Insights from Machine Learning for Evaluating Production Function Estimators on Manufacturing Survey Data

José Luis Preciado Arreola[†] and Andrew L. Johnson[*†]

Abstract

Organizations like census bureaus rely on non-exhaustive surveys to estimate industry population-level production functions. In this paper we propose selecting an estimator based on a weighting of its in-sample and predictive performance on actual application datasets. We compare Cobb-Douglas functional assumptions to existing nonparametric shape constrained estimators and a newly proposed estimated presented in this paper. For simulated data, we find that our proposed estimator has the lowest weighted errors. For actual data, specifically the 2010 Chilean Annual National Industrial Survey, a Cobb-Douglas specification describes at least 90% as much variance as the best alternative estimators in practically all cases considered.

JEL Codes: C30, C61

Keywords: Convex Nonparametric Least Squares, Adaptive Partitioning, Multivariate Convex Regression, Nonparametric Stochastic Frontier

[†] Department of Industrial and Systems Engineering, Texas A&M University

* Corresponding Author: Email: ajohnson@tamu.edu. Address: 3131 TAMU, College Station, TX 77843. Phone: 979-458-2356.

1. Introduction

The importance of sample-specificity and prediction error over-optimism for the assessment of production functions becomes immediately evident when estimating a production function using non-exhaustive survey data[1]. These machine learning concepts allow us to investigate the extent to which an estimated production function characterizes both the set of surveyed establishments and the set of un-surveyed establishments for a particular industry.

Classical frontier estimators such as Stochastic Frontier Analysis (SFA, Aigner et al., 1977), and Data Envelopment Analysis (DEA, Banker et al., 1984), as well as more recent developments such as Stochastic DEA (Simar and Zelenyuk, 2008), Convex Nonparametric Least Squares (CNLS, Kuosmanen, 2008), Constraint Weighted Bootstrapping (CWB, Du et al., 2013), and Shape-Constrained Kernel Weighted Least Squares (SCKLS, Yagi et al., 2016) have all use Monte Carlo simulation results taking random samples drawn from a known data generation process (DGP) to justify their use or benefit. The developers of the methods also evaluate estimator performance on the input vectors of the same sample used for estimation[2]. The literature provides independent comparisons between some of these

[1] We will primarily focus on establishment censuses and surveys performed by census bureaus where an establishment is defined as a single physical location where business is conducted or where services or industrial operations are performed. An example is the U.S. annual survey of manufacturers. This survey is conducted annually, except for years ending in 2 and 7 in which a Census is performed, Foster et al. (2008). This survey includes approximately 50000 establishments selected from the census universe of 346000, or approximately a 15% sampling, Fort et al. (2014).

[2] A body of economic literature of a less computational and more aggregate nature is the growth accounting literature, Solow (1957) and Barro and Sala-i-Martin (2004). These methods rely on price information, the cost minimization assumption and parametric forms to deterministically compute the coefficients of a first order approximation of a general production function using observed input cost shares (see for example Syverson, 2011). This literature's main model adequacy check is to compute the R-squared value on the full

methods considering a more ample set of scenarios in a Monte Carlo simulation framework, but still using the same dataset to fit the production function/frontier and test its goodness-of-fit, see Andor and Hesse (2012) as an example. Numerous applied studies to fit production functions and frontiers with real data have been conducted using the aforementioned methods without assessing estimator sample-specificity or even comparing the performance of multiple estimators on the actual application datasets[3]. Finally, the Monte Carlo simulations used to illustrate the performance of existing estimators do not explore the accuracy with which an estimator predicts the noise level of the DGP, which further obscures insights about the performance of a functional estimator.

To address the issues above, we consider a model selection strategy that improves on the current paradigm for selecting a production function model among a pool of models generated by different functional estimators both on simulated and the actual application datasets. Our primary interest is estimating industry population-level production functions from potentially non-exhaustive manufacturing survey data. Accordingly, we focus on estimating the production function for the observed production units and the unobserved production units that we know exist, but not observed in a survey. For example, when we observe a full census, we will disregard out-of-sample performance because we will have a measurement for each point of interest. The framework we apply to both simulated and real data encompasses three elements: estimation of the optimism-corrected in-sample error

observed dataset, again foregoing insights into the adequacy of the estimator in survey-to-full sample data settings.

[3] For DEA-based studies, see the survey paper by Emrouznejad, Parker and G. Tavares (2008). For relevant SFA applications, see Section 2.10 on Greene (1993). Practical studies involving more recent methods include Mekaroonreung and Johnson (2012) and Eskelinen and Kuosmanen (2013). This is also common in the productivity literature, see for example Olley and Pakes (1997), Levinsohn and Petrin (2003), and De Loecker et al. (2015).

(defined in Section 3) for the observed establishment set, use of a learning set-testing set context to estimate the predictive error on the unobserved establishment set (Hastie, Tibshirani and Friedman, 2009 pp. 222), and a finite-sample weighting, which acknowledges the potential existence of only a finite set of establishments, thus weighting the in-sample and predictive errors proportionally to the survey size[4].

For the simulated datasets, we take advantage of the practically infinite data-generating capability in the Monte Carlo context. We estimate the expected optimism-corrected in-sample error and the predictive error by computing mean squared errors for our fitted estimators on previously unobserved testing sets. These important error measures provide estimates of the functional estimator's expected predictive power for a full census of firms, of which some are unobserved. We also compute the performance against the known true DGP in both the in-sample and learning-to-testing set contexts.

For real manufacturing survey datasets, we use different estimators to calculate the expected in-sample and predictive errors because we do not have the ability to generate new, unobserved datasets from the underlying DGP a priori, as in the case of simulated data. For the in-sample error, given the different natures of the considered estimators we cannot simply compute Mallow's - C_p (1973), Akaike Information Criterion (AIC, 1974), or similar optimism penalization, which are specific to linear models. Instead, we employ the parametric bootstrap approach by Efron (2004) to estimate in-sample optimism for potentially non-linear functional estimators. For our predictive error estimation, we estimate production functions using random subsamples of the survey data and assess the predictive

[4] If the researcher is interested in the descriptive ability of the production function at every point of the input space, only the predictive error is of interest. See Appendix E for some results using this alternative assumption.

error of the fit, thus following a cross-validation strategy (see Stone, 1974; Allen, 1974 and Geisser, 1975 for seminal work on cross-validation).

The universal nature of cross-validation is important in comparing production function estimators (Arlot and Celisse, 2010), as they usually arise from different regression paradigms and have different sets of assumptions. Specifically, Hastie, Tibshirani and Friedman (2009, pp. 230) mention that when evaluating nonlinear, adaptive regression techniques[5], it is usually difficult to estimate the effective number of parameters, and cross-validation is one of the few available model selection strategies available. The most commonly used cross-validation methods are non-exhaustive methods, k-fold cross-validation, and repeated learning-testing (RLT) (Breiman et al, 1984). All have significant computational advantages over methods that exhaustively explore all the possible combinations of learning and validation sets, such as leave-one out cross-validation (Stone, 1974, Geisser, 1975), or leave-p-out cross-validation (Shao, 1993). Unlike k-fold cross-validation, the variance of RLT can be controlled by increasing the number of replicates, given any learning set size (Burman, 1989). We do not use RLT to estimate the predictive error of a learning set of the size of our census set, as is normally the objective on cross-validation, but rather to obtain the expected predictive error for a dataset of the size of the survey set itself. We then use our model selection strategy to assess the required survey sizes for obtaining reliable production function estimates for each of the studied industries.

The main contribution of this paper to the published literature is the proposed optimism-corrected model selection method, which allows evaluation of estimator performance in both simulated and actual manufacturing survey data. The proposed method will benefit

[5] The nonparametric estimators we evaluate are in this category.

organizations unable to collect census data on an annual basis. Further, because applications may have characteristics that favor the use of that particular estimator, we propose that relative performance of an estimator on the real application dataset should be the main criterion to follow when choosing a production frontier estimation method for an application. Furthermore, as an additional contribution we propose a functional estimator, Convex Adaptively Partitioned Nonparametric Least Squares (CAP-NLS), which integrates the idea of adaptive partitioning from Convex Adaptive Partitioning (CAP) (Hannah and Dunson, 2013) with the global optimization strategy of the CNLS estimator.

The CNLS estimator is an example of a sieve estimator which is extremely flexible and is optimized to fit the observed data set, White and Wooldridge (1991) and Chen (2007). Alternatively, the adaptive least squares-based CAP developed in the machine-learning literature has demonstrated good predictive performance by integrating model estimation and selection strategies, (and thus resulting in parsimonious functional estimates) as opposed to only optimizing fit on the observed dataset. Specifically, Hannah and Dunson (2013) recognize that the CNLS estimator overfits the observed dataset at the boundaries of the data, thus affecting the quality of prediction for the true underlying function. Other researchers, such as Huang and Szepesvári (2014) and Balázs, György and Szepesvári (2015) build examples in which CNLS estimation results in infinite Mean Squared Error due to overfitting of the sample. Using our proposed model-selection method, we illustrate that this overfitting has detrimental effects when estimating production functions from survey data to infer the true underlying industry population-level production behavior.

The remainder of this paper is organized as follows. Section 2 discusses Convex Adaptively Partitioned Nonparametric Least Squares (CAP-NLS), a method that integrates CAP and CNLS using an adaptive partitioning strategy using the Afriat (1967; 1972) inequalities and global optimization which greatly mitigates the overfitting issues of CNLS. Section 3 describes a Monte Carlo simulation analysis to demonstrate the performance of the proposed estimator for both in-sample and learning set-to-testing set scenarios. Section 4 describes fitting production data for the five industries with the largest sample sizes in the 2010 Chilean Annual National Industrial Survey, compares the proposed method to the performance of other estimation methods, and discusses the results. Section 5 discusses the implications of our research, summarizes the contributions to the production/cost function estimation literature, and suggests future research.

2. Convex Adaptively Partitioned Nonparametric Least Squares

2.1 *Production Function Model*

We define the regression model for our nonparametric estimation procedure as

$$Y = f(\boldsymbol{X}) + \varepsilon \, , \tag{1}$$

where Y represents observed output, $f(\boldsymbol{X})$ denotes the attainable output level, given a certain input mix $\boldsymbol{X} = (X_1, \dots, X_d)'$, d is the dimensionality of the input vector, and ε is a symmetric random term, which we call *noise*, assuming a mean 0. For our estimator, we use the establishment-specific equation (2) to derive our objective function:

$$Y_i = f(X_{1i}, \dots, X_{di}) + \varepsilon_i, \quad i = 1, \dots, n \, . \tag{2}$$

For notational simplicity, we let $f_i = f(X_{1i}, \dots, X_{di})$ and $\boldsymbol{X_i} = X_{1i}, \dots, X_{di}$. We describe the decreasing marginal productivity (concavity) property in terms of $\nabla f(\boldsymbol{X})$, i.e., the gradient of f with respect to X, as

$$f(\boldsymbol{X}_i) \leq + f(\boldsymbol{X}_j) + \nabla f(\boldsymbol{X}_j)^T (\boldsymbol{X}_i - \boldsymbol{X}_j) \; \forall i, j. \tag{3}$$

Given that the constraints in (3) hold, the additional constraint $\nabla f(\boldsymbol{X}_i) > 0\ \forall i$ imposes monotonicity.

2.2 Convex Adaptively Partitioned Nonparametric Least Squares

In this paper, we consider nonparametric approximation of $f(\boldsymbol{X})$ with several piecewise linear estimators. These estimators can consistently describe a general concave function allowing the concavity constraints in (3) to be written as a system of linear inequalities. The first estimator we consider, the CNLS estimator, is a sieve estimator consistent with the functional description in (1)-(3), Kuosmanen (2008). CNLS is also the most flexible piecewise linear estimator we consider because it allows and has the most piecewise linear segments or hyperplanes. There are two limitations, however. The estimator imposes condition (3) by a set of numerous pairwise constraints, which requires significant computational enhancements to be applied on moderate datasets (see Lee et al., 2013 and Mazumder et al., 2015). It also results in a parameter-intensive representation of $f(\boldsymbol{X})$, since it allows for potentially *N* distinct hyperplanes. Thus, the highly detailed sample-specific fit limits the estimator's ability to predict the performance unobserved establishments from the same industry. From an economics perspective, allowing for such a large number of distinct hyperplanes is an issue, because individual establishment observations can specify their own vector of marginal products i.e., they can place zero weight on some set of inputs and exclude them from the analysis of that establishment's production function. This implies that even if the establishment uses the inputs intensively, it can ignore the inputs recorded in the data when evaluating its performance.

Hannah and Dunson (2013) propose CAP, a convex regression method also consistent with (1)-(3). The CAP algorithm partitions the dataset into input-space defined subsets (hereafter, *Basis Regions*) and estimate one hyperplane per basis region. CAP explores proposals for basis regions and greedily selects models with incrementally better fits as the number of hyperplanes increases/decreases/refit. CAP transition from simpler (initially linear) to more detailed models of the concave function and select the model that results in the best tradeoff between model fit and the number of parameters used.

We will now introduction Convex Adaptively Partitioned Nonparametric Least Squares (CAP-NLS) which conbines the advantages of both CNLS and CAP. We let $[i]$ be the index of the basis region to which observation i is assigned for a given input set partitioning proposal and K be the number of basis regions. Then, we approximate concave function $f(\boldsymbol{X})$ at input vector $\boldsymbol{X}_i$ with the estimator

$$\hat{f}_K(\boldsymbol{X}_i) = \beta^*_{0[i]} + \boldsymbol{\beta}^{*T}_{-0[i]}\boldsymbol{X}_i \tag{4}$$

where

$$(\beta^*_{0k}, \boldsymbol{\beta}^*_{-0k})_{k=1}^K = \underset{(\beta_{0k}, \boldsymbol{\beta}_{-0k})_{k=1}^K}{\operatorname{argmin}} \sum_{i=1}^n (\beta_{0[i]} + \boldsymbol{\beta}^T_{-0[i]}\boldsymbol{X}_i - Y_i)^2$$

$$s.t.\ \ \beta_{0[i]} + \boldsymbol{\beta}^T_{-0[i]}\boldsymbol{X}_i \le \beta_{0k} + \boldsymbol{\beta}^T_{-0k}\boldsymbol{X}_i \ \ \forall\, i = 1, \dots N, k = 1, \dots, K$$

$$\boldsymbol{\beta}_{-0k} \ge \boldsymbol{0}\ \forall k = 1, \dots, K,$$

the *kth* basis region is fitted by a hyperplane with parameters $\boldsymbol{\beta}_k = (\beta^*_{0k}, \boldsymbol{\beta}^*_{-0k})$. Note that like CNLS we are optimally fitting a piece-wise linear function with a fixed limit on the number of hyperplanes, K, thus the total number of Afriat inequality constraints is NK as opposed to the $N(N-1)$ constraints implied by (3). Further, note that (4) estimates $f(\boldsymbol{X})$ *conditionally* on an input-space partition. Thus, to obtain an unconditional estimator of $f(\boldsymbol{X})$, we need to explore different input-space partitions as in CAP. Consequently, we nest the solution to problem (4) into an algorithm that proposes partitions resulting in a more parsimonious estimator of $f(\boldsymbol{X})$ than the CNLS solution.

We estimate the function $\hat{f}(\boldsymbol{X})$ by iteratively solving (4) inside of the partitioning proposal strategy. At each iteration, the strategy evaluates KML partition-splitting proposals, where M, a tunable parameter, is the number of random input-space location proposals for a new knot at each iteration, $L = d$ is the number of randomly proposed directions, given the current basis regions and a proposed new knot location, that will define the new dataset partition, and K is the current number of partitions at the current iteration. The full estimation algorithm which nests (4) in the adaptive partitioning strategy is:

Algorithm 1. CAP-NLS Estimator

1. Start with $K = 1$ and fit (4).
2. Consider splitting each of the current K hyperplanes at M random knot locations in L random directions.
3. Fit (4) for each of the (at most) KML partition proposals with at least $n_0/2$ observations on each basis region. If no partition proposal with enough observations exists, stop.
4. Select the proposal that minimizes MSE, save in collection of models and let $K = K + 1$. Return to Step 2.

To ensure model parsimony, we select the smallest model from the collection of models (in terms of K) for which MSE is within a prespecified tolerance of the MSE of the largest Kconsidered available in such collection. [6] Note the tunable parameter n_0 is bounded below by $2(d + 1)$. CAP-NLS has one-to-many hyperplane to observations mapping and requires

[6] The tolerance is set to 1% in all of our examples. Initially, we do not use the Generalized Cross Validation (GCV) score approximation used by Hannah and Dunson (2013), because they assert that GCV's predictive results are only comparable with full cross validation strategies for problems with $n \geq 5000$, which are larger than the datasets we consider in this paper.

at least $2(d+1)$ observations per partition to fit each hyperplane like CAP. This property is the key for superior out of sample performance.

Even though CAP and CAP-NLS use the same partitioning strategy, there are three main differences between. First, CAP-NLS imposes concavity via the Afriat Inequalities rather than a minimum-of-hyperplanes construction. Second, CAP-NLS requires solving a global optimization problem rather than multiple localized optimization problems. As we will observe in Sections 3 and 4, the additional structure results in increased rates of convergence and improved robustness against the local monotonicity violations common in manufacturing survey data. Third, CAP-NLS does not require a refitting step, because it retains the observation-to-basis region correspondence before and after fitting problem (4).

2.3 CAP-NLS as a series of Quadratic Programs

Taking advantage of the linearly-constrained quadratic programming structure of CAP-NLS is essential to achieve computational feasibility. Therefore, we write Problem (4) in the standard form

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\boldsymbol{\beta}^T H\boldsymbol{\beta} + \boldsymbol{\beta}^T g \tag{5}$$

$$s.t.\ \ A\boldsymbol{\beta} \leq 0,\ \ \boldsymbol{\beta} \geq \boldsymbol{l}.$$

Starting with the objective function from (4), we let $\widetilde{\boldsymbol{X}} = (\mathbf{1}, \boldsymbol{X})$ and write

$$\min_{(\beta_{0k},\boldsymbol{\beta}_{-0k})_{k=1}^K} \sum_{i=1}^n \left(\beta_{0[i]} + \boldsymbol{\beta}_{-0[i]}^T \boldsymbol{X}_i - Y_i\right)^2 = \min_{(\boldsymbol{\beta}_k)_{k=1}^K} \sum_{i=1}^n \left(\boldsymbol{\beta}_{[i]}^T \widetilde{\boldsymbol{X}}_i - Y_i\right)^2 = \cdots \tag{6}$$

$$= \min_{(\boldsymbol{\beta}_k)_{k=1}^K} \frac{1}{2}\sum_{i=1}^n \left(\boldsymbol{\beta}_{[i]}^T \widetilde{\boldsymbol{X}}_i\right)^2 - \sum_{i=1}^n \left(\boldsymbol{\beta}_{[i]}^T \widetilde{\boldsymbol{X}}_i Y_i\right),$$

where we have dropped constant $\sum_{i=1}^n Y_i^2$ and multiplied times one half. To write the last expression in (6) in standard form, we first write $\sum_{i=1}^n \left(\boldsymbol{\beta}_{[i]}^T \widetilde{\boldsymbol{X}}_i\right)^2$ using matrix operations. We define observation-to-hyperplane $n(d+1) \times K(d+1)$-dimensional mapping matrix P, with elements $P\big((i-1)*(d+1)+j,([i]-1)*(d+1)+i\big) = \tilde{X}_{ij},\ i = 1, \dots, n,\ j = 1, \dots d+1$ and all other elements equal to zero. Similarly, we define $n \times n(d+1)$-dimensional observation-specific vector product matrix S, with elements $S(i,(i-1)*(d+1)+l) = 1$ for $l = 1,..,3,\ i = 1, \dots, n,\ j = 1, \dots d+1$. Then, we concatenate vectors $(\boldsymbol{\beta}_k)_{k=1}^{K}$ in $K(d+1) \times 1$-dimensional vector $\boldsymbol{\beta}$. It follows that

$$\sum_{i=1}^{n}\big(\boldsymbol{\beta}_{[i]}^{T}\widetilde{\boldsymbol{X}}_i\big)^2 = \boldsymbol{\beta}^T P^T (S^T S) P \boldsymbol{\beta} \quad \text{and} \quad \sum_{i=1}^{n}\big(\boldsymbol{\beta}_{[i]}^{T}\widetilde{\boldsymbol{X}}_i Y_i\big) = \boldsymbol{\beta}^T P^T S^T \boldsymbol{Y}, \tag{7}$$

from which we easily see that $H = P^T(S^T S)P$ and $g = -P^T S^T \boldsymbol{Y}$.

