau}_j^2$ is lessened and so we see that much though not all of its interaction importance was introduced by the softmax layer. For g_5 when sampling from h_0 we see that a region just northeast of the center of the image has the most involvement in interactions as measured by $\bar{\tau}_j^2$.

6. Discussion. We have found that the strategy under which differences of function values are collected can make a big difference in the statistical efficiency of estimates of mean dimension. Computational efficiency in reusing function values can increase some correlations enough to more than offset that advantage. Whether this happens depends on the function involved. We have seen examples where high kurtoses make the problem worse.

Our interest in mean dimension leads us to consider sums of $\bar{\tau}_j^2$. In other uncertainty quantification problems we are interested in comparing and ranking $\bar{\tau}_j^2$. For a quantity like $\hat{\bar{\tau}}_j^2 - \hat{\bar{\tau}}_k^2$ we actually prefer a large positive value for $\mathrm{Cov}(\hat{\bar{\tau}}_j^2, \hat{\bar{\tau}}_k^2)$. In this case, the disadvantages we described for the radial method become a strength. Correlation effects are more critical for mean dimension than for these differences of Sobol' indices, because mean dimension is affected by $O(d^2)$ covariances, not just one.

The radial strategy and the truncated winding stairs strategy can both be represented in terms of a tree structure connecting $d+1$ function values. There is a one to one correspondence between the d edges in that tree and the components of $\boldsymbol{x}$ getting changed. There is no particular reason to think that either of these strategies is the optimal graph structure or even the optimal tree.

The mean dimension derives from an ANOVA decomposition that in turn is based on models with independent inputs. There has been work on ANOVA for dependent inputs, such as Stone (1994), Hooker (2007), and Chastaing, Gamboa, and Prieur (2012, 2015). The underlying models require the density to have an unrealistically strong absolute continuity property with respect to a product measure that makes them unrealistic for the MNIST example. There are also approaches to global sensitivity analysis based on Shapley values that do not require independence of the underlying variables (Song, Nelson, and Staum, 2016;

Owen and Prieur, 2017).

Recent work by Hart and Gremaud (2018) shows how to define some Sobol' indices directly without recourse to the ANOVA, which may provide a basis for mean dimension without ANOVA. Kucherenko, Tarantola, and Annoni (2012) have a copula based approach to Sobol' indices on dependent data, though finding a specific copula that describes points near a manifold would be hard.

We have studied the accuracy of mean dimension estimates as if the sampling were done by plain Monte Carlo (MC). When P is the uniform distribution on $[0,1]^d$ then we can instead use randomized quasi-Monte Carlo (RQMC) sampling, surveyed in L'Ecuyer and Lemieux (2002). The naive method can be implemented using N points in $[0,1]^{d+1}$ for each of $j = 1, \dots, d$. The first column of the jth input matrix could contain $\boldsymbol{z}_{ij}$ for $i = 1, \dots, N$ while the remaining d columns would have $\boldsymbol{x}_i^{(j)} \in [0,1]^d$. The $d+1$st point contains the values $\boldsymbol{x}_{i,j}$. The radial method can be implemented with N points in $[0,1]^{2d}$ with the first d columns providing $\boldsymbol{x}_i$ and the second d columns providing $\boldsymbol{z}_i$, both for $i = 1, \dots, N$. Truncated winding stairs similarly requires N points in $[0,1]^{2d}$. For RQMC sampling by scrambled nets, the resulting variance is $o(1/N)$. A reasonable choice is to use RQMC in whichever method one thinks would have the smallest MC variance. The rank ordering of RQMC variances could, however, be different from that of MC and it could even change with N, so results on MC provide only a suggestion of which method would be best for RQMC.

A QMC approach to plain winding stairs would require QMC methods designed specifically for MCMC sampling; see, for instance, one based on completely uniformly distributed sequences described in Owen and Tribble (2005).

We have used a neural network black box function to illustrate our computations. It is yet another example of an extremely complicated function that nonetheless is dominated by low order interactions. In problems like this where the input images had a common registration, an individual pixel has some persistent meaning between images and then visualizations of $\underline{\tau}_j^2$ can be informative. Many neural network problems are applied to data that have not been so carefully registered as the MNIST data. For those problems the link from predictions back to inputs may need to be explored in a different way.