To write in the Afriat Inequality constraints as $nK \times K(d+1)$ - dimensional matrix A, we let elements $A\big(K(i-1)+k, j+(d+1)([i]-1)\big) = \tilde{X}_{ij},\quad i = 1, \dots, n,\ j = 1, \dots d+1,\ k = 1, \dots, K,$ and let all other elements equal zero. Finally, we define $K(d+1)$ – dimensional vector $\boldsymbol{l}$ to have elements $\boldsymbol{l}\big((k-1)(d+1)+1\big) = 0, k = 1, \dots, K,$ and all other elements be equal to negative infinity.

3. Experiments on Simulated Data

We compare four estimators via Monte Carlo simulations: the proposed CAP-NLS estimator, a correctly specified parametric estimator, a monotonically-constrained version of CAP estimator, and the CNLS estimator. Our analysis of simulated data is similar to the comparison of methods in published studies that propose classical frontier production function estimators (see Section 4 for a comparison of estimation methods using application data).

We consider Data Generation Processes (DGP) based on Cobb-Douglas functions and calculate our estimates for the expected *in-sample error* of the production function estimators against the true DGP, $E(Err_{ISf})$, where the expectation is taken against all possible learning sets. We also estimate the following expected quantities: *learning-to-testing set* or *predictiv*e error against the true DGP, (Err_f), *in-sample error* against observed output, $E(Err_{ISy})$, and *predictive error* against observed output $E(Err_y)$ (Hastie, Tibshirani and Friedman, 2009 pp. 228-229. We note that on a real dataset, the estimate of in-sample error against observed output is the most reliable fitting diagnostic when working with a *census* or *full set* of establishments. Conversely, the estimator's estimate compared to an additional sample drawn from the same DGP, which defines the expected *predictive error*, is the primary diagnostic when assessing the fit of a functional estimator obtained from estimation on a learning set relative to a much larger population. Thus, assessing the fit of a functional estimator on a finite census from a non-exhaustive sample requires weighting the two errors by the relative sizes of the sets of observed and unobserved establishments.

Thus, we estimate the expected in-sample error against the true DGP for a learning set of size $nLearn$, $E(Err_{ISf}^{nL})$, by $E\left(\widehat{Err_{ISf}^{nL}}\right) = \overline{MSE_{ISf}^{nL}} = \sum_{v=1}^{V}\sum_{i=1}^{n_L}\left(\hat{f}_{vLi}^{nL} - f_{vLi}\right)^2/nV$ [7] for each functional estimator**,** where $\hat{f}_{vL}^{nL}$ is the production function estimate obtained with the vth learning set and learning set of size nL, f_{vLi} is the ith observation of the vth learning set, n_L is the size of the learning set, and V is the number of different learning sets considered. Analogously, we estimate the expected predictive error against the true DGP for a learning set of size $nLearn$, $E(Err_f^{nL})$, by computing the averaged MSE across the V learning-testing set combinations of the same DGP, $E\left(\widehat{Err_f^{nL}}\right) = \overline{MSE_f^{nL}} = \sum_{v=1}^{V}\sum_{i=1}^{n_T}\left(\hat{f}_{vLi}^{nL} - f_{vTi}\right)^2/nV$,

[7] Note that the estimator "hat" character is over $E\left(Err_{ISf}^{nL}\right)$ rather than Err_{ISf}^{nL}, the in-sample error for the particular learning set fitted with the production function.

where we choose the size of the testing set, n_T = 1000, and f_{vTi} is the ith output observation of the vth testing set. When estimating $E(Err^{nL}_{ISy})$, unlike when estimating $E\left(Err^{nL}_{ISf}\right)$, we need to vary the random component of each observation of each learning set to avoid over-optimism (Hastie, Tibshirani and Friedman, 2009 p. 228). Thus, we generate W different sets of noise terms[8] for each learning set and estimate $E\left(\widehat{Err^{nL}_{ISy}}\right) = \overline{MSE^{nL}_{yIS}} = \sum_{v=1}^{V}\sum_{w=1}^{W}\sum_{i=1}^{n_L}\left(\hat{f}_{vLi} - f(\boldsymbol{x}_{vLi}) + \varepsilon_{wTi}\right)^2/nVW$, where $\boldsymbol{x}_{vLi}$ is the ith input vector of the vth learning set, and ε_{wTi} is the ith residual of the wth testing set. Finally, we compute $E\left(\widehat{Err^{nL}_{y}}\right) = \overline{MSE^{nL}_{y}} = \sum_{v=1}^{V}\sum_{i=1}^{n_T}\left(\hat{f}_{vLi} - Y_{vTi}\right)^2/nV$ to estimate the predictive error against observed outputs, where Y_{vTi} is the ith output observation of the vth testing set.

We consider results for *full-sample* or *census* scenarios, and *learning set-to-full set* with finite full set scenarios. For the *full-sample* scenarios, we report $E\left(\widehat{Err^{n}_{ISf}}\right)$ and $E\left(\widehat{Err^{n}_{ISy}}\right)$. For the *learning set-to-full set* scenarios, we compute an estimator for the *full set* error

$$E(\widehat{Err^{nL}_{FS\cdot}}) = \overline{MSE^{nL}_{FS}} = (n_L/n)\, E(\widehat{Err^{nL}_{IS\cdot}}) + ((n_F - n_L)/n)\, E(\widehat{Err_{\cdot}^{nL}}) \qquad (8)$$

where FS stands for full set, and either f or y replaces the dot operator. Note that $E\left(\widehat{Err^{nL}_{ISy}}\right)$, $E\left(\widehat{Err^{nL}_{y}}\right)$, and $E\left(\widehat{Err^{nL}_{FSy}}\right)$ are also estimators for the noise level σ^2 of the DGP, we can use σ^2 as a benchmark for their estimations. Further, without our corrections for over-optimism, computing an estimator $\hat{\sigma}^2$ will be complicated by the nonparametric nature of the regression methods used to fit the production functions[9]. For our learning-to-testing scenarios, we

---

[8] Computing the in-sample error provides a more realistic estimate of the quality of the production function on a full set than the *learning error* $\overline{MSE^{nL}_{Ly}} = \sum_{v=1}^{V}\sum_{i=1}^{n_L}\left(\hat{f}_{vi} - Y_{vi}\right)^2/nv$, because it averages performance across many possible ε_i residual values for the learning set input vector.

[9] Specifically, if we intend to use the learning set's residual sum of squares, calculation of an estimator $\hat{\sigma}^2$ would require knowledge of the functional estimator's effective number of parameters. However, effective parameters can be difficult to calculate for both nonparametric and sieve estimators.

compare the performance of the three methods on 100 learning-testing set pairs, $V = 100$, using learning datasets of size n_L= 30, 50, 80, 100, 150, 200, 240, and 300. For our *full-sample* scenarios, we consider n_{Learn}= 100, 200, 300. For all scenarios, we consider 30 randomly drawn sets of noise testing vectors, W= 30, to compute the in-sample portion of (8).We also estimate the correctly specified parametric estimator for the DGP. The true parametric form is never known in an application, i.e., we cannot select the correctly specified parametric estimator as an estimator for a practitioner, and our estimation results are best-case benchmarks.

Below, we present our estimates of expected *full set* errors measured against the true DGP, $\overline{MSE_{FSf}^{nL}}$, and expected fraction of unexplained variance on the *full set*, $\overline{MSE_{FSy}^{nL}}/var(Y_{FS})$, respectively. Also, note that the expected full set error is equal to expected In-Sample error for the full-sample scenarios. Due to the extensive nature of our results, we present them in graphical form. Tabular results for all experiments are in Appendix A. We record and report other relevant performance indicators, such as the number of hyperplanes fitted and the estimation time.

3.1 Bivariate input Cobb-Douglas DGP

We consider the DGP $Y_i = X_{i1}^{0.4} X_{i2}^{0.5} + \varepsilon_i$, where $\varepsilon_i \sim N(0, \sigma^2)$ and $\sigma = 0.01, 0.05, 0.1, 0.2, 0.3, 0.4$ for our six noise settings, which we split into low and high-noise settings. We assume $X_{ij} \sim Unif(0.1,1)$ for $j = 1,2$ and $i = 1, \ldots, n_L$. Our first observation from Figures 1a and 1b is that our estimated expected full set error results for all learning-to-testing set scenarios for CNLS exceed the scale of the y-axes (due to very high predictive error values); thus, we only present the full set error values for CNLS for the full set scenarios. The top set of graphs in Figure 1a shows that CAP-NLS has similar to slightly better expected full set error values performance than both CAP and CNLS on full set scenarios, whereas CAP-NLS clearly outperforms both methods on learning-to-testing set scenarios. The bottom panels of Figure 1a show that for these low-noise scenarios, $\sigma = 0.01, 0.05, 0.1$, where values correspond to 0.3%, 6.5% and 22% of the output variance is due to noise, the improvement of CAP-NLS against CAP rarely exceeds 2% of the variance of the full dataset. In other words, an r-squared measurement would differ by less than two percent. We observe that the difference between the correctly specified parametric estimator and CAP-NLS is also within 2% for almost all cases.

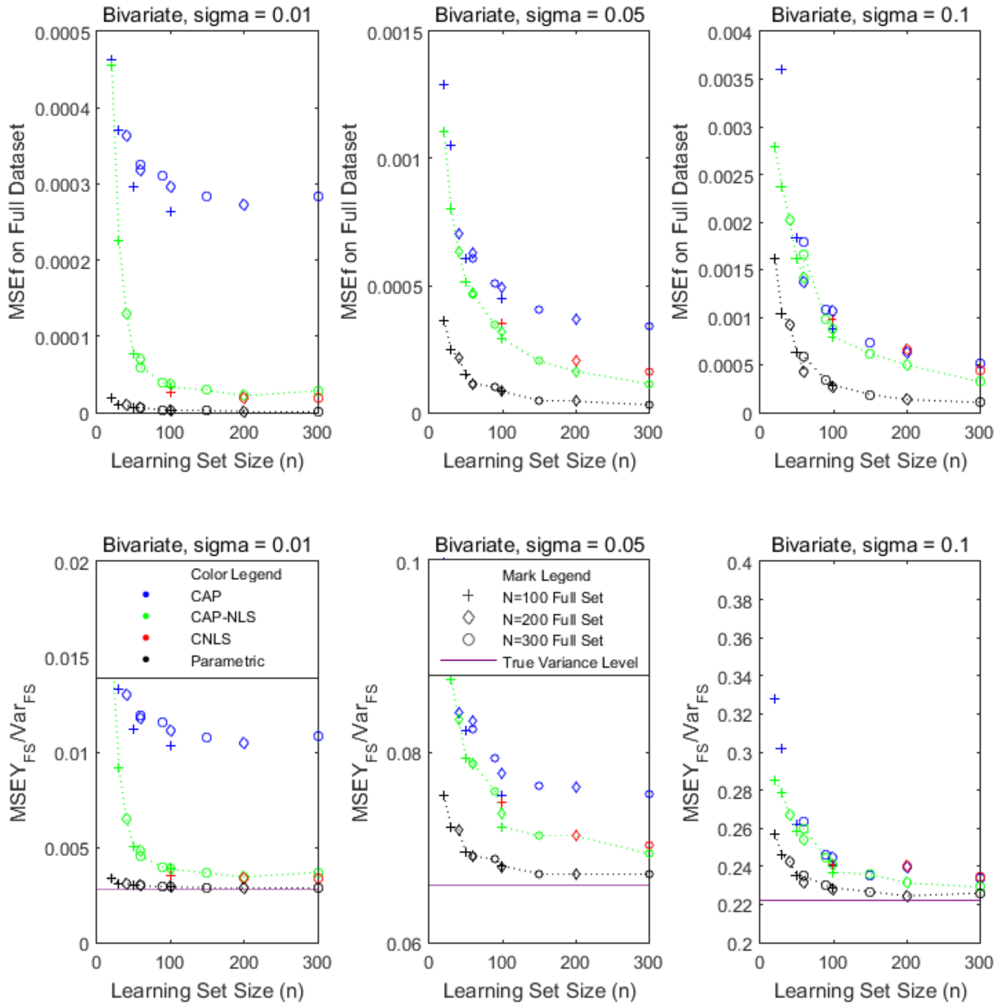

Figure 1a. Bivariate Input Cobb-Douglas DGP results for small noise settings

Surprisingly, Figure 1b shows that for large noise settings, $\sigma = 0.2, 0.3, 0.4$, CAP-NLS and the other nonparametric estimators are competitive with the correctly specified parametric estimator for large sample size or large noise scenarios. This performance gap reduction against the correctly specified parametric estimator is partly due to the generally nonlinear objective function of the Least-Squares estimator of Cobb-Douglas with an additive error term (the correctly specified error structure for our DGP), which can potentially lead to multiple local optimal solutions. If we had instead considered a multiplicative error structure on the DGP for which neither of the nonparametric estimators has a convex programming formulation, the Cobb-Douglas function would have been easy to estimate. Further, only CAP-NLS and the parametric estimator perform consistently regardless of the full set size on these higher noise settings, and the performance gap between CAP-NLS and CAP increases to more significant levels, and CAP's performance becomes unstable. The bottom panels of Figures 1a and 1b show that all estimators approach the true noise-to-total variance level (labeled *True Variance Level* for notational ease) as the learning set size increases, regardless of the noise setting. Finally and as expected, we observe very high correlation between the expected full set error measured against the true DGP and the true noise-to-total variance level.

Table 1 lists the number of hyperplanes fitted for the Full Sample scenarios for the three nonparametric estimators. Larger values indicate a more complex production function using more hyperplanes to characterize the curvature. CNLS fits a much larger number of hyperplanes relative to CAP-NLS, whereas CAP fits functions that are only slightly more complex than linear, by employing two hyperplanes in all estimates. Finally, although CAP-NLS's runtimes are the highest, they are still small in absolute terms.

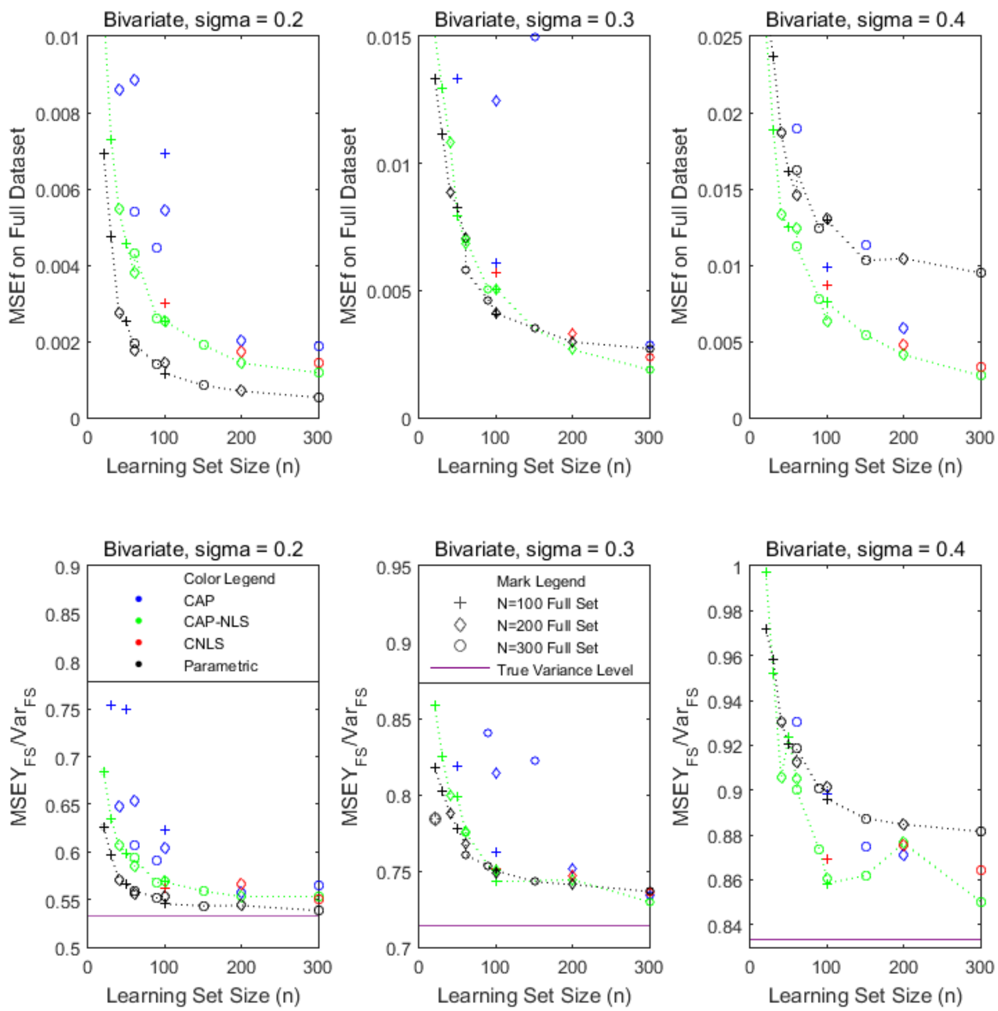

Figure 1b. Bivariate Input Cobb-Douglas DGP results for large noise settings

		CAP-NLS			CAP			CNLS		
σ	n	100	200	300	100	200	300	100	200	300
0.01	K	9	12	12	2	2	2	93	164	242
	Time (s)	4	15	27	1	0.56	0.78	1	8	22
0.05	K	8	10	12	2	2	2	80	135	198
	Time (s)	5	12	30	0.42	0.60	0.77	1	6	23
0.1	K	9	10	11	2	2	2	60	148	172
	Time (s)	4	17	40	0.42	0.54	0.75	1	8	26
0.2	K	9	10	11	2	2	2	54	101	157
	Time (s)	5	16	32	0.47	0.66	0.79	1	8	23
0.3	K	8	10	11	2	2	2	50	98	147
	Time (s)	5	15	28	0.41	0.58	0.71	1	8	22
0.4	K	8	10	11	2	2	2	47	90	135
	Time (s)	4	15	29	0.46	0.62	0.74	1	7	22

Table 1. Number of Hyperplanes and Runtimes for Bivariate Input Cobb-Douglas DGP

3.2 Trivariate input Cobb-Douglas DGP

We consider the DGP $Y_i = X_{i1}^{0.4} X_{i2}^{0.3} X_{i3}^{0.2} + \varepsilon_i$, where $\varepsilon_i \sim N(0, \sigma^2)$, $\sigma = 0.01, 0.05, 0.1, 0.2, 0.3, 0.4$ for our six noise settings having the same small and large noise split as the previous example, and $X_{ij} \sim Unif(0.1,1)$ for $j = 1,2,3$, $i = 1, \ldots, n_L$. Again, CNLS's expected full set errors exceed the displayed range for the learning-to-full set scenarios regardless of noise level, due to the poor predictive error values, which are partly linked to the higher proportion of non-fully dimensional hyperplanes[10] CNLS fits. Compared to Example 3.1, the higher dimensional form of the parametric estimator adds estimation complexity, i.e., the higher errors for the parametric estimator exceed the very small scale of

[10] These are hyperplanes which have zero coefficients on some input dimensions, implying it is possible to obtain output without the zero-coefficient inputs. Olesen and Petersen (1996) were the first to discuss extensively these types of hyperplanes in a DEA context.

most panels in Figure 2a.

Figure 2b, however, shows that the errors given by the parametric estimator are lower than the errors for CNLS in learning-to-full settings. Further, CAP's expected performance deteriorates relative to Example 3.1, and the performance gain obtained by employing CAP-NLS is relevant in an increased number of settings. As in Example 3.1, as the learning set grows, the expected full set errors gap between CAP-NLS and the correctly specified parametric estimator either favors CAP-NLS at every learning set size or becomes more favorable for CAP-NLS as the learning set size increases. Finally, Figure 2b shows that CAP-NLS can accurately recover a production function even when noise composes nearly 85% of the variance, as for the $\sigma = 0.4$ results.

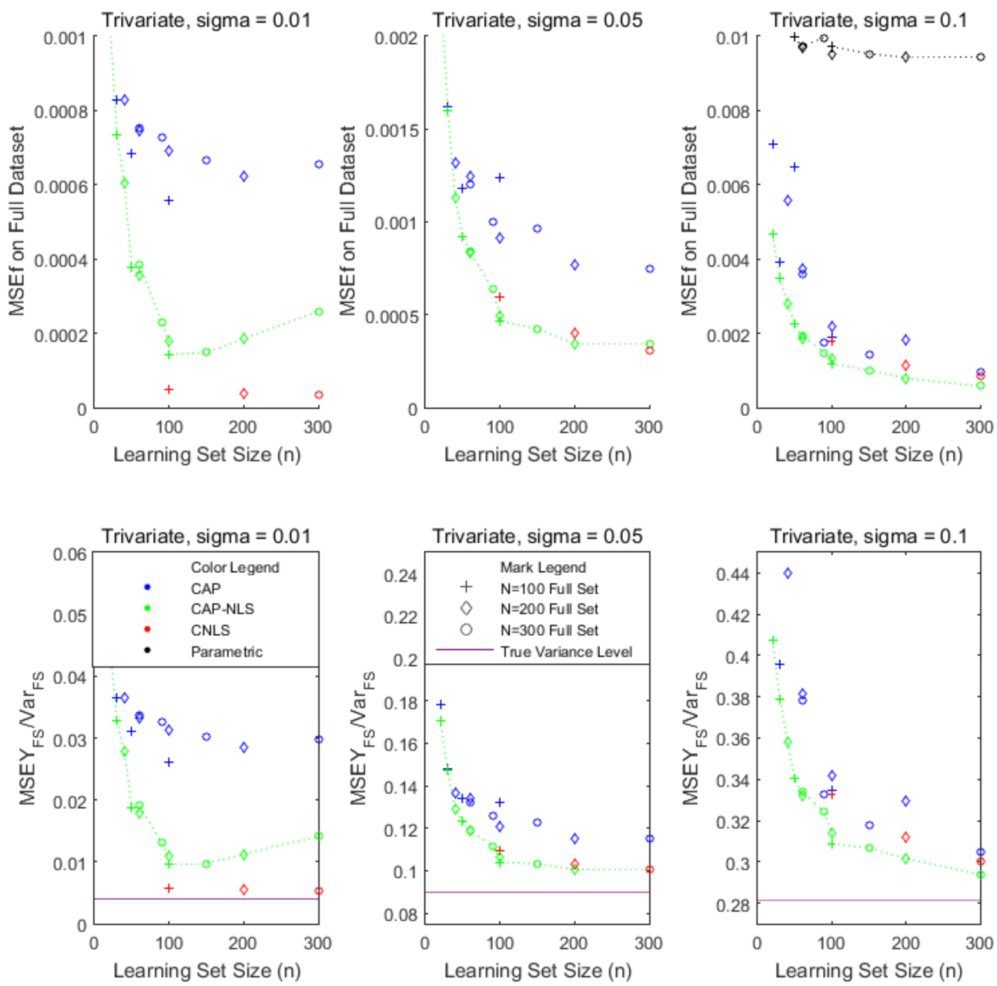

Figure 2a. Trivariate Input Cobb-Douglas DGP results for small noise settings

In Table 2, we observe that all methods fit slightly more hyperplanes than for the Bivariate-input example. The increase in the number of hyperplanes with increased dimensionality is moderate for both CAP-NLS and CAP at all settings. For CNLS, while the number of hyperplanes does not significantly increase for $n = 100$, it significantly increases for the two larger datasets. The runtimes for all methods are also higher than in the previous example, i.e., CAP-NLS's times nearly double, although staying below one minute for all scenarios.

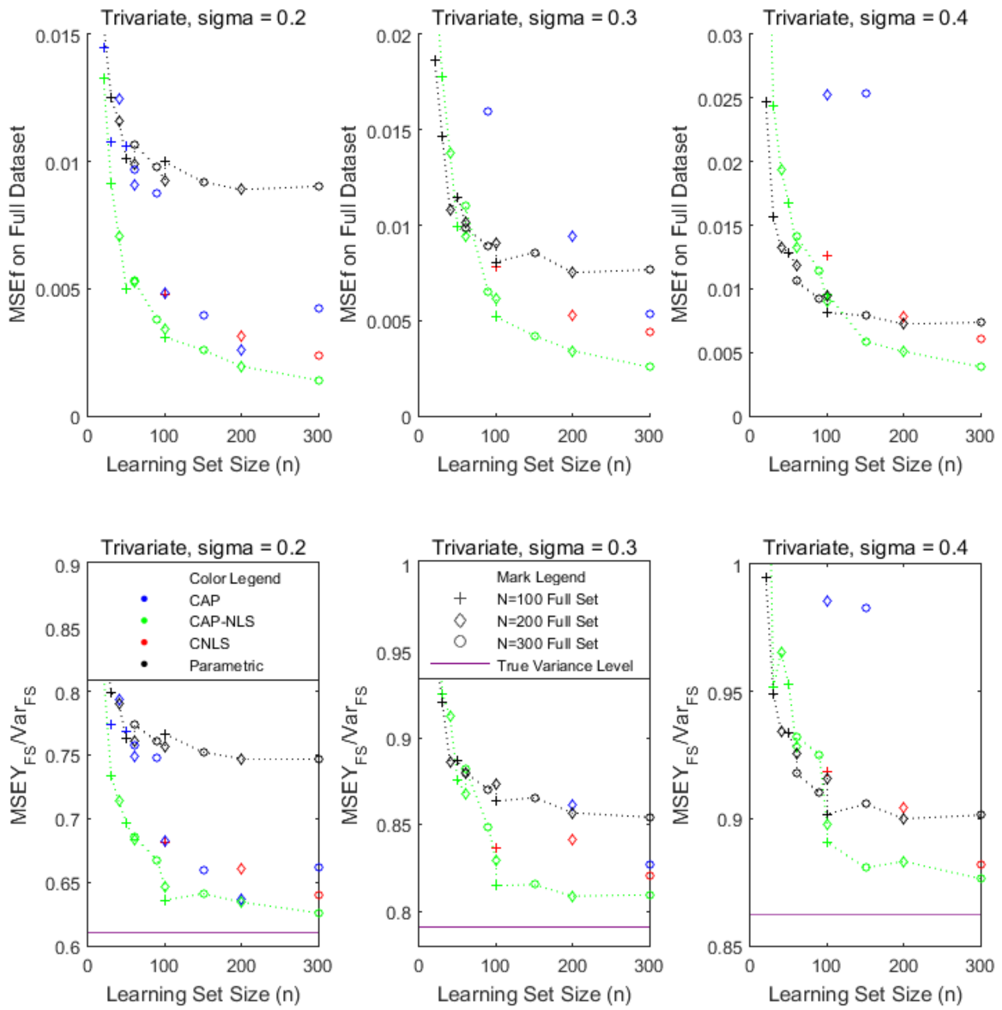

Figure 2b. Trivariate Input Cobb-Douglas DGP results for large noise settings

		CAP-NLS			CAP			CNLS		
σ	n	100	200	300	100	200	300	100	200	300
0.01	K	9	12	13	2	2	2	96	193	294
	Time (s)	5	28	51	0.59	0.78	1	1	9	27
0.05	K	10	11	12	2	2	3	80	169	235
	Time (s)	5	28	45	0.53	0.83	1	1	9	28
0.1	K	8	12	14	2	2	3	75	136	199
	Time (s)	6	24	54	0.54	1	1	1	10	24
0.2	K	8	11	12	2	3	2	61	126	193
	Time (s)	8	23	50	0.52	0.93	1	1	9	28
0.3	K	8	11	12	2	3	3	57	123	184
	Time (s)	5	23	49	0.53	0.92	1	1	9	28
0.4	K	8	11	12	2	3	3	54	115	179
	Time (s)	5	25	49	0.52	0.92	1	1	9	29

Table 2. Number of Hyperplanes and Runtimes for Trivariate Input Cobb-Douglas DGP

3.3 Four-variate input Cobb-Douglas DGP

We consider the DGP $Y_i = X_{i1}^{0.3} X_{i2}^{0.25} X_{i3}^{0.25} X_{i4}^{0.1} + \varepsilon_i$, where $\varepsilon_i \sim N(0, \sigma^2)$ and $\sigma = 0.01, 0.05, 0.1, 0.2, 0.3, 0.4$ for our six noise settings and $X_{ij} \sim Unif(0.1,1)$ for $j = 1,2,3,4$; $i = 1, \dots, n_L$. Figures 3a and 3b show that for this higher dimensional example, the parameters in the parametric estimator are increasingly difficult to learn, and thus the parametric estimator can only predict the true function up to a certain accuracy, namely $\overline{MSE_{FSf}^{nL}} = 0.015$, and then tends to plateau at this error level even as the learning set size increases. Moreover, the benefits of CAP-NLS over the other nonparametric methods are similar to Example 3.2 for the small noise settings, but significantly larger for the large noise settings. Finally, the gap between CAP-NLS and all the other functional estimators, parametric or nonparametric, favors CAP-NLS for all noise settings and learning set sizes.

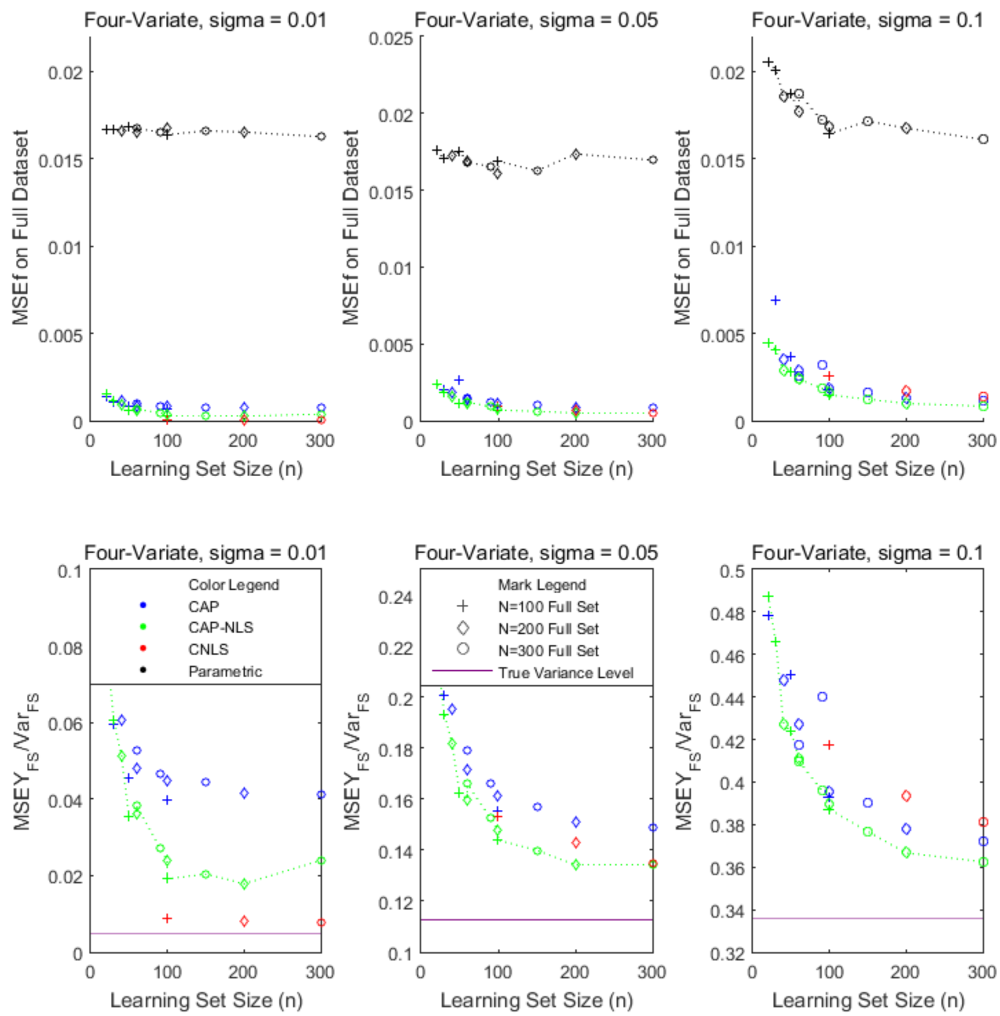

Figure 3a. Four-variate Input Cobb-Douglas DGP results for small noise settings

Table 3 shows that the number of hyperplanes needed to fit the four-variate input production function does not significantly increase from the trivariate-input case of Example 3.2 for any of the methods. CAP-NLS has 40-60% longer runtimes compared to the trivariate-input case. The runtime increase with dimensionality, however, is not a severe concern, because the input information to fit a production function (or output in the case of a cost function) rarely exceeds four variables. The maximum recorded runtime for CAP-NLS is still below two minutes, i.e., it is not large in absolute terms.

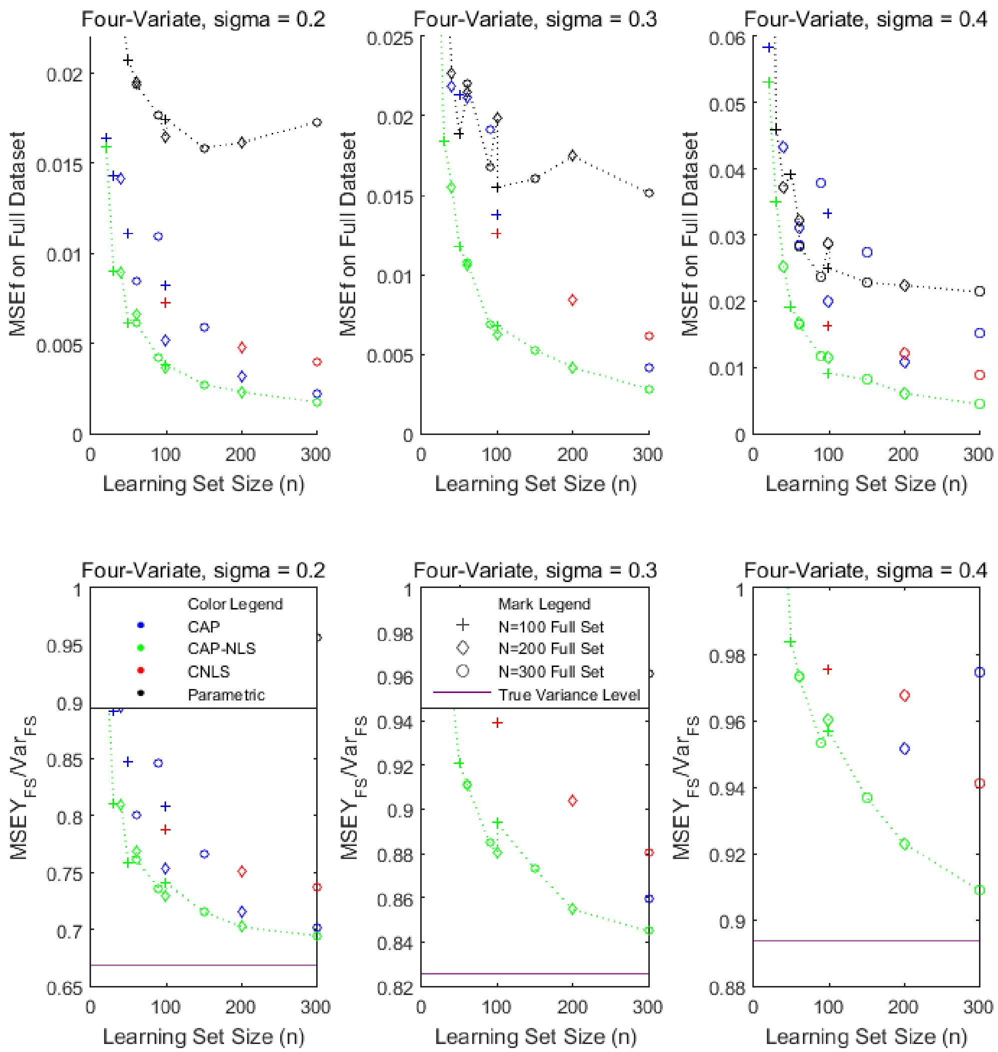

Figure 3b. Four-variate Input Cobb-Douglas DGP results for large noise settings

		CAP-NLS			CAP			CNLS		
σ	*n*	100	200	300	100	200	300	100	200	300
0.01	*K*	7	11	12	2	2	2	98	194	234
	Time (s)	5	29	70	0.42	1	2	1	8	24
0.05	*K*	7	12	13	2	2	2	87	170	215
	Time (s)	4	33	65	0.39	1	1	1	9	27
0.1	*K*	7	12	12	2	2	2	55	166	207
	Time (s)	4	30	62	0.60	1	2	1	10	32
0.2	*K*	7	12	13	2	2	2	63	132	192
	Time (s)	4	36	79	0.45	1	2	1	9	31
0.3	*K*	7	12	12	2	2	2	59	122	192
	Time (s)	4	36	74	0.46	1	2	1	10	32
0.4	*K*	7	12	12	2	2	2	57	122	186
	Time (s)	4	36	75	0.48	1	2	1	10	33

Table 3. Number of Hyperplanes and Runtimes for Four-variate Input Cobb-Douglas DGP

3.4 Implications of Examples 3.1-3.3

This section discusses the implications of the three examples for our proposed estimator. Notably, CAP-NLS is the only functional estimator which performs robustly on a learning-to-full set basis across all dimensionalities and noise levels, while also being the nonparametric estimator with the lowest in-sample error on nearly all of the full set scenarios. Even though CNLS's overfitting of the learning set (as observed by the large number of hyperplanes it fitted) has a severe detrimental effect on the expected full set error due to low predictive power,i.e., high expected predictive error, the overfitting has little effect on the in-sample performance (as observed through its expected full set error in the full set scenarios). Therefore, CNLS is a robust candidate estimator for analyzing full census datasets. CAP performs well on both full set and learning-to-full set scenarios for small noise settings at all dimensionalities, but its learning-to-full set performance deteriorates as the level of noise increases.