Appendix A. Covariances under winding stairs. Winding stairs expressions are more complicated than the others and require somewhat different notation. Hence we employ some notation local to this appendix. For instance, in winding stairs $\ell(i)$ has a special meaning as a newly updated component of $\boldsymbol{x}_i$. Accordingly when we need a variable index other than j and k we use t instead of ℓ in this appendix. We revert the ts back to ℓ when quoting these theorems in the main body of the paper. Similarly, differences in function values are more conveniently described via which observation i is involved and not which variable. Accordingly, we work with Δ_i here instead of Δ_j in the main body of the article.

We begin with the regular winding stairs estimates and let $\Delta_i = f(\boldsymbol{x}_i) - f(\boldsymbol{x}_{i-1})$. For $i' > i$, the differences Δ_i and $\Delta_{i'}$ are independent if $\boldsymbol{x}_{i'-1}$ has no common component with $\boldsymbol{x}_i$. This happens if $i' - 1 \geqslant i + d$, that is, if $i' - i > d$. For any index i, the difference Δ_i may be dependent on $\Delta_{i'}$ for $-d < i' < d$ but no other $\Delta_{i'}$. It is not necessarily true that $\mathrm{Cov}(\Delta_i^2, \Delta_{i+s}^2) = \mathrm{Cov}(\Delta_i^2, \Delta_{i-s}^2)$ because different shared components of $\boldsymbol{x}$ are involved in these two covariances.