Expected full set error on the full set scenarios is similar for all nonparametric methods, with the exception of CAP in the high noise settings with 3 or 4 inputs, when its performance deteriorates. CAP-NLS and CNLS perform similarly in the full set scenarios in all cases. Runtimes for CAP-NLS are the only ones to deteriorate significantly with dimensionality and they are the largest of the three nonparametric methods in all cases. Its runtimes, however, are still small in relative terms, i.e., no larger than 2 minutes for any fitted dataset. Finally, while dimensionality of production functions is typically low and therefore CAP-NLS's scalability in dimensionality is not a concern, it implies that scalability in *n* could be an issue to fit large production datasets[11] (see Appendix B for a modification to CAP-NLS to address this potential issue). The next section presents the dataset.

4. Chilean Annual National Industrial Survey

4.1 Dataset and considerations

The Chilean Annual Industrial Survey (ENIA, by its initials in Spanish) is an annual census of all industrial establishments with 10 or more employees which are located inside the Chilean territory. The census's main goal is to characterize Chile's manufacturing activity in terms of input usage, manufactured products, and means of production utilized in the diverse transformation processes. We focus on the five largest 4-digit industries in terms of sample size and only remove observations with non-positive value added or input values for

[11] Oh et al. (2015) and Crispim Sarmento et al. (2015) discuss the computational challenges for fitting existing nonparametric piecewise linear estimators in large application datasets.

any of the used input variables. In this paper, we refer to the learning sets as the survey subsamples and to the full sets as the survey full sample or census.

Our objective is to illustrate three key points largely overlooked in the production function estimation literature on working with national survey data for manufacturing. First, as real production data is highly clustered around a particular scale size and input ratios, the data lacks the more complex curvature of data simulated from monotonic and concave DGPs. In view of this difference, the performance of estimators and their resulting rankings can vary significantly between Monte Carlo simulation experiments and the estimators' performance on survey data. Therefore, we assess the ability of the estimators discussed in Section 3 to fit industry-specific data from the ENIA dataset on a subsample to full sample setting. Second, we illustrate the replicate-specific performance of the selected functional estimators. Third, we graphically explore the increase in explanatory capability of our fitted production functions as a function of the relative size of survey subsample to survey full sample. The section concludes with a discussion of the implications of practical survey sample sizes.

4.2 Methodology to compare functional estimator performance on real data

We begin by comparing the additive error formulations of CAP-NLS, CAP, and CNLS. We consider the additive-error Cobb-Douglas formulation used in Example 3.3, $Y = X_1^{\alpha_1} X_2^{\alpha_2} X_3^{\alpha_3} X_4^{4} + \varepsilon$, which we label CDA. As theory would imply, we restrict all input powers to be nonnegative for the Cobb-Douglas functional estimator. In total, we compare four different functional estimators. Our comparison focuses on the estimated expected error on the full survey set of establishments, given a survey subset size $E(\widehat{Err}_{FSy}^{nL})$, but reports the scale-invariant quantity $R_{FS}^2 = \max(1 - E(\widehat{Err_{FSy}})/Var(Y_{FS}), 0)$, where $Var(Y_{FS})$ is the sample variance of the output on the full industry dataset[12].

As discussed in Section 3, Monte Carlo simulations, to compute $E\left(\widehat{Err_{FSy}^{nL}}\right)$ we rely on separate estimations of the expected predictive $E\left(Err_y^{nL}\right)$ and in-sample $E\left(Err_{ISy}^{nL}\right)$ errors, which we later weight by the relative size of the observed and unobserved establishment sets. Unlike in section 3, we cannot generate more data from the same DGP as that of the observed dataset, or vectors of residuals with the same level of noise as the DGP, and thus we cannot compute the error estimators $\overline{MSE_y^{nL}}$ and $\overline{MSE_{ISy}^{nL}}$. To circumvent these issues, we estimate $E\left(Err_y^{nL}\right)$ via an RLT procedure and we estimate $E\left(Err_{ISy}^{nL}\right)$ by summing the *learning error* MSE_{yL}^{nL} for a n_L-sized learning set and a parametric bootstrap covariance penalty estimator $\widehat{E(\omega^{nL}})$ for expected *in-sample optimism* $E(\omega^{nL})$ (Efron, 2004). For the RLT procedure, we consider 20%, 30%, 40%, and 50% learning subsets and $V = 100$ replicates to understand the predictive power of subsample-fitted functional estimators when inferring the industry-level production function as the subsample size increases[13]. For the bootstrap procedure, we consider $B = 500$ parametric bootstrap replicates.

We compute our expected predictive error estimate given by RLT, $MSE_{RLT}^{nL} = \sum_{\alpha=1}^{V} \frac{n_L^\alpha}{n} \sum_{i \notin \{\alpha\}} \left(\hat{f}_i^\alpha - Y_i\right)^2 / n_T^\alpha$, where $\{\alpha\}$ is the index set of the α^{th} learning set, $\hat{f}_i^\alpha$ are the estimated functional values obtained from the α^{th} learning set, and $n_T^\alpha = n - n_L^\alpha$, where n_L^α is the size of the α^{th}learning set. Given that we only want to estimate the expected

[12] Note that the definition of R_{FS}^2 implies that if the evaluated estimator fails to explain more variability than the simply taking the mean of the output variable over the full sample, we will instead use the mean as our estimator.

[13] We reemphasize that unlike cross validation procedures in which the goal is to estimate $E\left(Err_y^n\right)$, our goal is to estimate $E\left(Err_y^{nL}\right)$.

predictive error for a set of the size of our *learning set*, our RLT estimator does not have the bias described by Burman (1989) when estimating the usual cross-validation objective, which is the expected predictive error for a set of the size of our *full set*. Burman (1989) shows that the variance of RLT can be partially controlled with the number of replicates V. Finally, we acknowledge that independently of V, the variance of our RLT expected predictive error estimate could increase with the learning set size as the testing set size decreases, given our finite full survey. However, we do not observe an increase in variance in our estimates, as we explained in Section 4.3.

To compute the estimator for expected[14] in-sample error $E\left(\widehat{Err}_{ISy}^{nL}\right)$, we add the learning error MSE_{yL}^{nL} and a covariance penalty term $E(\widehat{\omega}^{nL})$ to account for expected optimism $E(\omega^{nL})$. If we consider an arbitrary estimator $\widehat{Y}_i$ and a uniformly weighted squared loss function, i.e., $g\left(\widehat{Y}_i, Y_i\right) = \widehat{Y}_i - Y_i$ in our notation, Efron (2004) shows that

$$E(\widehat{\omega}^{nL}) = \frac{2}{n_L}\sum_{i=1}^{n_L} cov(\widehat{Y}_i, Y_i). \tag{8}$$

We note that if all of the functional estimators being considered were in the linear smoother form $\widehat{\boldsymbol{Y}} = \mathrm{S}\boldsymbol{Y}$, we would write the penalty term in terms of the trace. Clearly, the Cobb-Douglas functional estimator is not, and so we use the parametric bootstrap algorithm by Efron (2004), which directly estimates $cov(\widehat{Y}_i, Y_i)$ (see Appendix C for the details about this algorithm). Thus, our full expression for $E\left(\widehat{Err}_{FSy}^{nL}\right)$ for learning set sizes of size nL is

$$E\left(\widehat{Err}_{FSy}^{nL}\right) = (n_T/n)MSE_{RLT}^{nL} + (n_L/n)(MSE_{yL}^{nL} + E(\widehat{\omega}^{nL})). \tag{9}$$

[14] Again, expectations and averages over the error and optimism metrics discussed are done over all possible learning sets of a given size.

We assume the data is truly homoscedastic and use an error measure that is uniformly-weighted over observations, such as (10)[15]. Thus, if we intended to use multiplicative or other residual assumptions, our error estimators would need to reflect a similar residual-weighting scheme.

To define the inputs and output for our production function, we follow the KLEMS framework and fit a Value-Added production function

$$VA = Y - M = f(KLES) \ , \tag{10}$$

where *VA* is value added, *Y* is output, *M* is intermediate goods, *K* is capital stock, *L* are labor man-hours, *E* is energy, and *S* is service expenditures, respectively. The variables are readily found in the Chilean manufacturing dataset, except for Energy, for which we also add the fuel expenditures costs. All variables except for L are measured in thousands of Chilean pesos.

4.3. Functional Estimator Comparison Results

In Table 4, the *Best Method* field lists the functional estimator with the highest R^2_{FS} for each subset size, considering ties for functional estimators with R^2_{FS} values within 2% of the best estimator. Table 4 also shows a field for K_{nL}^{CAPNLS}, the average number of CAP-NLS hyperplanes fitted to either the learning sets in the case of 20, 30, 40, and 50 percent subset sizes, or the bootstrapped sets used to compute $E(\widehat{\omega}^{nL})$ in the case of the full set. The average number of CAP-NLS hyperplanes fitted allows us to compare the complexity of the estimated production functions relative to those considered in Section 3. As expected due to the simpler curvature and more concentrated nature of real manufacturing survey data relative to Monte

[15] Although a different assumption can be made within this same framework by introducing a weighting function on the individual residual terms, we focus on the uniformly weighted error.

Carlo simulated data, the number of CAP-NLS hyperplanes fitted for data sets with 100 or 200 observations is generally smaller than those fitted to similar sample sizes in Example 3.3, in which the production function also has a four-dimensional input space.

Further exploring our real data results, we observe both similarities and discrepancies regarding the insights obtained from testing estimators with real data. The clearest similarity to all our low noise settings[16] is the multiple ties across functional estimators in terms of R_{FS}^2, meaning that several of the estimators describe the production function with the same accuracy. Thus, the model selection results are consistent with the small noise setting results for all of our small noise simulated data examples. Discrepancies include better CDA performance for larger datasets, regardless of the residual noise level, which are closer to the insights obtained by using lower-dimensional example 3.1. Surprisingly, CDA's performance is remarkably good, especially if we consider that now the true DGP is unknown. Table 5 presents the capabilities of the CDA parametric estimator against the best estimate achieved for each subset size. In general, the CDA estimator describes nearly as much variance as the best estimator. Further in Appendix D, we include equivalent results to those of Table 5 including estimates from the classical multiplicative error assumption for Cobb-Douglas (labeled CDM) [17]. The results for CDM show that a multiplicative error assumption when fitting the Cobb-Douglas model is a significantly better assumption for the other metal products and wood industries (industry codes 2899 and 2010) (even if tested in terms of (10) with uniform error weighting) and a significantly worse assumption for bakeries

[16] The maximum attainable, i.e., using the full set as the learning set, noise-to-total variance levels of our real datasets are very similar to those of our low noise settings. Compare 1-$\overline{MSE_{FSy}^{nL}}/var(Y_{FS})$ in our low noise settings against the R_{FS}^2results of the 100% survey real datasets.

[17] Recall we use the Cobb-Douglas function with an additive error term is used to maintain consistency of the error structure across estimators.

(1541). These results show that common characteristics of manufacturing survey data, such as a high concentration of establishments around popular scale sizes or the economically efficient input ratios, sparse data on large establishments and simpler curvature, reduce the performance gap between other shape constrained estimators such as CAP or Cobb-Douglas assumption and our proposed estimator.

Industry Name and Code	n	Survey Size	R_{FS}^2	$K_{nL}^{CAP-NLS}$	Best Method
Other Metal Products (2899)	144	20%	50%	1	CAP-NLS, CDA
		30%	60%	2	CAP-NLS, CDA
		40%	64%	2	CAP-NLS, CDA
		50%	72%	3	CAP-NLS
		100%	88%	7	CAP-NLS
Wood (2010)	150	20%	35%	1	CDA
		30%	40%	1	CAP-NLS, CDA
		40%	47%	2	CAP-NLS, CDA
		50%	52%	3	CAP-NLS, CDA
		100%	66%	6	CAP-NLS
Structural Use Metal (2811)	161	20%	77%	1	CAP-NLS, CAP
		30%	82%	2	CAP-NLS
		40%	87%	3	CAP-NLS, CAP
		50%	90%	4	CAP-NLS
		100%	95%	9	CAP-NLS, CAP
Plastics (2520)	249	20%	54%	2	CAP-NLS, CAP, CDA
		30%	57%	3	CDA
		40%	57%	5	CAP-NLS, CAP, CDA
		50%	60%	7	CAP-NLS, CAP, CDA
		100%	64%	11	CAP-NLS, CAP, CDA
Bakeries (1541)	250	20%	72%	3	CAP
		30%	77%	3	CAP
		40%	78%	4	CAP, CDA
		50%	85%	4	CAP
		100%	99%	5	CAP-NLS, CAP, CDA

Table 4. Method comparison across the 5 largest sampled industries from the Chilean Annual National Industrial Survey, 2010.

Industry Name and Code	n	Survey Size	R^2_{FS}	R^2_{CDA}	Ratio vs. Best Method
Other Metal Products (2899)	144	20%	50%	49%	CDA ties for Best Method
		30%	60%	59%	CDA ties for Best Method
		40%	64%	64%	CDA ties for Best Method
		50%	72%	60%	0.83 vs. CAP-NLS
		100%	88%	79%	0.90 vs. CAP-NLS
Wood (2010)	150	20%	35%	35%	CDA ties for Best Method
		30%	40%	40%	CDA ties for Best Method
		40%	47%	47%	CDA ties for Best Method
		50%	52%	51%	CDA ties for Best Method
		100%	66%	62%	0.94 vs. CAP-NLS
Structural Use Metal (2811)	161	20%	77%	69%	0.90 vs. CAP-NLS
		30%	82%	76%	0.93 vs. CAP-NLS
		40%	87%	81%	0.93 vs. CAP-NLS
		50%	90%	87%	0.97 vs. CAP-NLS
		100%	95%	91%	0.96 vs. CAP-NLS
Plastics (2520)	249	20%	54%	53%	CDA ties for Best Method
		30%	57%	57%	CDA ties for Best Method
		40%	57%	57%	CDA ties for Best Method
		50%	60%	60%	CDA ties for Best Method
		100%	64%	64%	CDA ties for Best Method
Bakeries (1541)	250	20%	72%	61%	0.85 vs. CAP
		30%	77%	71%	0.92 vs. CAP
		40%	78%	78%	CDA ties for Best Method
		50%	85%	82%	0.96 vs. CAP
		100%	99%	99%	CDA ties for Best Method

Table 5. Ratio of CDA to Best Model performance

Table 6 shows that the best estimator in the Chilean manufacturing dataset is perhaps more closely related to the learning set size regardless of the residual noise level. CAP-NLS is dominant for very small learning set sizes (less than 50 observations). CAP-NLS, CAP and CDA perform similarly for larger datasets. The additional structure of CAP-NLS relative to CAP seems to lose its benefits as the learning set size increases for our application datasets, which is a much more direct statement than we could make from extrapolating the results across the three small noise settings in Example 3.3. Some insights obtained from evaluating estimators on the actual application dataset are not observed from those on the simulated data. For instance, our simulated data examples show potential problems when fitting the CDA model at high dimensionalities or high noise settings, yet for the application datasets considered, CDA is a reliable production function estimator at learning sets of all considered sizes.

	Learning Set Size			
Times selected as "Best Method"	29 - 50	51 - 80	81 - 149	150+
CAP-NLS	7	5	3	4
CAP	3	3	4	3
CDA	5	2	2	2

Table 6. Most frequently selected Best Method for different sample size ranges.

4.4. Estimator performance measures as a function of subsample size and surveying implications

We apply the results from our framework to make recommendations about the minimal size that a randomly-sampled production survey needs to represent a census. We compute simulation-based confidence intervals on R_{FS}^2 across the replicates of our RLT results. As mentioned, increased testing set variance as the learning set size increases does not seem to be large enough to affect the variance of our estimates across the different learning and testing set sizes considered. Based on Table 4, we label CAP-NLS as the Best Method across the different survey sizes for all industries, except Bakeries, for which CAP is identied as the Best Method. Figure 4 shows the learning subset-specific results for the Best Method in terms of goodness-of-fit, R_{FS}^2, for the industries. We note that the variance of R_{FS}^2 and overall predictive power is significantly enhanced by the inclusion of the in-sample component of the expected full set error. In Appendix E, we further explore the sensitivity of the results shown in Figure 4 to our assumption of a finite population of firms and discuss the consequences of considering an infinite amount of unobserved firms when assessing the predictive capabilities of our estimators, thus only evaluating estimator performance in terms of predictive error.

The mean goodness-of-fit increases in survey subsample size for all industries with different degrees of diminishing returns. The results are of significant practical importance for countries and organizations that do not conduct annual censuses. Although the goodness-of-fit results we obtain are specific to the particular census data sets, under mild assumptions they can still guide survey design in the years following the census. Specifically, to use the data from the census year to inform the sample size needed in the following (non-census) years, requires assuming that both the set of establishments within an industry and the complexity of the production function do not changed significantly over the time period. For example based on the Chilean 2010 census data, if production functions with 75% of the predictive power of a census-fitted production function are desired in 2011, the relative survey sample size needs to be approximately 40%, 45%, <20%, <20%,and 25% for industry codes 2899, 2010, 2811, 2520, and 1541, respectively.

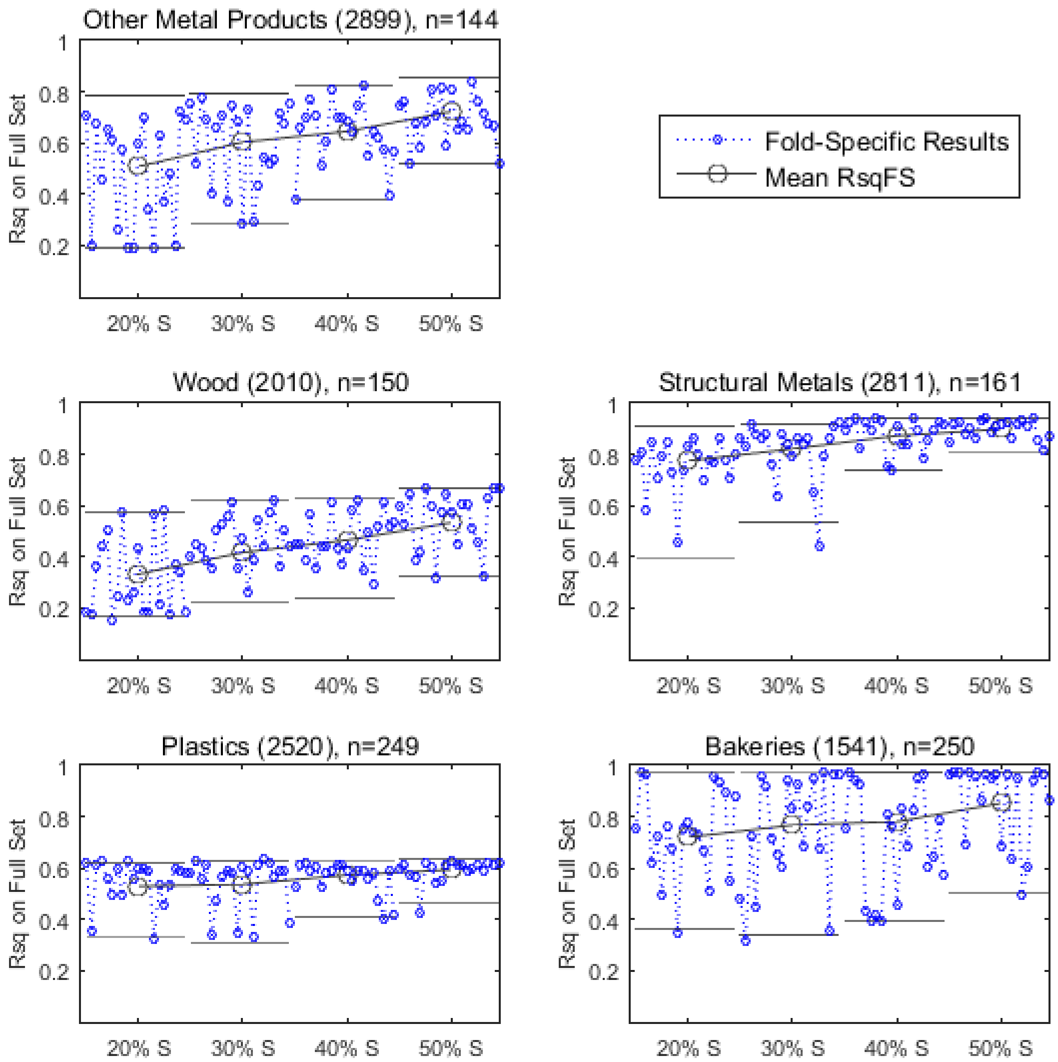

Figure 4. Best Method's fit on the full census, R^2_{Full}, as a function of relative subset size for selected industries. CAP-NLS is the Best Method for industry codes 2899, 2010, 2811, and 2520, whereas CAP is the Best Method for industry code 1541.