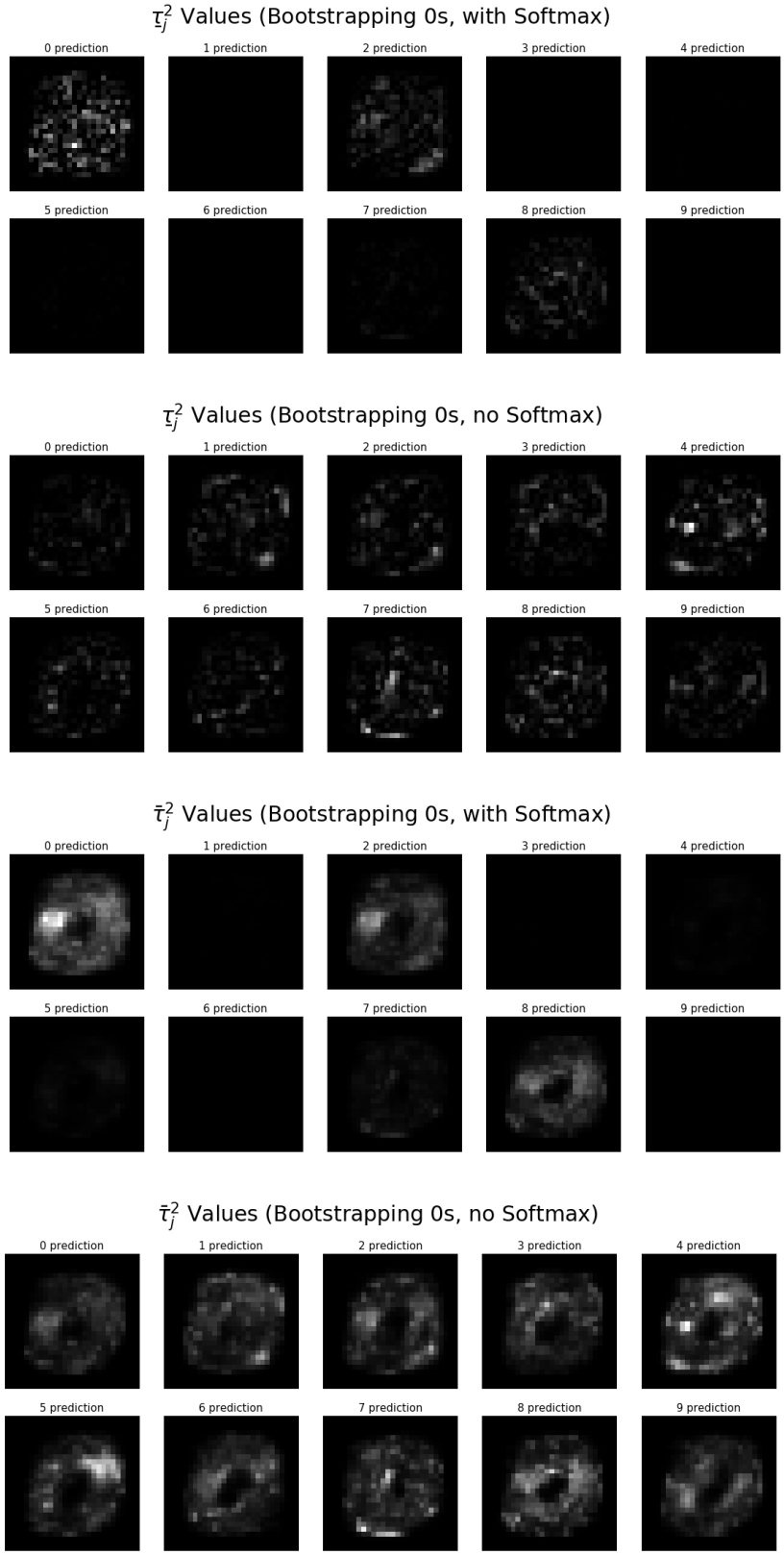

Figure 6. *From top to bottom: maps of* $\underline{\tau}_j^2(f_y)$, $\underline{\tau}_j^2(g_y)$, $\bar{\tau}_j^2(f_y)$, *and* $\bar{\tau}_j^2(g_y)$ *versus pixels* j *when sampling from* h_0.

The winding stairs estimate of $\bar\tau_j^2$ is $\breve\tau_j^2 = (1/(2N))\sum_{i=1}^N \Delta^2_{d(i-1)+j}$. Because $\mathrm{Cov}(\Delta^2_{i+d}, \Delta^2_{i'+d}) = \mathrm{Cov}(\Delta_i^2, \Delta_{i'}^2)$, we find that for $1 \leqslant j < k \leqslant d$,

$$\mathrm{Cov}(\breve\tau_j^2, \breve\tau_k^2) = \frac{1}{4N}\Big(\mathrm{Cov}(\Delta^2_{d+j}, \Delta^2_{d+k}) + \mathrm{Cov}(\Delta^2_{2d+j}, \Delta^2_{d+k})\Big). \tag{17}$$

The truncated winding stairs algorithm has

$$\mathrm{Cov}(\ddot\tau_j^2, \ddot\tau_k^2) = \frac{1}{4N}\mathrm{Cov}(\Delta^2_{d+j}, \Delta^2_{d+k}) \tag{18}$$

because Δ_{2d+j} has no zs in common with Δ_{d+k}.

Theorem 3. *For the additive function f_A of (8),*

$$\mathrm{Var}(\breve\delta) = \frac{1}{N}\sum_{j=1}^d \Big(2 + \frac{\kappa_j}{2}\Big)\sigma_j^4 + \frac{N-1}{2N^2}\sum_{j=1}^d(\kappa_j+2)\sigma_j^4, \tag{19}$$

$$\mathrm{Var}(\ddot\delta) = \frac{1}{N}\sum_{j=1}^d \Big(2 + \frac{\kappa_j}{2}\Big)\sigma_j^4. \tag{20}$$

Proof. For an additive function under winding stairs

$$\begin{aligned}\Delta_{d(i-1)+j} &= g_j(\boldsymbol{x}_{d(i-1)+j,j}) - g_j(\boldsymbol{x}_{d(i-2)+j,j})\\ &= g_j(z_{d(i-1)+j}) - g_j(z_{d(i-2)+j})\end{aligned}$$

because $r(i,j) = d\lfloor (i-j)/d\rfloor + j$ yields $r(d(i-1)+j, j) = d(i-1)+j$. It follows that $\breve\tau_j^2$ and $\breve\tau_k^2$ have no zs in common when $j \neq k$ and so they are independent. Now define the independent and identically distributed random variables $Y_i = g_j(z_{d(i-1)+j})$ for $i = 1, \dots, N$. Then

$$\begin{aligned}\mathrm{Var}(\breve\tau_j^2) &= \mathrm{Var}\left(\frac{1}{2N}\sum_{i=1}^N (Y_i - Y_{i-1})^2\right)\\ &= \frac{1}{4N}\mathrm{Var}((Y_1-Y_0)^2) + \frac{N-1}{2N^2}\mathrm{Cov}((Y_1-Y_0)^2, (Y_2-Y_1)^2)\\ &= \frac{(8+2\kappa_j)\sigma^4}{4N} + \frac{(N-1)(\kappa+2)\sigma^4}{2N^2}\end{aligned}$$

by Lemma 1, establishing (19). For truncated winding squares all of the Δ_i are independent in the additive model establishing (20). ∎