5. Conclusions

This paper has two main contributions to the production function estimation literature. Firstly and most importantly, we constructed a framework to test the adequateness of a production function estimator on real data. Specifically, we established a procedure based on repeated learning-testing and parametric bootstrapping that is able to assess the quality of subsample-fitted production functions to fit full survey (census) samples. Further, this procedure estimates the relative quality of the subsample-fitted production function to that of one fitted with a full sample. Using our framework, we demonstrated for our application that unlike for simulated data, CAP-NLS, CAP and a Cobb-Douglas specification performed similarly. Our functional estimator selection procedure is widely applicable, and thus should be routinely used for model selection of econometrically-estimated production functions. Finally, we discovered that the commonly-used Cobb Douglas production function results in very competitive approximations on the Chilean manufacturing dataset at all learning set sizes if an additive residual is used.

Secondly, we introduced CAP-NLS, a nonparametric estimator, which imposes global optimization and no refitting relative to CAP, and additional smoothing relative to CNLS. We formulated a homoscedastic version of CAP-NLS as a series of quadratic programs, which improves computational performance. We demonstrated that CAP-NLS' additional structure relative to CAP and parsimonious structure relative to CNLS translates into superior performance, smaller sensitivity to noise and input vector dimensionality, increased robustness in learning-to-full estimation and a faster empirical rate of convergence on simulated data when the noise level is high relative to the full variance of the output. When the noise level is relatively low to the full variance of the output, CAP-NLS's performance is similar to CAP and better than CNLS.

Our results highlight the need for production function estimators that are reliable on a survey-to-full census basis. In this regard, both "first generation" production function/frontier estimators, such as DEA and SFA, as well as "second generation" generalizations, such as CNLS and CWB were developed ignoring this performance-measurement criterion. Furthermore, our framework demonstrated that methods, which are based on optimization of specific observed datasets, can have important challenges when survey data is used to estimate an industry population-level production function. Thus, we conclude that a new generation of estimators which is able to overcome these challenges is needed. CAP and CAP-NLS, along other smoothed versions of Least Squares-based estimators, such as the estimators in Yagi et al. (2015) and Mazumder et al. (2015), are members of this new generation.

Further work can be done in applying our estimator selection framework to a broader array of datasets, as we have restricted this exposition to the largest industries in the Chilean manufacturing dataset. Theoretical research related to CAP-NLS, such as proving consistency and setting bounds on CAP-NLS' fast rate of convergence remain open. Incorporation of smoothing strategies to CAP-NLS, such as the one presented in Mazumder et al. (2015), also are outstanding future lines of work.

References

Afriat, Sydney N., "The construction of utility functions from expenditure data," *International Economic Review* 8 (1967), 67-77.

Afriat, Sydney N., "Efficiency estimation of production functions," *International Economic Review* 13 (1972), 568-598.

Aigner, Dennis J., C.A. Knox Lovell, and Peter Schmidt, "Formulation and Estimation of Stochastic Frontier Production Function Models," *Journal of Econometrics* 6 (1977), 21–37.

Akaike, Hirotugu,"A new look at the statistical model identification," *IEEE Transactions on Automatic Control* 19 (1974), 716–723.

Allen, David M., "The relationship between variable selection and data augmentation and a method for prediction," *Technometrics* 16 (1974), 125–127.

Andor, Mark and Frederik Hesse, "The StoNED age: the departure into a new era of efficiency analysis? A Monte Carlo comparison of StoNED and the "oldies" (SFA and DEA)," *Journal of Productivity Analysis* 41 (2014), 85-109.

Arlot, Sylvain and Alain Celisse, "A survey of cross-validation procedures for model selection," *Statistics Surveys* 4 (2010), 40—79.

Balázs, Gábor, András György, and Csaba Szepesvári, "Near-optimal max-affine estimators for convex regression," *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics* (2015), 56-64.

Banker, Rajiv D., Abraham Charnes, and William W. Cooper, "Some models for estimating technical and scale inefficiencies in data envelopment analysis," *Management Science* 30 (1984), 1078–1092.

Barro, Robert J., and Xavier Sala-I-Martin, *Economic Growth* (Boston, MA: MIT Press, 2004).

Breiman, Leo, Jerome H. Friedman, Richard Olshen, and Charles Stone, *Classification and regression trees* (Belmont, CA: Wadsworth Advanced Books and Software, 1984).

Breiman, Leo, "The little bootstrap and other methods for dimensionality selection in regression: X-Fixed predictor error*." Journal of the American Statistical Association* 87 (1992), 738-754.

Burman, Prabir, "A comparative study of ordinary cross-validation, *v*-fold cross-validation and the repeated learning-testing methods." *Biometrika* 76 (1989), 503–514.

Chen, Xiaohong, “Large Sample Sieve Estimation of Semi-Nonparametric Models,” in: J.J. Heckman and E.E. Leamer, (eds.), *Handbook of Econometrics,* 6 (North Holland: Elsevier, 2007).

Du, Pang, Christopher F. Parmeter, and Jeffrey S. Racine, “Nonparametric Kernel Regression with Multiple Predictors and Multiple Shape Constraints,” *Statistica Sinica* 23 (2013), 1347-1371.

Efron, Bradley, “The estimation of prediction error: covariance penalties and cross-validation,” *Journal of the American Statistical Association* 99 (2004), 619–642.

Emrouznejad, Ali, Barnett R. Parker, and Gabriel Tavares, “Evaluation of research in efficiency and productivity: a survey and analysis of the first 30 years of scholarly literature in DEA,” *Socio-Economic Planning Sciences* 42 (2008), 151–157.

Fort, Teresa C., John Haltiwanger, Ron Jarmin, and Javier Miranda, “How firms respond to business cycles: The role of firm age and firm size,” *IMF Economic Review* 61 (2013), 520-559.

Foster, Lucia, John Haltiwanger, and Chad Syverson, “Reallocation, firm turnover, and efficiency: Selection on productivity or profitability?” *American Economic Review* 98 (2008), 394-425.

Geisser, Seymour, “The predictive sample reuse method with applications,” *Journal of the American Statistical Association* 70 (1975), 320–328.

Greene, William, “The econometric approach to efficiency analysis” in H.O. Fried, C.A.K. Lovell, S.S. Schmidt (eds.), *The Measurement of Productive Efficiency: Techniques and Applications* (Oxford: Oxford University Press, 1993).

Hannah, Lauren A. and David Dunson, “Multivariate Convex Regression with Adaptive Partitioning,*” Journal of Machine Learning Research* 14 (2013), 3207–3240.

Hastie, Trevor, Robert Tibshirani, and Jerome H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (New York, NY: Springer, 2009).

Huang, Ruitong, and Csaba Szepesvári, "A finite-sample generalization bound for semiparametric regression: Partially linear models," *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics* (2014), 402-410.

Kuosmanen, Timo, "Representation Theorem for Convex Nonparametric Least Squares," *The Econometrics Journal* 11 (2008), 308-325.

Lee, Chia-Yen, Andrew L. Johnson, Erick Moreno-Centeno and Timo Kuosmanen, "A more efficient algorithm for convex nonparametric least squares," *European Journal of Operational Research* 227 (2013), 391-400.

Mallows, Colin L., “Some Comments on C_P,” *Technometrics* 15 (1973), 661–675.

Mazumder, Rahul, Arkopal Choudhury, Garud Iyengar and Bodhisattva Sen, "A Computational Framework for Multivariate Convex Regression and its Variants," arXiv preprint:1509.08165 (2015).

Mekaroonreung, Mathee and Andrew L. Johnson, “Estimating the Shadow Prices of SO2 and NOx for U.S. Coal Power Plants: A Convex Nonparametric Least Squares Approach,” *Energy Economics* 34 (2012), 723-732.

Olesen, Ole B. and Niels C. Petersen, “Indicators of Ill-Conditioned Data Sets and Model Misspecification in Data Envelopment Analysis: An Extended Facet Approach,” *Management Science* 42 (1996), 205-219.

Preciado Arreola, Jose L. and Andrew L. Johnson, "Estimating Stochastic Production Frontiers: A One-stage Multivariate Semi-Nonparametric Bayesian Concave Regression Method," arXiv preprint: 1510.01772 (2015).

Shao, Jun, “Linear model selection by cross-validation,” *Journal of the American Statistical Association* 88 (1993), 486–494.

Simar, Leopold and Valentin Zelenyuk, “Stochastic FDH/DEA estimators for frontier analysis,” *Journal of Productivity Analysis* 36 (2008), 1-20.

Solow, Robert, "Technical change and the aggregate production function," *Review of Economics and Statistics* 39 (1957), 312–320.

Stone, Mervyn, “Cross-validatory choice and assessment of statistical predictions (with discussion and a reply by the authors),” *Journal of the Royal Statistical Society Series B* 36 (1974), 111–147.

Syverson, Chad, "What Determines Productivity?" *Journal of Economic Literature* 49 (2012), 326-365.

White, Halbert, and Jeffrey M. Wooldridge, “Some results on sieve estimation with dependent observations,” in: Barnett, W.A., Powell, J., Tauchen, G. (eds.), *Non-parametric and Semi-parametric Methods in Econometrics and Statistics* (Cambridge: Cambridge University Press, 1991)

Yagi, Daisuke, Andrew L. Johnson. and Timo Kuosmanen, “Shape constrained kernel weighted least squares for the estimation of production functions,” working paper Department of Industrial and Systems Engineering, Texas A&M University (2016).

Appendix A. Results for homoscedastic additive error simulated datasets

d = 2, sigma = 0.2		nFull = 100			nFull = 200			nFull = 300		
nL/nFull	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0025	0.0069	0.0030	0.0014	0.0020	0.0017	0.0012	0.0019	0.0015
100%	MSEFTesting	0.0028	0.0073	292.9882	0.0014	0.0020	93.3874	0.0012	0.0019	168.9078
100%	MSEFCensus	0.0025	0.0069	0.0030	0.0014	0.0020	0.0017	0.0012	0.0019	0.0015
80%	MSEFInSamp	0.0045	0.0160	0.0054	0.0024	0.0053	0.0029	0.0019	0.0273	0.0023
80%	MSEFTesting	0.0047	0.0173	56.2393	0.0026	0.0055	414.9028	0.0019	0.0249	148.8055
80%	MSEFCensus	0.0046	0.0166	28.1224	0.0025	0.0054	207.4528	0.0019	0.0261	74.4039
50%	MSEFInSamp	0.0066	0.0145	0.0081	0.0035	0.0085	0.0041	0.0026	0.0045	0.0031
50%	MSEFTesting	0.0076	0.0169	3.6138	0.0039	0.0090	76.2141	0.0026	0.0044	819.0860
50%	MSEFCensus	0.0073	0.0162	2.5321	0.0038	0.0089	53.3511	0.0026	0.0045	573.3612
30%	MSEFInSamp	0.0093	0.0153	0.0117	0.0051	0.0081	0.0063	0.0041	0.0048	0.0047
30%	MSEFTesting	0.0112	0.0188	3.7562	0.0056	0.0087	50.9327	0.0044	0.0055	85.4171
30%	MSEFCensus	0.0108	0.0181	3.0073	0.0055	0.0086	40.7474	0.0043	0.0054	68.3346
100%	MSEYInSamp/var(Y)	56.84%	62.26%	56.25%	55.30%	55.75%	56.63%	55.32%	56.53%	54.96%
100%	MSEYTest/var(Y)	57.11%	62.96%	3.91E+03	55.36%	56.01%	1.25E+03	55.28%	56.14%	2.25E+03
100%	MSEYCensus/var(Y)	56.84%	62.26%	56.25%	55.30%	55.75%	56.63%	55.32%	56.53%	54.96%
80%	MSEYInSamp/var(Y)	59.90%	73.46%	59.64%	57.04%	60.19%	56.19%	55.80%	87.94%	56.15%
80%	MSEYTest/var(Y)	59.70%	76.45%	7.51E+02	56.83%	60.61%	5.53E+03	55.95%	86.55%	1.98E+03
80%	MSEYCensus/var(Y)	59.80%	74.96%	3.76E+02	56.94%	60.40%	2.77E+03	55.87%	87.25%	9.93E+02
50%	MSEYInSamp/var(Y)	62.70%	73.61%	65.84%	57.40%	64.56%	58.72%	55.75%	58.24%	58.69%
50%	MSEYTest /var(Y)	63.77%	76.16%	4.88E+01	58.98%	65.60%	1.02E+03	57.17%	59.53%	1.09E+04
50%	MSEYCensus/var(Y)	63.45%	75.39%	3.43E+01	58.51%	65.29%	7.12E+02	56.75%	59.14%	7.65E+03
30%	MSEYInSamp/var(Y)	67.83%	77.51%	69.24%	58.21%	62.28%	63.96%	58.19%	59.22%	59.45%
30%	MSEYTest/var(Y)	68.53%	78.59%	5.06E+01	61.23%	65.36%	6.80E+02	59.60%	60.99%	1.14E+03
30%	MSEYCensus/var(Y)	68.39%	78.38%	4.07E+01	60.62%	64.74%	5.44E+02	59.32%	60.63%	9.12E+02
100%	K (Full Census)	8.6	2.22	54.14	10.46	2.12	101.2	10.82	2.06	157.12
100%	Time (Full Census)	4.5916	0.4667	1.1398	15.9820	0.6556	7.6636	32.2248	0.7935	22.9329

Table A1. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=2, sigma = 0.2

d = 2, sigma = 0.3		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0051	0.0061	0.0057	0.0027	0.0030	0.0033	0.0019	0.0029	0.0024
100%	MSEFTesting	0.0056	0.0067	308.9863	0.0028	0.0031	102.7481	0.0019	0.0029	198.8724
100%	MSEFCensus	0.0051	0.0061	0.0057	0.0027	0.0030	0.0033	0.0019	0.0029	0.0024
80%	MSEFInSamp	0.0075	0.0122	0.0084	0.0048	0.0111	0.0057	0.0035	0.0150	0.0042
80%	MSEFTesting	0.0084	0.0144	10.9325	0.0052	0.0139	217.2453	0.0035	0.0150	116.0226
80%	MSEFCensus	0.0079	0.0133	5.4704	0.0050	0.0125	108.6255	0.0035	0.0150	58.0134
50%	MSEFInSamp	0.0118	0.0264	0.0141	0.0067	0.0349	0.0077	0.0048	0.0153	0.0055
50%	MSEFTesting	0.0134	0.0305	10.6080	0.0069	0.0353	32.6064	0.0052	0.0158	186.1256
50%	MSEFCensus	0.0129	0.0293	7.4298	0.0069	0.0352	22.8268	0.0051	0.0156	130.2895
30%	MSEFInSamp	0.0134	0.0215	0.0176	0.0100	0.1021	0.0116	0.0067	0.0410	0.0079
30%	MSEFTesting	0.0155	0.0273	3.4131	0.0110	0.1009	36.0041	0.0071	0.0442	51.8847
30%	MSEFCensus	0.0151	0.0261	2.7340	0.0108	0.1012	28.8056	0.0070	0.0436	41.5094
100%	MSEYInSamp/var(Y)	74.31%	76.22%	75.10%	74.42%	75.12%	74.71%	72.98%	73.45%	73.56%
100%	MSEYTest/var(Y)	75.95%	76.71%	2.45E+03	73.81%	74.00%	8.16E+02	73.35%	74.12%	1.58E+03
100%	MSEYCensus/var(Y)	74.31%	76.22%	75.10%	74.42%	75.12%	74.71%	72.98%	73.45%	73.56%
80%	MSEYInSamp/var(Y)	81.35%	80.78%	79.21%	74.35%	80.49%	75.79%	74.37%	81.15%	75.88%
80%	MSEYTest/var(Y)	78.35%	82.96%	8.76E+01	75.66%	82.41%	1.73E+03	74.34%	83.33%	9.22E+02
80%	MSEYCensus/var(Y)	79.85%	81.87%	4.42E+01	75.01%	81.45%	8.63E+02	74.35%	82.24%	4.61E+02
50%	MSEYInSamp/var(Y)	82.81%	93.18%	84.80%	77.92%	97.76%	74.87%	73.90%	83.22%	77.84%
50%	MSEYTest /var(Y)	82.42%	95.97%	8.50E+01	77.47%	99.92%	2.60E+02	75.97%	84.34%	1.48E+03
50%	MSEYCensus/var(Y)	82.54%	95.13%	5.98E+01	77.60%	99.28%	1.82E+02	75.35%	84.00%	1.04E+03
30%	MSEYInSamp/var(Y)	92.83%	94.09%	83.86%	76.69%	155.28%	83.18%	77.15%	102.55%	76.05%
30%	MSEYTest/var(Y)	84.12%	93.45%	2.78E+01	80.81%	152.02%	2.87E+02	77.63%	106.96%	4.13E+02
30%	MSEYCensus/var(Y)	85.87%	93.58%	2.24E+01	79.99%	152.67%	2.29E+02	77.53%	106.08%	3.30E+02
100%	K (Full Census)	8.84	2.16	49.78	9.84	2.12	94.96	10.7	2.26	143.94
100%	Time (Full Census)	4.6754	0.4517	1.1370	15.0989	0.6220	7.6569	30.8449	0.7676	22.5022