Next, we turn to the multiplicative model $f_P(\boldsymbol{x}_i) = \prod_{j=1}^d g_j(z_{r(i,j)})$. A key distinction arises for variables "between" the jth and kth and variables that are not between those. For $j < k$ the indices t between them are designated by $t \in (j,k)$ and the ones "outside" of them are designated by $t \notin [j,k]$, meaning that $t \in \{1, \dots, j-1\} \cup \{k+1, \dots, d\}$. Recall that $\mu_{\ell j}$ is $\mathbb{E}(g_j(x_j)^\ell)$ for $\ell = 2, 3, 4$.

Theorem 4. *For the multiplicative function f_P of (9),*

$$\begin{aligned}\mathrm{Var}(\ddot{\delta}) &= \frac{1}{N}\sum_{j=1}^{d}\sigma_j^4\left(\Big(3+\frac{\kappa_j}{2}\Big)\prod_{t\neq j}\mu_{4t}-\prod_{t\neq j}\mu_{2t}^2\right)\\ &\quad+\frac{2}{N}\sum_{j<k}\left(\frac{\eta_j\eta_k}{4}\prod_{t\in(j,k)}\mu_{2t}^2\prod_{t\notin[j,k]}\mu_{4t}-\sigma_j^2\sigma_k^2\mu_{2j}\mu_{2k}\prod_{t\notin\{j,k\}}\mu_{2t}^2\right)\end{aligned} \tag{21}$$

and

$$\mathrm{Var}(\check{\delta}) = \mathrm{Var}(\ddot{\delta}) + \frac{2}{N}\sum_{j<k}\left(\frac{\eta_j\eta_k}{4}\prod_{t\in(j,k)}\mu_{4t}\prod_{t\notin[j,k]}\mu_{2t}^2-\sigma_j^2\sigma_k^2\mu_{2j}\mu_{2k}\prod_{t\notin j:k}\mu_{2t}^2\right), \tag{22}$$

where $\eta_j = \mu_{4j} - 2\mu_j\mu_{3j} + \mu_{2j}^2$.

Proof. We use (18) to write covariances in terms of the first few $\boldsymbol{x}_i$. For $1 \leqslant j \leqslant d$ we have $\Delta_{d+j} = \prod_{t=1}^{j-1} g_t(z_{d+t}) \times \big(g_j(z_{d+j}) - g_j(z_d)\big) \times \prod_{t=j+1}^{d} g_t(z_t)$ so that

$$\mathbb{E}(\Delta_{d+j}^2) = 2\sigma_j^2\prod_{t\neq j}\mu_{2t} \quad\text{and}\quad \mathbb{E}(\Delta_{d+j}^4) = (12+2\kappa_j)\sigma_j^4\prod_{t\neq j}\mu_{4t}$$

and $\mathrm{Var}(\Delta_{d+j}^2) = \eta_j\prod_{t\neq j}\mu_{4t} - 4\sigma_j^4\prod_{t\neq j}\mu_{2t}^2$. Then for $1 \leqslant j < k \leqslant d$ and using a convention that empty products are one,

$$\begin{aligned}\mathbb{E}(\Delta_{d+j}^2\Delta_{d+k}^2) &= \prod_{t=1}^{j-1}\mu_{4t}\times\eta_j\times\prod_{t=j+1}^{k-1}\mu_{2t}^2\times\eta_k\times\prod_{t=k+1}^{d}\mu_{4t} \quad\text{and}\\ \mathbb{E}(\Delta_{2d+j}^2\Delta_{d+k}^2) &= \prod_{t=1}^{j-1}\mu_{2t}^2\times\eta_j\times\prod_{t=j+1}^{k-1}\mu_{4t}\times\eta_k\times\prod_{t=k+1}^{d}\mu_{2t}^2.\end{aligned}$$

Therefore,

$$\begin{aligned}\mathrm{Cov}(\Delta_{d+j}^2,\Delta_{d+k}^2) &= \eta_j\eta_k\prod_{t\in(j,k)}\mu_{2t}^2\prod_{t\notin[j,k]}\mu_{4t}-4\sigma_j^2\sigma_k^2\mu_{2j}\mu_{2k}\prod_{t\notin\{j,k\}}\mu_{2t}^2 \quad\text{and}\\ \mathrm{Cov}(\Delta_{2d+j}^2,\Delta_{d+k}^2) &= \eta_j\eta_k\prod_{t\in(j,k)}\mu_{4t}\prod_{t\notin[j,k]}\mu_{2t}^2\prod_{t=1}^{j-1}\mu_{2t}^2-4\sigma_j^2\sigma_k^2\mu_{2j}\mu_{2k}\prod_{t\notin\{j,k\}}\mu_{2t}^2.\end{aligned}$$

Putting these together establishes the theorem. ∎

Acknowledgments. We thank Masayoshi Mase of Hitachi for helpful discussions about variable importance and explainable AI. We also thank the anonymous reviewers for suggestions that have improved our presentation of this paper.

REFERENCES

M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng (2016), *Tensorflow: A system for large-scale machine learning*, in Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16), pp. 265–283.

E. Borgonovo and E. Plischke (2016), *Sensitivity analysis: A review of recent advances*, European J. Oper. Res., 248, pp. 869–887.

F. Campolongo, A. Saltelli, and J. Cariboni (2011), *From screening to quantitative sensitivity analysis: A unified approach*, Comput. Phys. Commun., 182, pp. 978–988.

G. Chastaing, F. Gamboa, and C. Prieur (2012), *Generalized Hoeffding-Sobol decomposition for dependent variables – application to sensitivity analysis*, Electron. J. Stat., 6, pp. 2420–2448.

G. Chastaing, F. Gamboa, and C. Prieur (2015), *Generalized Sobol' sensitivity indices for dependent variables: Numerical methods*, J. Stat. Comput. Simul., 85, pp. 1306–1333.

B. Efron and C. Stein (1981), *The jackknife estimate of variance*, Ann. Statist., 9, pp. 586–596.

G. Glen and K. Isaacs (2012), *Estimating Sobol' sensitivity indices using correlations*, Environ. Model. Softw., 37, pp. 157–166.

J. Hart and P. A. Gremaud (2018), *An approximation theoretic perspective of Sobol' indices with dependent variables*, Int. J. Uncertain. Quantif., 8, pp. 483–493.

W. Hoeffding (1948), *A class of statistics with asymptotically normal distribution*, Ann. Math. Statist., 19, pp. 293–325.

T. Homma and A. Saltelli (1996), *Importance measures in global sensitivity analysis of nonlinear models*, Reliab. Eng. Syst. Safe., 52, pp. 1–17.

G. Hooker (2007), *Generalized functional ANOVA diagnostics for high-dimensional functions of dependent variables*, J. Comput. Graph. Statist., 16, pp. 709–732.

C. R. Hoyt and A. B. Owen (2020), *Mean dimension of ridge functions*, SIAM J. Numer. Anal., 58, pp. 1195–1216, https://doi.org/10.1137/19M127149X.

B. Iooss and P. Lemaître (2015), *A review on global sensitivity analysis methods*, in Uncertainty Management in Simulation-Optimization of Complex Systems, G. Dellino and C. Meloni, eds., Springer, Boston, pp. 101–122.

A. Janon, T. Klein, A. Lagnoux, M. Nodet, and C. Prieur (2014), *Asymptotic normality and efficiency of two Sobol' index estimators*, ESAIM Probab. Stat., 18, pp. 342–364.

M. J. W. Jansen (1999), *Analysis of variance designs for model output*, Comput. Phys. Commun., 117, pp. 35–43.

M. J. W. Jansen, W. A. H. Rossing, and R. A. Daamen (1994), *Monte Carlo estimation of uncertainty contributions from several independent multivariate sources*, in Predictability and Nonlinear Modelling in Natural Sciences and Economics, J. Gasman and G. van Straten, eds., Kluwer Academic Publishers, Norwell, MA, pp. 334–343.