Table A2. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=2, sigma = 0.3

d = 2, sigma = 0.4		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0076	0.0099	0.0087	0.0041	0.0059	0.0048	0.0027	0.1022	0.0033
100%	MSEFTesting	0.0081	0.0105	281.3251	0.0043	0.0064	198.6378	0.0027	0.1093	89.5596
100%	MSEFCensus	0.0076	0.0099	0.0087	0.0041	0.0059	0.0048	0.0027	0.1022	0.0033
80%	MSEFInSamp	0.0124	0.0408	0.0136	0.0062	0.1578	0.0073	0.0054	0.0112	0.0063
80%	MSEFTesting	0.0125	0.0398	10.2707	0.0064	0.1723	118.2487	0.0055	0.0115	198.9407
80%	MSEFCensus	0.0125	0.0403	5.1422	0.0063	0.1650	59.1280	0.0054	0.0113	99.4735
50%	MSEFInSamp	0.0177	0.1723	0.0200	0.0117	0.0647	0.0129	0.0074	0.0720	0.0085
50%	MSEFTesting	0.0193	0.2279	3.2368	0.0127	0.0584	147.9070	0.0080	0.0729	720.6396
50%	MSEFCensus	0.0188	0.2112	2.2717	0.0124	0.0603	103.5388	0.0078	0.0726	504.4503
30%	MSEFInSamp	0.0232	0.0693	0.0265	0.0129	0.0381	0.0146	0.0105	0.0176	0.0118
30%	MSEFTesting	0.0262	0.0731	6.4528	0.0134	0.0435	15.3265	0.0114	0.0193	50.9989
30%	MSEFCensus	0.0256	0.0723	5.1675	0.0133	0.0424	12.2641	0.0112	0.0190	40.8015
100%	MSEYInSamp/var(Y)	85.84%	89.82%	86.93%	87.64%	87.10%	87.54%	85.04%	137.2%	86.46%
100%	MSEYTest/var(Y)	87.57%	88.80%	1.47E+03	85.72%	86.78%	1.04E+03	85.22%	140.6%	4.67E+02
100%	MSEYCensus/var(Y)	85.84%	89.82%	86.93%	87.64%	87.10%	87.54%	85.04%	137.2%	86.46%
80%	MSEYInSamp/var(Y)	94.60%	104.3%	91.38%	85.43%	167.1%	85.79%	86.12%	85.62%	88.40%
80%	MSEYTest/var(Y)	90.15%	104.2%	5.44E+01	86.65%	173.0%	6.17E+02	86.26%	89.31%	1.04E+03
80%	MSEYCensus/var(Y)	92.37%	104.3%	2.77E+01	86.04%	170.1%	3.09E+02	86.19%	87.46%	5.19E+02
50%	MSEYInSamp/var(Y)	98.44%	172.8%	91.64%	90.43%	114.1%	88.98%	85.72%	120.9%	90.70%
50%	MSEYTest /var(Y)	93.82%	202.4%	1.77E+01	90.55%	114.2%	7.71E+02	88.04%	121.2%	3.75E+03
50%	MSEYCensus/var(Y)	95.21%	193.5%	1.27E+01	90.51%	114.2%	5.40E+02	87.34%	121.1%	2.63E+03
30%	MSEYInSamp/var(Y)	108.9%	126.7%	95.09%	88.67%	102.8%	94.60%	90.55%	89.48%	87.85%
30%	MSEYTest/var(Y)	97.4%	121.8%	3.45E+01	91.05%	106.7%	8.07E+01	89.86%	93.94%	2.67E+02
30%	MSEYCensus/var(Y)	99.7%	122.8%	2.78E+01	90.58%	105.9%	6.48E+01	90.00%	93.04%	2.13E+02
100%	K (Full Census)	8.34	2.18	46.24	10.02	2.32	93.9	10.66	2.12	137.24
100%	Time (Full Census)	4.5859	0.4570	1.1464	15.6152	0.6628	7.4554	30.7520	0.7735	23.0779

Table A3. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=2, sigma = 0.4

d = 3, sigma = 0.2		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0031	0.0173	0.0048	0.0019	0.0026	0.0031	0.0014	0.0042	0.0024
100%	MSEFTesting	0.0034	0.0205	591.2154	0.0020	0.0028	1240.0575	0.0014	0.0045	1174.2546
100%	MSEFCensus	0.0031	0.0173	0.0048	0.0019	0.0026	0.0031	0.0014	0.0042	0.0024
80%	MSEFInSamp	0.0047	0.0099	0.0074	0.0033	0.0046	0.0050	0.0025	0.0038	0.0040
80%	MSEFTesting	0.0053	0.0114	53.6331	0.0035	0.0051	357.0193	0.0027	0.0041	907.1555
80%	MSEFCensus	0.0050	0.0106	26.8203	0.0034	0.0048	178.5121	0.0026	0.0039	453.5797
50%	MSEFInSamp	0.0075	0.0088	0.0121	0.0048	0.0083	0.0070	0.0036	0.0083	0.0055
50%	MSEFTesting	0.0098	0.0116	4.6067	0.0055	0.0094	142.5195	0.0039	0.0090	251.8661
50%	MSEFCensus	0.0091	0.0107	3.2283	0.0053	0.0091	99.7658	0.0038	0.0087	176.3079
30%	MSEFInSamp	0.0097	0.0106	0.0136	0.0060	0.0115	0.0089	0.0049	0.0090	0.0072
30%	MSEFTesting	0.0142	0.0154	21.3351	0.0073	0.0127	11.9207	0.0054	0.0099	59.9345
30%	MSEFCensus	0.0133	0.0145	17.0708	0.0070	0.0125	9.5384	0.0053	0.0097	47.9491
100%	MSEYInSamp/var(Y)	63.59%	87.57%	68.17%	63.43%	63.62%	66.10%	62.58%	66.11%	64.01%
100%	MSEYTest/var(Y)	65.53%	91.51%	9.03E+03	63.48%	64.59%	1.89E+04	63.06%	67.74%	1.79E+04
100%	MSEYCensus/var(Y)	63.59%	87.57%	68.17%	63.43%	63.62%	66.10%	62.58%	66.11%	64.01%
80%	MSEYInSamp/var(Y)	70.73%	75.78%	72.11%	63.46%	68.47%	68.10%	63.76%	65.27%	65.78%
80%	MSEYTest/var(Y)	68.51%	77.83%	8.20E+02	65.82%	68.11%	5.45E+03	64.42%	66.61%	1.39E+04
80%	MSEYCensus/var(Y)	69.62%	76.81%	4.10E+02	64.64%	68.29%	2.73E+03	64.09%	65.94%	6.93E+03
50%	MSEYInSamp/var(Y)	68.40%	75.73%	82.11%	66.79%	74.56%	72.78%	66.68%	75.56%	71.31%
50%	MSEYTest /var(Y)	75.48%	78.17%	7.10E+01	69.01%	74.97%	2.18E+03	66.79%	74.47%	3.85E+03
50%	MSEYCensus/var(Y)	73.35%	77.44%	4.99E+01	68.34%	74.85%	1.52E+03	66.76%	74.80%	2.69E+03
30%	MSEYInSamp/var(Y)	78.10%	74.93%	80.60%	70.43%	77.42%	77.07%	67.33%	76.20%	72.97%
30%	MSEYTest/var(Y)	82.29%	84.11%	3.26E+02	71.58%	79.81%	1.83E+02	68.90%	75.67%	9.16E+02
30%	MSEYCensus/var(Y)	81.46%	82.27%	2.61E+02	71.35%	79.33%	1.46E+02	68.59%	75.77%	7.33E+02
100%	K (Full Census)	8.46	2.42	60.88	11.46	2.64	125.88	12.42	2.46	193.14
100%	Time (Full Census)	5.0010	0.5231	1.2491	23.2401	0.9280	8.6596	49.3375	1.1707	28.3374

Table A4. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=3, sigma = 0.2

d = 3, sigma = 0.3		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0052	0.0699	0.0078	0.0034	0.0094	0.0053	0.0026	0.0053	0.0044
100%	MSEFTesting	0.0057	0.0753	782.2253	0.0035	0.0095	717.8571	0.0026	0.0055	1235.0907
100%	MSEFCensus	0.0052	0.0699	0.0078	0.0034	0.0094	0.0053	0.0026	0.0053	0.0044
80%	MSEFInSamp	0.0095	0.0429	0.0144	0.0058	0.0303	0.0088	0.0041	0.1537	0.0064
80%	MSEFTesting	0.0104	0.0491	119.1442	0.0064	0.0312	768.1003	0.0043	0.1663	926.5316
80%	MSEFCensus	0.0099	0.0460	59.5793	0.0061	0.0307	384.0546	0.0042	0.1600	463.2690
50%	MSEFInSamp	0.0145	0.0218	0.0209	0.0090	0.0835	0.0129	0.0063	0.0146	0.0091
50%	MSEFTesting	0.0191	0.0323	56.0073	0.0096	0.0964	70.1889	0.0066	0.0165	271.2729
50%	MSEFCensus	0.0178	0.0291	39.2114	0.0094	0.0925	49.1361	0.0065	0.0160	189.8938
30%	MSEFInSamp	0.0192	0.0240	0.0244	0.0124	0.0195	0.0176	0.0092	0.0619	0.0132
30%	MSEFTesting	0.0246	0.0365	13.3936	0.0142	0.0253	12.1662	0.0114	0.0651	163.4610
30%	MSEFCensus	0.0235	0.0340	10.7197	0.0138	0.0242	9.7365	0.0110	0.0645	130.7715
100%	MSEYInSamp/var(Y)	81.47%	140.1%	83.67%	80.86%	86.12%	84.10%	80.94%	82.65%	82.01%
100%	MSEYTest/var(Y)	83.26%	144.3%	6.87E+03	81.36%	86.59%	6.31E+03	81.08%	83.64%	1.09E+04
100%	MSEYCensus/var(Y)	81.47%	140.1%	83.67%	80.86%	86.12%	84.10%	80.94%	82.65%	82.01%
80%	MSEYInSamp/var(Y)	87.78%	117.4%	88.21%	82.03%	105.0%	84.68%	81.09%	211.6%	82.48%
80%	MSEYTest/var(Y)	87.35%	121.4%	1.05E+03	83.82%	105.4%	6.75E+03	81.96%	224.3%	8.14E+03
80%	MSEYCensus/var(Y)	87.56%	119.4%	5.24E+02	82.93%	105.2%	3.38E+03	81.52%	217.9%	4.07E+03
50%	MSEYInSamp/var(Y)	86.60%	93.3%	102.68%	86.16%	155.6%	90.00%	85.57%	95.36%	87.30%
50%	MSEYTest /var(Y)	95.05%	106.4%	4.93E+02	87.03%	163.2%	6.18E+02	84.56%	93.36%	2.38E+03
50%	MSEYCensus/var(Y)	92.51%	102.5%	3.45E+02	86.77%	160.9%	4.33E+02	84.86%	93.96%	1.67E+03
30%	MSEYInSamp/var(Y)	103.7%	103.6%	94.49%	93.63%	93.34%	95.45%	86.37%	135.5%	90.32%
30%	MSEYTest/var(Y)	99.94%	110.3%	1.19E+02	90.61%	100.5%	1.08E+02	88.68%	135.9%	1.44E+03
30%	MSEYCensus/var(Y)	100.7%	108.9%	9.50E+01	91.22%	99.09%	8.64E+01	88.22%	135.8%	1.15E+03
100%	K (Full Census)	8.32	2.22	56.74	11.12	2.54	122.6	12.16	2.54	184.08
100%	Time (Full Census)	5.0325	0.5227	1.2846	23.2854	0.9178	8.6964	49.1229	1.1300	27.6230

Table A5. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=3, sigma = 0.3

d = 3, sigma = 0.4		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0096	0.3470	0.0126	0.0051	0.0437	0.0078	0.0039	0.0836	0.0061
100%	MSEFTesting	0.0104	0.3180	940.7160	0.0052	0.0429	1025.0415	0.0040	0.0848	1404.5695
100%	MSEFCensus	0.0096	0.3470	0.0126	0.0051	0.0437	0.0078	0.0039	0.0836	0.0061
80%	MSEFInSamp	0.0154	0.0340	0.0195	0.0089	0.0247	0.0138	0.0056	0.0240	0.0086
80%	MSEFTesting	0.0181	0.0397	75.3542	0.0092	0.0258	669.2816	0.0061	0.0267	1050.6757
80%	MSEFCensus	0.0168	0.0369	37.6869	0.0090	0.0252	334.6477	0.0058	0.0253	525.3422
50%	MSEFInSamp	0.0204	0.0294	0.0280	0.0125	0.0322	0.0171	0.0105	0.0478	0.0142
50%	MSEFTesting	0.0261	0.0372	12.2046	0.0136	0.0331	77.9379	0.0118	0.0544	287.5137
50%	MSEFCensus	0.0243	0.0349	8.5516	0.0132	0.0328	54.5617	0.0114	0.0524	201.2638
30%	MSEFInSamp	0.0389	0.0442	0.0405	0.0183	0.0353	0.0226	0.0131	0.0948	0.0168
30%	MSEFTesting	0.0483	0.0601	0.6002	0.0197	0.0399	30.7514	0.0144	0.0928	100.0441
30%	MSEFCensus	0.0464	0.0570	0.4882	0.0194	0.0389	24.6056	0.0142	0.0932	80.0387
100%	MSEYInSamp/var(Y)	89.07%	273.1%	91.86%	88.32%	109.2%	90.42%	87.63%	129.5%	88.18%
100%	MSEYTest/var(Y)	90.90%	256.6%	5.07E+03	88.10%	108.4%	5.52E+03	88.02%	131.6%	7.57E+03
100%	MSEYCensus/var(Y)	89.07%	273.1%	91.86%	88.32%	109.2%	90.42%	87.63%	129.5%	88.18%
80%	MSEYInSamp/var(Y)	95.48%	105.2%	92.29%	89.25%	98.0%	92.15%	87.62%	97.0%	88.14%
80%	MSEYTest/var(Y)	95.07%	106.6%	4.07E+02	90.25%	99.1%	3.61E+03	88.51%	99.5%	5.66E+03
80%	MSEYCensus/var(Y)	95.28%	105.9%	2.04E+02	89.75%	98.5%	1.80E+03	88.06%	98.3%	2.83E+03
50%	MSEYInSamp/var(Y)	85.30%	97.7%	100.18%	92.62%	105.5%	93.88%	93.03%	115.0%	95.24%
50%	MSEYTest /var(Y)	99.39%	105.4%	6.67E+01	92.90%	103.3%	4.21E+02	92.29%	115.3%	1.55E+03
50%	MSEYCensus/var(Y)	95.17%	103.1%	4.70E+01	92.82%	103.9%	2.95E+02	92.51%	115.2%	1.09E+03
30%	MSEYInSamp/var(Y)	110.8%	115.5%	105.7%	99.43%	104.5%	98.59%	93.00%	139.7%	93.92%
30%	MSEYTest/var(Y)	111.5%	118.0%	4.11E+00	95.83%	106.7%	1.67E+02	93.29%	135.4%	5.40E+02
30%	MSEYCensus/var(Y)	111.4%	117.5%	3.50E+00	96.55%	106.3%	1.33E+02	93.23%	136.3%	4.32E+02
100%	K (Full Census)	7.84	2.38	54.3	11.3	2.56	115.12	12.38	2.82	178.82
100%	Time (Full Census)	4.9215	0.5182	1.2876	24.9839	0.9162	8.7400	49.3384	1.1136	28.9099

Table A6. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=3, sigma = 0.4

d = 4, sigma = 0.2		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0038	0.0082	0.0072	0.0023	0.0031	0.0048	0.0017	0.0022	0.0040
100%	MSEFTesting	0.0041	0.0084	366.5847	0.0023	0.0034	1201.9533	0.0018	0.0023	2096.6968
100%	MSEFCensus	0.0038	0.0082	0.0072	0.0023	0.0031	0.0048	0.0017	0.0022	0.0040
80%	MSEFInSamp	0.0056	0.0101	0.0105	0.0036	0.0049	0.0071	0.0026	0.0055	0.0054
80%	MSEFTesting	0.0067	0.0120	13.2270	0.0038	0.0054	569.4433	0.0028	0.0063	1140.8696
80%	MSEFCensus	0.0061	0.0110	6.6187	0.0037	0.0052	284.7252	0.0027	0.0059	570.4375
50%	MSEFInSamp	0.0082	0.0109	0.0154	0.0058	0.0209	0.0105	0.0039	0.0113	0.0083
50%	MSEFTesting	0.0094	0.0157	10.0245	0.0069	0.0237	70.0020	0.0043	0.0107	268.5121
50%	MSEFCensus	0.0090	0.0143	7.0218	0.0066	0.0229	49.0046	0.0042	0.0109	187.9609
30%	MSEFInSamp	0.0115	0.0117	0.0180	0.0073	0.0107	0.0130	0.0057	0.0073	0.0109
30%	MSEFTesting	0.0170	0.0175	0.6706	0.0094	0.0150	3.4263	0.0062	0.0088	43.7317
30%	MSEFCensus	0.0159	0.0164	0.5401	0.0089	0.0141	2.7436	0.0061	0.0085	34.9876
100%	MSEYInSamp/var(Y)	74.08%	80.8%	78.81%	70.22%	71.6%	75.13%	69.41%	70.2%	73.67%
100%	MSEYTest/var(Y)	73.24%	80.5%	6.13E+03	70.48%	72.2%	2.01E+04	69.21%	70.1%	3.51E+04
100%	MSEYCensus/var(Y)	74.08%	80.8%	78.81%	70.22%	71.6%	75.13%	69.41%	70.2%	73.67%
80%	MSEYInSamp/var(Y)	74.18%	83.0%	83.62%	72.95%	75.1%	78.13%	71.70%	76.0%	75.65%
80%	MSEYTest/var(Y)	77.47%	86.4%	2.22E+02	72.82%	75.5%	9.52E+03	71.34%	77.2%	1.91E+04
80%	MSEYCensus/var(Y)	75.82%	84.7%	1.11E+02	72.88%	75.3%	4.76E+03	71.52%	76.6%	9.54E+03
50%	MSEYInSamp/var(Y)	79.00%	80.7%	85.27%	74.82%	100.6%	84.26%	73.76%	85.1%	79.74%
50%	MSEYTest /var(Y)	81.97%	92.7%	1.68E+02	77.66%	105.9%	1.17E+03	73.56%	84.4%	4.49E+03
50%	MSEYCensus/var(Y)	81.08%	89.1%	1.18E+02	76.81%	104.3%	8.20E+02	73.62%	84.6%	3.14E+03
30%	MSEYInSamp/var(Y)	90.9%	81.0%	93.2%	76.33%	81.7%	91.78%	74.66%	76.8%	85.30%
30%	MSEYTest/var(Y)	94.8%	95.7%	1.19E+01	82.06%	91.5%	5.80E+01	76.49%	80.8%	7.32E+02
30%	MSEYCensus/var(Y)	94.0%	92.7%	9.69E+00	80.92%	89.6%	4.66E+01	76.12%	80.0%	5.86E+02
100%	K (Full Census)	7.04	2.02	63.02	11.7	2.16	132.46	12.92	2.08	192.36
100%	Time (Full Census)	4.0210	0.4488	1.3800	35.6645	1.2499	9.3455	78.7762	1.6994	31.3752