S. Kucherenko, S. Tarantola, and P. Annoni (2012), *Estimation of global sensitivity indices for models with dependent variables*, Comput. Phys. Commun., 183, pp. 937–946.

P. L'Ecuyer and C. Lemieux (2002), *A survey of randomized quasi-Monte Carlo methods*, in Modeling Uncertainty: An Examination of Stochastic Theory, Methods, and Applications, M. Dror, P. L'Ecuyer, and F. Szidarovszki, eds., Kluwer Academic Publishers, Norwell, MA, pp. 419–474.

R. Liu and A. B. Owen (2006), *Estimating mean dimensionality of analysis of variance decompositions*, J. Amer. Statist. Assoc., 101, pp. 712–721.

W. Mauntz (2002), *Global Sensitivity Analysis of General Nonlinear Systems*, Master's thesis, Imperial College, London.

H. Monod, C. Naud, and D. Makowki (2006), *Uncertainty and sensitivity analysis for crop models*, in Working with Dynamic Crop Models: Evaluation, Analysis, Parametrization and Examples, D. Wallach, D. Makowski, and J. W. Jones, eds., Elsevier, pp. 55–99.

A. B. Owen (2003), *The dimension distribution and quadrature test functions*, Statist. Sinica, 13, pp. 1–17.

A. B. Owen and C. Prieur (2017), *On Shapley value for measuring importance of dependent inputs*, SIAM/ASA J. Uncertain. Quantif., 5, pp. 986–1002, https://doi.org/10.1137/16M1097717.

A. B. Owen and S. D. Tribble (2005), *A quasi-Monte Carlo Metropolis algorithm*, Proc. Natl. Acad. Sci. USA, 102, pp. 8844–8849.
A. Puy, W. Becker, S. L. Piano, and A. Saltelli (2020), *The Battle of Total-order Sensitivity Estimators*, Princeton University, Princeton, NJ.
V. K. Rohatgi and G. Székely (1989), *Sharp inequalities between skewness and kurtosis*, Statist. Probab. Lett., 8, pp. 297–299.
A. Saltelli (2002), *Making best use of model evaluations to compute sensitivity indices*, Comput. Phys. Commun., 145, pp. 280–297.
A. Saltelli and I. M. Sobol' (1995), *About the use of rank transformation in sensitivity analysis of model output*, Reliab. Eng. Syst. Safe., 50, pp. 225–239.
A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola (2008), *Global Sensitivity Analysis. The Primer*, John Wiley & Sons, New York.
A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola (2010), *Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index*, Comput. Phys. Commun., 181, pp. 259–270.
R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra (2017), *Grad-CAM: Visual explanations from deep networks via gradient-based localization*, in Proceedings of the IEEE International Conference on Computer Vision, IEEE, pp. 618–626.
I. M. Sobol' (1969), *Multidimensional Quadrature Formulas and Haar Functions*, Nauka, Moscow, (in Russian).
I. M. Sobol' (1990), *On sensitivity estimation for nonlinear mathematical models*, Mat. Model., 2, pp. 112–118 (in Russian).
I. M. Sobol' (1993), *Sensitivity estimates for nonlinear mathematical models*, Math. Modeling Comput. Experiment, 1, pp. 407–414.
E. Song, B. L. Nelson, and J. Staum (2016), *Shapley effects for global sensitivity analysis: Theory and computation*, SIAM/ASA J. Uncertain. Quantif., 4, pp. 1060–1083, https://doi.org/10.1137/15M1048070.
C. J. Stone (1994), *The use of polynomial splines and their tensor products in multivariate function estimation*, Ann. Statist., 22, pp. 118–184.
O. G. Yalcin (2018), *Image Classification in* 10 *Minutes with MNIST Dataset*, https://towardsdatascience.com/imageclassification-in-10-minutes-with-mnist-dataset-54c35b77a38d.
J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson (2015), *Understanding Neural Networks Through Deep Visualization*, preprint, https://arxiv.org/abs/1506.06579.