Table A7. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=4, sigma = 0.2

d = 4, sigma = 0.3		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0068	0.0138	0.0126	0.0042	0.0267	0.0084	0.0028	0.0042	0.0061
100%	MSEFTesting	0.0074	0.0157	330.2075	0.0043	0.0294	956.4055	0.0028	0.0043	2061.1434
100%	MSEFCensus	0.0068	0.0138	0.0126	0.0042	0.0267	0.0084	0.0028	0.0042	0.0061
80%	MSEFInSamp	0.0113	0.0198	0.0199	0.0063	0.1558	0.0127	0.0051	0.0505	0.0098
80%	MSEFTesting	0.0123	0.0227	6.9686	0.0061	0.1795	231.3875	0.0054	0.0542	846.5812
80%	MSEFCensus	0.0118	0.0213	3.4943	0.0062	0.1677	115.7001	0.0053	0.0523	423.2955
50%	MSEFInSamp	0.0156	0.0245	0.0255	0.0096	0.0188	0.0177	0.0066	0.0188	0.0122
50%	MSEFTesting	0.0196	0.0330	1.7114	0.0110	0.0222	126.0220	0.0070	0.0193	256.3481
50%	MSEFCensus	0.0184	0.0305	1.2056	0.0106	0.0212	88.2207	0.0069	0.0192	179.4474
30%	MSEFInSamp	0.0245	0.0259	0.0350	0.0139	0.0176	0.0216	0.0100	0.0517	0.0178
30%	MSEFTesting	0.0364	0.0400	5.9935	0.0159	0.0229	1.7440	0.0110	0.0516	54.2180
30%	MSEFCensus	0.0340	0.0372	4.8018	0.0155	0.0218	1.3996	0.0108	0.0517	43.3780
100%	MSEYInSamp/var(Y)	89.37%	95.0%	93.88%	85.47%	106.8%	90.39%	84.48%	86.0%	88.02%
100%	MSEYTest/var(Y)	88.84%	96.5%	3.03E+03	86.19%	109.1%	8.78E+03	84.41%	85.9%	1.89E+04
100%	MSEYCensus/var(Y)	89.37%	95.0%	93.88%	85.47%	106.8%	90.39%	84.48%	86.0%	88.02%
80%	MSEYInSamp/var(Y)	91.03%	99.2%	101.05%	88.41%	225.7%	93.76%	87.36%	129.0%	91.44%
80%	MSEYTest/var(Y)	93.10%	102.8%	6.48E+01	87.61%	246.9%	2.12E+03	87.22%	132.1%	7.77E+03
80%	MSEYCensus/var(Y)	92.06%	101.0%	3.29E+01	88.01%	236.3%	1.06E+03	87.29%	130.5%	3.88E+03
50%	MSEYInSamp/var(Y)	94.78%	98.7%	97.20%	89.65%	97.0%	99.59%	88.63%	98.6%	93.33%
50%	MSEYTest /var(Y)	99.77%	112.2%	1.65E+01	91.70%	102.0%	1.16E+03	88.39%	99.8%	2.35E+03
50%	MSEYCensus/var(Y)	98.27%	108.1%	1.19E+01	91.08%	100.5%	8.10E+02	88.46%	99.5%	1.65E+03
30%	MSEYInSamp/var(Y)	111.2%	100.6%	108.9%	91.16%	93.8%	105.01%	89.08%	129.2%	97.61%
30%	MSEYTest/var(Y)	115.3%	118.6%	5.58E+01	96.58%	103.1%	1.68E+01	91.62%	129.1%	4.98E+02
30%	MSEYCensus/var(Y)	114.5%	115.0%	4.49E+01	95.50%	101.3%	1.37E+01	91.11%	129.1%	3.99E+02
100%	K (Full Census)	6.94	2.26	59.42	11.94	2.22	122.16	12.4	2.18	192.14
100%	Time (Full Census)	3.8932	0.4635	1.3946	36.0235	1.2504	9.5493	73.5000	1.6338	31.5927

Table A8. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=4, sigma = 0.3

d = 4, sigma = 0.4		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0091	0.0333	0.0162	0.0061	0.0109	0.0120	0.0044	0.0151	0.0088
100%	MSEFTesting	0.0104	0.0336	269.5879	0.0063	0.0123	1809.4380	0.0044	0.0149	2638.1731
100%	MSEFCensus	0.0091	0.0333	0.0162	0.0061	0.0109	0.0120	0.0044	0.0151	0.0088
80%	MSEFInSamp	0.0183	0.0637	0.0296	0.0110	0.0194	0.0177	0.0078	0.0270	0.0141
80%	MSEFTesting	0.0200	0.0755	14.6548	0.0120	0.0206	378.6360	0.0084	0.0279	671.0586
80%	MSEFCensus	0.0191	0.0696	7.3422	0.0115	0.0200	189.3269	0.0081	0.0274	335.5364
50%	MSEFInSamp	0.0300	0.0387	0.0424	0.0160	0.0289	0.0272	0.0109	0.0369	0.0189
50%	MSEFTesting	0.0371	0.0489	3.0819	0.0170	0.0320	26.7000	0.0119	0.0383	403.4055
50%	MSEFCensus	0.0349	0.0459	2.1701	0.0167	0.0311	18.6981	0.0116	0.0379	282.3895
30%	MSEFInSamp	0.0411	0.0441	0.0526	0.0225	0.0361	0.0367	0.0159	0.0257	0.0274
30%	MSEFTesting	0.0562	0.0620	1.6564	0.0258	0.0450	102.2316	0.0166	0.0291	32.9122
30%	MSEFCensus	0.0532	0.0585	1.3356	0.0251	0.0432	81.7926	0.0164	0.0284	26.3352
100%	MSEYInSamp/var(Y)	95.67%	107.2%	97.55%	92.30%	95.2%	96.74%	90.89%	97.5%	94.13%
100%	MSEYTest/var(Y)	94.65%	107.7%	1.51E+03	92.55%	95.9%	1.01E+04	91.05%	97.0%	1.47E+04
100%	MSEYCensus/var(Y)	95.67%	107.2%	97.55%	92.30%	95.2%	96.74%	90.89%	97.5%	94.13%
80%	MSEYInSamp/var(Y)	97.00%	122.5%	104.83%	96.52%	99.8%	99.05%	93.59%	104.3%	97.10%
80%	MSEYTest/var(Y)	99.77%	130.9%	8.28E+01	95.53%	100.4%	2.12E+03	93.79%	104.7%	3.75E+03
80%	MSEYCensus/var(Y)	98.38%	126.7%	4.19E+01	96.02%	100.1%	1.06E+03	93.69%	104.5%	1.88E+03
50%	MSEYInSamp/var(Y)	104.1%	104.4%	102.41%	96.18%	102.0%	104.83%	95.41%	108.7%	99.18%
50%	MSEYTest /var(Y)	109.4%	116.1%	1.81E+01	97.83%	106.3%	1.50E+02	95.27%	110.2%	2.25E+03
50%	MSEYCensus/var(Y)	107.8%	112.6%	1.30E+01	97.33%	105.0%	1.05E+02	95.32%	109.8%	1.58E+03
30%	MSEYInSamp/var(Y)	118.2%	107.6%	112.0%	98.40%	105.2%	113.07%	96.25%	101.3%	104.57%
30%	MSEYTest/var(Y)	120.1%	123.3%	1.01E+01	103.16%	114.0%	5.72E+02	97.59%	104.7%	1.85E+02
30%	MSEYCensus/var(Y)	119.7%	120.2%	8.34E+00	102.21%	112.2%	4.58E+02	97.32%	104.0%	1.48E+02
100%	K (Full Census)	6.94	2.02	56.82	11.8	2.38	122.26	12.22	2.1	185.84
100%	Time (Full Census)	3.8504	0.4765	1.4262	35.9447	1.2765	9.8326	75.0789	1.6454	32.8098

Table A9. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=4, sigma = 0.4

d = 2, sigma = 0.01		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0000	0.0003	0.0000	0.0000	0.0003	0.0000	0.0000	0.0003	0.0000
100%	MSEFTesting	0.0000	0.0003	87.9100	0.0000	0.0003	76.2429	0.0000	0.0003	55.3655
100%	MSEFCensus	0.0000	0.0003	0.0000	0.0000	0.0003	0.0000	0.0000	0.0003	0.0000
80%	MSEFInSamp	0.0001	0.0002	0.0000	0.0000	0.0003	0.0000	0.0000	0.0003	0.0000
80%	MSEFTesting	0.0001	0.0004	69.4946	0.0000	0.0003	138.3418	0.0000	0.0003	252.9521
80%	MSEFCensus	0.0001	0.0003	34.7473	0.0000	0.0003	69.1709	0.0000	0.0003	126.4761
50%	MSEFInSamp	0.0001	0.0002	0.0000	0.0000	0.0002	0.0000	0.0000	0.0003	0.0000
50%	MSEFTesting	0.0003	0.0004	34.5701	0.0001	0.0004	251.8564	0.0000	0.0003	74.6915
50%	MSEFCensus	0.0002	0.0004	24.1991	0.0001	0.0003	176.2995	0.0000	0.0003	52.2840
30%	MSEFInSamp	0.0002	0.0002	0.0001	0.0001	0.0002	0.0000	0.0000	0.0002	0.0000
30%	MSEFTesting	0.0005	0.0005	1.5433	0.0001	0.0004	38.2043	0.0001	0.0003	132.1729
30%	MSEFCensus	0.0005	0.0005	1.2346	0.0001	0.0004	30.5635	0.0001	0.0003	105.7384
100%	MSEYInSamp/var(Y)	0.32%	0.33%	0.33%	0.31%	0.33%	0.33%	0.30%	0.31%	0.31%
100%	MSEYTest/var(Y)	0.34%	0.34%	3.89E+06	0.31%	0.32%	1.34E+06	0.31%	0.32%	5.68E+05
100%	MSEYCensus/var(Y)	0.32%	0.33%	0.33%	0.31%	0.33%	0.33%	0.30%	0.31%	0.31%
80%	MSEYInSamp/var(Y)	0.37%	0.38%	0.40%	0.33%	0.34%	0.34%	0.31%	0.30%	0.33%
80%	MSEYTest/var(Y)	0.39%	0.41%	4.81E+05	0.34%	0.35%	1.36E+06	0.32%	0.33%	3.06E+06
80%	MSEYCensus/var(Y)	0.38%	0.39%	1.20E+05	0.33%	0.34%	3.40E+05	0.32%	0.32%	7.66E+05
50%	MSEYInSamp/var(Y)	0.44%	0.46%	0.46%	0.35%	0.37%	0.40%	0.33%	0.34%	0.36%
50%	MSEYTest /var(Y)	0.45%	0.55%	8.25E+03	0.38%	0.37%	8.22E+05	0.35%	0.35%	5.04E+05
50%	MSEYCensus/var(Y)	0.44%	0.52%	4.05E+03	0.37%	0.37%	4.03E+05	0.34%	0.35%	2.47E+05
30%	MSEYInSamp/var(Y)	0.44%	0.55%	0.47%	0.37%	9.80%	0.44%	0.36%	0.38%	0.40%
30%	MSEYTest/var(Y)	0.47%	0.63%	4.53E+07	0.42%	9.29%	1.58E+05	0.39%	0.40%	1.08E+06
30%	MSEYCensus/var(Y)	0.47%	0.62%	2.90E+07	0.41%	9.39%	1.01E+05	0.39%	0.40%	6.88E+05
100%	K (Full Census)	9.1200	2.0000	89.6600	11.1000	2.0000	159.2000	12.0000	2.0600	221.3200
100%	Time (Full Census)	5.0034	0.4731	1.0831	16.5378	0.6387	7.0558	32.8468	0.7533	21.8691

Table A10. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=2, sigma = 0.01

d = 2, sigma = 0.05		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0003	0.0004	0.0004	0.0002	0.0004	0.0002	0.0001	0.0003	0.0002
100%	MSEFTesting	0.0003	0.0005	344.2265	0.0002	0.0004	104.3058	0.0001	0.0004	208.4270
100%	MSEFCensus	0.0003	0.0004	0.0004	0.0002	0.0004	0.0002	0.0001	0.0003	0.0002
80%	MSEFInSamp	0.0004	0.0005	0.0005	0.0003	0.0005	0.0004	0.0002	0.0004	0.0002
80%	MSEFTesting	0.0006	0.0007	56.7960	0.0003	0.0005	163.3753	0.0002	0.0004	276.1076
80%	MSEFCensus	0.0005	0.0006	28.3982	0.0003	0.0005	81.6878	0.0002	0.0004	138.0539
50%	MSEFInSamp	0.0006	0.0008	0.0008	0.0004	0.0005	0.0005	0.0003	0.0004	0.0003
50%	MSEFTesting	0.0009	0.0012	22.0658	0.0005	0.0007	237.7376	0.0004	0.0005	212.6012
50%	MSEFCensus	0.0008	0.0011	15.4463	0.0005	0.0006	166.4165	0.0003	0.0005	148.8209
30%	MSEFInSamp	0.0008	0.0009	0.0010	0.0005	0.0006	0.0006	0.0004	0.0005	0.0005
30%	MSEFTesting	0.0012	0.0014	1.5623	0.0007	0.0007	34.0284	0.0005	0.0006	92.0718
30%	MSEFCensus	0.0011	0.0013	1.2501	0.0006	0.0007	27.2228	0.0005	0.0006	73.6575
100%	MSEYInSamp/var(Y)	7.21%	7.54%	7.46%	7.12%	7.63%	7.12%	6.93%	7.55%	7.03%
100%	MSEYTest/var(Y)	7.46%	7.94%	9.08E+03	7.08%	7.63%	2.75E+03	6.94%	7.55%	5.50E+03
100%	MSEYCensus/var(Y)	7.21%	7.54%	7.46%	7.12%	7.63%	7.12%	6.93%	7.55%	7.03%
80%	MSEYInSamp/var(Y)	7.71%	8.09%	7.92%	7.22%	7.60%	7.42%	7.08%	7.59%	7.11%
80%	MSEYTest/var(Y)	8.16%	8.37%	1.50E+03	7.50%	7.94%	4.31E+03	7.16%	7.70%	7.29E+03
80%	MSEYCensus/var(Y)	7.94%	8.23%	7.49E+02	7.36%	7.77%	2.16E+03	7.12%	7.65%	3.64E+03
50%	MSEYInSamp/var(Y)	8.37%	8.57%	8.58%	7.75%	8.19%	7.97%	7.60%	7.71%	7.34%
50%	MSEYTest /var(Y)	8.93%	9.70%	5.82E+02	7.94%	8.38%	6.27E+03	7.59%	8.03%	5.61E+03
50%	MSEYCensus/var(Y)	8.76%	9.36%	4.08E+02	7.88%	8.32%	4.39E+03	7.59%	7.94%	3.93E+03
30%	MSEYInSamp/var(Y)	9.05%	8.80%	9.46%	8.08%	7.84%	8.29%	7.76%	8.08%	7.76%
30%	MSEYTest/var(Y)	9.73%	10.29%	4.13E+01	8.41%	8.56%	8.98E+02	7.91%	8.28%	2.43E+03
30%	MSEYCensus/var(Y)	9.59%	9.99%	3.31E+01	8.34%	8.42%	7.18E+02	7.88%	8.24%	1.94E+03
100%	K (Full Census)	9.1600	2.0000	75.3400	10.7400	2.0600	133.4800	11.4400	2.0400	192.1800
100%	Time (Full Census)	4.9175	0.4587	1.1222	16.5608	0.6480	7.6042	32.8749	0.7590	23.0120

Table A11. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=2, sigma = 0.05

d = 2, sigma = 0.3		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0008	0.0009	0.0010	0.0005	0.0006	0.0007	0.0003	0.0005	0.0004
100%	MSEFTesting	0.0009	0.0010	370.9240	0.0005	0.0007	218.0245	0.0003	0.0005	141.7350
100%	MSEFCensus	0.0008	0.0009	0.0010	0.0005	0.0006	0.0007	0.0003	0.0005	0.0004
80%	MSEFInSamp	0.0015	0.0017	0.0018	0.0008	0.0010	0.0011	0.0006	0.0007	0.0007
80%	MSEFTesting	0.0017	0.0020	130.4580	0.0009	0.0011	219.5168	0.0007	0.0008	329.2754
80%	MSEFCensus	0.0016	0.0018	65.2299	0.0009	0.0011	109.7589	0.0006	0.0007	164.6381
50%	MSEFInSamp	0.0020	0.0029	0.0027	0.0013	0.0013	0.0015	0.0009	0.0010	0.0011
50%	MSEFTesting	0.0025	0.0039	17.0756	0.0015	0.0014	170.5202	0.0010	0.0011	133.5198
50%	MSEFCensus	0.0024	0.0036	11.9537	0.0014	0.0014	119.3646	0.0010	0.0011	93.4642
30%	MSEFInSamp	0.0023	0.0036	0.0032	0.0018	0.0485	0.0022	0.0013	0.0016	0.0016
30%	MSEFTesting	0.0029	0.0049	1266.1997	0.0021	0.0473	74.8704	0.0017	0.0018	195.1175
30%	MSEFCensus	0.0028	0.0047	1012.9604	0.0020	0.0475	59.8968	0.0017	0.0018	156.0943
100%	MSEYInSamp/var(Y)	23.69%	24.05%	24.10%	23.12%	23.92%	24.00%	22.91%	23.38%	23.42%
100%	MSEYTest/var(Y)	24.22%	24.41%	8.24E+03	23.43%	23.69%	4.85E+03	23.11%	23.47%	3.15E+03
100%	MSEYCensus/var(Y)	23.69%	24.05%	24.10%	23.12%	23.92%	24.00%	22.91%	23.38%	23.42%
80%	MSEYInSamp/var(Y)	25.59%	25.72%	26.51%	23.90%	24.30%	24.29%	23.44%	23.09%	23.85%
80%	MSEYTest/var(Y)	26.09%	26.65%	2.90E+03	24.28%	24.66%	4.88E+03	23.71%	23.89%	7.32E+03
80%	MSEYCensus/var(Y)	25.84%	26.19%	1.45E+03	24.09%	24.48%	2.44E+03	23.58%	23.49%	3.66E+03
50%	MSEYInSamp/var(Y)	27.58%	28.44%	28.24%	24.69%	25.37%	26.31%	24.04%	24.21%	25.23%
50%	MSEYTest /var(Y)	27.97%	30.96%	3.80E+02	25.65%	25.46%	3.79E+03	24.62%	24.80%	2.97E+03
50%	MSEYCensus/var(Y)	27.85%	30.20%	2.66E+02	25.36%	25.43%	2.65E+03	24.44%	24.62%	2.08E+03
30%	MSEYInSamp/var(Y)	27.61%	30.93%	28.78%	25.47%	130.87%	27.62%	24.95%	25.93%	26.40%
30%	MSEYTest/var(Y)	28.78%	33.28%	2.81E+04	27.01%	127.47%	1.66E+03	26.26%	26.41%	4.34E+03
30%	MSEYCensus/var(Y)	28.54%	32.81%	2.25E+04	26.70%	128.15%	1.33E+03	26.00%	26.31%	3.47E+03
100%	K (Full Census)	8.9200	2.1000	61.9200	10.6600	2.1200	120.6400	11.1800	2.1000	169.9800
100%	Time (Full Census)	4.8844	0.4632	1.1780	16.5917	0.6587	7.6625	33.8377	0.7826	23.3258

Table A12. $MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}$, time, and K results for d=2, sigma = 0.1

d = 3, sigma = 0.2		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0001	0.0006	0.0001	0.0002	0.0006	0.0000	0.0003	0.0007	0.0000
100%	MSEFTesting	0.0002	0.0008	413.5722	0.0002	0.0007	423.1706	0.0003	0.0007	714.7379
100%	MSEFCensus	0.0001	0.0006	0.0001	0.0002	0.0006	0.0000	0.0003	0.0007	0.0000
80%	MSEFInSamp	0.0003	0.0005	0.0001	0.0001	0.0006	0.0001	0.0001	0.0006	0.0000
80%	MSEFTesting	0.0005	0.0009	32.9737	0.0002	0.0008	187.1054	0.0002	0.0007	429.8912
80%	MSEFCensus	0.0004	0.0007	16.4869	0.0002	0.0007	93.5527	0.0001	0.0007	214.9456
50%	MSEFInSamp	0.0004	0.0004	0.0001	0.0002	0.0005	0.0001	0.0002	0.0006	0.0001
50%	MSEFTesting	0.0009	0.0010	4.2185	0.0004	0.0008	79.8074	0.0003	0.0008	325.6861
50%	MSEFCensus	0.0007	0.0008	2.9529	0.0004	0.0007	55.8652	0.0002	0.0007	227.9803
30%	MSEFInSamp	0.0004	0.0004	0.0001	0.0003	0.0005	0.0001	0.0002	0.0005	0.0001
30%	MSEFTesting	0.0012	0.0012	0.3865	0.0007	0.0009	20.5994	0.0004	0.0008	80.4559
30%	MSEFCensus	0.0011	0.0011	0.3092	0.0006	0.0008	16.4795	0.0004	0.0008	64.3647
100%	MSEYInSamp/var(Y)	0.95%	2.61%	0.58%	1.12%	2.86%	0.55%	1.41%	2.99%	0.53%
100%	MSEYTest/var(Y)	1.29%	3.46%	1.63E+04	1.30%	3.26%	1.67E+04	1.55%	3.22%	2.83E+04
100%	MSEYCensus/var(Y)	0.95%	2.61%	0.58%	1.12%	2.86%	0.55%	1.41%	2.99%	0.53%
80%	MSEYInSamp/var(Y)	1.44%	2.39%	0.68%	0.94%	2.74%	0.59%	0.87%	2.73%	0.57%
80%	MSEYTest/var(Y)	2.31%	3.81%	1.30E+03	1.24%	3.51%	7.40E+03	1.07%	3.30%	1.70E+04
80%	MSEYCensus/var(Y)	1.88%	3.10%	6.52E+02	1.09%	3.12%	3.70E+03	0.97%	3.02%	8.50E+03
50%	MSEYInSamp/var(Y)	1.88%	2.03%	0.69%	1.33%	2.54%	0.65%	0.99%	2.70%	0.61%
50%	MSEYTest /var(Y)	3.88%	4.35%	1.67E+02	1.99%	3.67%	3.15E+03	1.44%	3.50%	1.29E+04
50%	MSEYCensus/var(Y)	3.28%	3.65%	1.17E+02	1.79%	3.33%	2.21E+03	1.31%	3.26%	9.01E+03
30%	MSEYInSamp/var(Y)	1.99%	2.02%	0.73%	1.70%	2.27%	0.69%	1.30%	2.42%	0.66%
30%	MSEYTest/var(Y)	5.22%	5.27%	1.53E+01	3.06%	4.00%	8.14E+02	2.07%	3.60%	3.18E+03
30%	MSEYCensus/var(Y)	4.57%	4.62%	1.22E+01	2.79%	3.65%	6.51E+02	1.91%	3.37%	2.54E+03
100%	K (Full Census)	8.5200	2.0000	94.7200	11.7800	2.1600	185.3600	12.5600	2.1600	272.9800
100%	Time (Full Census)	5.4161	0.5377	1.1190	25.9682	0.8869	7.8553	53.5792	1.1314	26.0566

Table A13. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=3, sigma = 0.01

d = 3, sigma = 0.3		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0005	0.0012	0.0006	0.0003	0.0008	0.0004	0.0003	0.0007	0.0003
100%	MSEFTesting	0.0006	0.0014	762.8861	0.0004	0.0008	975.1576	0.0004	0.0008	823.8718
100%	MSEFCensus	0.0005	0.0012	0.0006	0.0003	0.0008	0.0004	0.0003	0.0007	0.0003
80%	MSEFInSamp	0.0008	0.0010	0.0009	0.0004	0.0008	0.0006	0.0004	0.0009	0.0005
80%	MSEFTesting	0.0011	0.0013	144.0639	0.0005	0.0010	601.6849	0.0004	0.0010	608.9285
80%	MSEFCensus	0.0009	0.0012	72.0324	0.0005	0.0009	300.8427	0.0004	0.0010	304.4645
50%	MSEFInSamp	0.0011	0.0011	0.0013	0.0007	0.0010	0.0009	0.0005	0.0009	0.0007
50%	MSEFTesting	0.0018	0.0018	12.5961	0.0009	0.0013	122.5333	0.0007	0.0011	627.3650
50%	MSEFCensus	0.0016	0.0016	8.8177	0.0008	0.0012	85.7736	0.0006	0.0010	439.1557
30%	MSEFInSamp	0.0013	0.0014	0.0015	0.0008	0.0009	0.0010	0.0006	0.0009	0.0008
30%	MSEFTesting	0.0025	0.0027	4.1813	0.0012	0.0014	68.7782	0.0009	0.0013	236.6423
30%	MSEFCensus	0.0023	0.0024	3.3453	0.0011	0.0013	55.0228	0.0008	0.0012	189.3140
100%	MSEYInSamp/var(Y)	10.41%	13.20%	10.94%	10.06%	11.54%	10.32%	10.08%	11.49%	10.08%
100%	MSEYTest/var(Y)	10.86%	13.76%	2.73E+04	10.23%	11.87%	3.50E+04	10.32%	11.77%	2.95E+04
100%	MSEYCensus/var(Y)	10.41%	13.20%	10.94%	10.06%	11.54%	10.32%	10.08%	11.49%	10.08%
80%	MSEYInSamp/var(Y)	12.04%	13.18%	12.23%	10.41%	11.68%	10.92%	10.19%	12.07%	10.60%
80%	MSEYTest/var(Y)	12.62%	13.62%	5.16E+03	10.81%	12.47%	2.16E+04	10.46%	12.52%	2.18E+04
80%	MSEYCensus/var(Y)	12.33%	13.40%	2.58E+03	10.61%	12.07%	1.08E+04	10.33%	12.29%	1.09E+04
50%	MSEYInSamp/var(Y)	13.32%	13.50%	13.74%	11.33%	12.67%	12.11%	10.69%	12.22%	11.31%
50%	MSEYTest /var(Y)	15.36%	15.39%	4.52E+02	12.15%	13.70%	4.39E+03	11.35%	12.70%	2.25E+04
50%	MSEYCensus/var(Y)	14.75%	14.82%	3.16E+02	11.90%	13.39%	3.07E+03	11.15%	12.56%	1.57E+04
30%	MSEYInSamp/var(Y)	13.49%	14.98%	13.83%	11.67%	12.72%	12.25%	11.10%	12.27%	12.06%
30%	MSEYTest/var(Y)	17.97%	18.55%	1.50E+02	13.20%	13.92%	2.47E+03	12.10%	13.45%	8.48E+03
30%	MSEYCensus/var(Y)	17.08%	17.83%	1.20E+02	12.90%	13.68%	1.97E+03	11.90%	13.21%	6.79E+03
100%	K (Full Census)	8.7400	2.4600	80.1600	11.8400	2.3400	154.8200	12.6800	2.2200	229.7600
100%	Time (Full Census)	5.5555	0.5153	1.2451	25.9777	0.9054	8.4133	55.2034	1.1440	27.9051

Table A14. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=3, sigma = 0.05

d = 3, sigma = 0.4		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0012	0.0019	0.0018	0.0008	0.0018	0.0011	0.0006	0.0010	0.0008
100%	MSEFTesting	0.0014	0.0022	978.6187	0.0009	0.0019	1562.1317	0.0006	0.0010	1432.7275
100%	MSEFCensus	0.0012	0.0019	0.0018	0.0008	0.0018	0.0011	0.0006	0.0010	0.0008
80%	MSEFInSamp	0.0019	0.0062	0.0027	0.0012	0.0021	0.0018	0.0010	0.0013	0.0014
80%	MSEFTesting	0.0026	0.0068	83.7592	0.0014	0.0023	545.8418	0.0011	0.0015	711.7752
80%	MSEFCensus	0.0023	0.0065	41.8809	0.0013	0.0022	272.9218	0.0010	0.0014	355.8883
50%	MSEFInSamp	0.0028	0.0031	0.0040	0.0017	0.0035	0.0025	0.0013	0.0016	0.0019
50%	MSEFTesting	0.0038	0.0043	2.3712	0.0019	0.0038	128.7269	0.0015	0.0019	272.2648
50%	MSEFCensus	0.0035	0.0039	1.6611	0.0019	0.0037	90.1096	0.0015	0.0018	190.5859
30%	MSEFInSamp	0.0031	0.0049	0.0047	0.0022	0.0049	0.0034	0.0017	0.0032	0.0025
30%	MSEFTesting	0.0050	0.0076	0.3994	0.0029	0.0058	32.1949	0.0020	0.0037	216.3175
30%	MSEFCensus	0.0047	0.0071	0.3205	0.0028	0.0056	25.7566	0.0019	0.0036	173.0545
100%	MSEYInSamp/var(Y)	30.88%	33.47%	33.26%	30.17%	32.97%	31.16%	29.39%	30.50%	30.05%
100%	MSEYTest/var(Y)	31.74%	33.95%	2.76E+04	30.31%	33.29%	4.40E+04	29.87%	30.97%	4.04E+04
100%	MSEYCensus/var(Y)	30.88%	33.47%	33.26%	30.17%	32.97%	31.16%	29.39%	30.50%	30.05%
80%	MSEYInSamp/var(Y)	32.80%	44.99%	35.55%	31.14%	33.94%	33.14%	30.56%	31.56%	31.42%
80%	MSEYTest/var(Y)	35.25%	47.03%	2.36E+03	31.69%	34.37%	1.54E+04	30.79%	32.05%	2.01E+04
80%	MSEYCensus/var(Y)	34.02%	46.01%	1.18E+03	31.42%	34.15%	7.69E+03	30.68%	31.81%	1.00E+04
50%	MSEYInSamp/var(Y)	36.51%	38.92%	37.60%	32.65%	36.69%	34.75%	32.46%	33.44%	33.41%
50%	MSEYTest /var(Y)	38.51%	39.85%	6.71E+01	33.41%	38.79%	3.63E+03	32.38%	33.21%	7.67E+03
50%	MSEYCensus/var(Y)	37.91%	39.57%	4.71E+01	33.18%	38.16%	2.54E+03	32.40%	33.28%	5.37E+03
30%	MSEYInSamp/var(Y)	35.28%	45.37%	38.53%	34.52%	43.72%	38.14%	32.68%	36.12%	35.10%
30%	MSEYTest/var(Y)	42.10%	49.42%	1.16E+01	36.16%	44.05%	9.07E+02	33.56%	38.26%	6.09E+03
30%	MSEYCensus/var(Y)	40.73%	48.61%	9.32E+00	35.83%	43.99%	7.26E+02	33.39%	37.83%	4.88E+03
100%	K (Full Census)	8.4200	2.5800	69.2400	11.6200	2.4400	140.6800	12.7000	2.3800	211.5800
100%	Time (Full Census)	5.3340	0.5241	1.2568	24.1476	0.9520	8.8607	53.7186	1.1511	29.1077

Table A15. $MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}$, time, and K results for d=3, sigma = 0.1

d = 4, sigma = 0.2		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0003	0.0007	0.0001	0.0003	0.0007	0.0001	0.0004	0.0007	0.0001
100%	MSEFTesting	0.0005	0.0010	161.6089	0.0003	0.0009	365.6785	0.0005	0.0008	584.3493
100%	MSEFCensus	0.0003	0.0007	0.0001	0.0003	0.0007	0.0001	0.0004	0.0007	0.0001
80%	MSEFInSamp	0.0004	0.0005	0.0001	0.0003	0.0007	0.0001	0.0003	0.0007	0.0001
80%	MSEFTesting	0.0008	0.0011	8.6689	0.0005	0.0009	204.9950	0.0004	0.0009	263.8065
80%	MSEFCensus	0.0006	0.0008	4.3345	0.0004	0.0008	102.4976	0.0003	0.0008	131.9033
50%	MSEFInSamp	0.0005	0.0005	0.0001	0.0004	0.0006	0.0001	0.0003	0.0006	0.0001
50%	MSEFTesting	0.0013	0.0014	1.2656	0.0007	0.0010	25.1762	0.0005	0.0009	91.1381
50%	MSEFCensus	0.0011	0.0011	0.8860	0.0006	0.0009	17.6233	0.0004	0.0008	63.7967
30%	MSEFInSamp	0.0008	0.0006	0.0001	0.0005	0.0005	0.0001	0.0004	0.0006	0.0001
30%	MSEFTesting	0.0017	0.0016	0.1625	0.0010	0.0013	1.8949	0.0007	0.0010	22.4333
30%	MSEFCensus	0.0015	0.0014	0.1301	0.0009	0.0011	1.5159	0.0007	0.0009	17.9467
100%	MSEYInSamp/var(Y)	1.91%	3.98%	0.89%	1.78%	4.14%	0.81%	2.40%	4.11%	0.77%
100%	MSEYTest/var(Y)	2.83%	5.40%	8.12E+03	2.21%	4.81%	1.84E+04	2.75%	4.68%	2.94E+04
100%	MSEYCensus/var(Y)	1.91%	3.98%	0.89%	1.78%	4.14%	0.81%	2.40%	4.11%	0.77%
80%	MSEYInSamp/var(Y)	2.42%	3.23%	0.95%	1.94%	3.86%	0.88%	1.78%	4.03%	0.84%
80%	MSEYTest/var(Y)	4.67%	5.88%	4.36E+02	2.87%	5.11%	1.03E+04	2.30%	4.84%	1.33E+04
80%	MSEYCensus/var(Y)	3.54%	4.56%	2.18E+02	2.40%	4.48%	5.15E+03	2.04%	4.44%	6.63E+03
50%	MSEYInSamp/var(Y)	3.26%	2.86%	0.90%	2.28%	3.32%	0.93%	1.96%	3.60%	0.88%
50%	MSEYTest /var(Y)	7.25%	7.28%	6.36E+01	4.20%	5.45%	1.27E+03	3.07%	5.14%	4.58E+03
50%	MSEYCensus/var(Y)	6.05%	5.95%	4.45E+01	3.62%	4.81%	8.86E+02	2.73%	4.68%	3.21E+03
30%	MSEYInSamp/var(Y)	4.70%	3.30%	0.99%	2.77%	3.02%	1.02%	2.27%	3.36%	0.93%
30%	MSEYTest/var(Y)	9.04%	8.50%	8.17E+00	5.72%	6.80%	9.52E+01	4.20%	5.73%	1.13E+03
30%	MSEYCensus/var(Y)	8.17%	7.46%	6.54E+00	5.13%	6.04%	7.62E+01	3.81%	5.25%	9.02E+02
100%	K (Full Census)	7.2600	2.0000	96.1600	11.8800	2.0000	190.2800	12.5000	2.0600	276.8200
100%	Time (Full Census)	4.3435	0.4711	1.1840	37.5370	1.2684	8.2187	75.3199	1.6647	26.7982

Table A16. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=4, sigma = 0.01

d = 4, sigma = 0.3		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0007	0.0010	0.0009	0.0005	0.0009	0.0007	0.0005	0.0008	0.0005
100%	MSEFTesting	0.0009	0.0012	440.3271	0.0006	0.0010	1105.7870	0.0006	0.0009	630.5477
100%	MSEFCensus	0.0007	0.0010	0.0009	0.0005	0.0009	0.0007	0.0005	0.0008	0.0005
80%	MSEFInSamp	0.0009	0.0024	0.0013	0.0007	0.0010	0.0009	0.0005	0.0009	0.0007
80%	MSEFTesting	0.0014	0.0029	20.5826	0.0009	0.0012	701.4585	0.0007	0.0011	1027.9976
80%	MSEFCensus	0.0011	0.0026	10.2919	0.0008	0.0011	350.7297	0.0006	0.0010	513.9992
50%	MSEFInSamp	0.0012	0.0014	0.0016	0.0009	0.0011	0.0012	0.0007	0.0010	0.0009
50%	MSEFTesting	0.0021	0.0023	2.8493	0.0012	0.0015	43.0414	0.0010	0.0013	281.3078
50%	MSEFCensus	0.0018	0.0020	1.9949	0.0011	0.0013	30.1293	0.0009	0.0012	196.9158
30%	MSEFInSamp	0.0015	0.0014	0.0020	0.0011	0.0014	0.0015	0.0008	0.0011	0.0012
30%	MSEFTesting	0.0025	0.0026	0.2964	0.0017	0.0020	3.9116	0.0013	0.0016	45.9066
30%	MSEFCensus	0.0023	0.0024	0.2375	0.0016	0.0019	3.1296	0.0012	0.0015	36.7255
100%	MSEYInSamp/var(Y)	14.39%	15.53%	15.32%	13.42%	15.10%	14.27%	13.43%	14.89%	13.50%
100%	MSEYTest/var(Y)	15.15%	16.60%	1.98E+04	13.81%	15.78%	4.98E+04	13.72%	15.36%	2.84E+04
100%	MSEYCensus/var(Y)	14.39%	15.53%	15.32%	13.42%	15.10%	14.27%	13.43%	14.89%	13.50%
80%	MSEYInSamp/var(Y)	15.24%	21.56%	16.96%	14.36%	15.53%	15.23%	13.73%	15.29%	14.44%
80%	MSEYTest/var(Y)	17.22%	24.05%	9.27E+02	15.22%	16.76%	3.16E+04	14.26%	16.11%	4.63E+04
80%	MSEYCensus/var(Y)	16.23%	22.81%	4.64E+02	14.79%	16.15%	1.58E+04	13.99%	15.70%	2.32E+04
50%	MSEYInSamp/var(Y)	16.60%	16.98%	17.15%	14.90%	15.80%	16.48%	14.51%	15.54%	15.04%
50%	MSEYTest /var(Y)	20.50%	21.37%	1.28E+02	16.42%	17.72%	1.94E+03	15.57%	17.08%	1.27E+04
50%	MSEYCensus/var(Y)	19.33%	20.05%	9.00E+01	15.96%	17.15%	1.36E+03	15.25%	16.61%	8.87E+03
30%	MSEYInSamp/var(Y)	19.00%	17.01%	19.47%	15.70%	16.93%	18.51%	14.86%	15.72%	16.31%
30%	MSEYTest/var(Y)	22.59%	22.90%	1.35E+01	18.81%	20.15%	1.76E+02	17.02%	18.44%	2.07E+03
30%	MSEYCensus/var(Y)	21.87%	21.72%	1.08E+01	18.19%	19.51%	1.41E+02	16.59%	17.90%	1.65E+03
100%	K (Full Census)	7.0800	2.0000	76.1800	11.8400	2.0600	154.8800	12.7800	2.0600	229.4800
100%	Time (Full Census)	4.1409	0.4665	1.3064	35.8179	1.2787	9.3666	77.7415	1.6637	29.6764

Table A17. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=4, sigma = 0.05

d = 4, sigma = 0.4		nFull = 100			nFull = 200			nFull = 300		
nL/nF	Metric	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS	CAP-NLS	CAP	CNLS
100%	MSEFInSamp	0.0015	0.0018	0.0025	0.0010	0.0013	0.0017	0.0008	0.0012	0.0014
100%	MSEFTesting	0.0017	0.0022	322.4462	0.0010	0.0015	1624.4689	0.0009	0.0013	2548.5803
100%	MSEFCensus	0.0015	0.0018	0.0025	0.0010	0.0013	0.0017	0.0008	0.0012	0.0014
80%	MSEFInSamp	0.0025	0.0030	0.0043	0.0015	0.0017	0.0025	0.0012	0.0015	0.0019
80%	MSEFTesting	0.0031	0.0043	20.1828	0.0017	0.0020	398.8514	0.0013	0.0017	1202.0194
80%	MSEFCensus	0.0028	0.0036	10.0935	0.0016	0.0019	199.4270	0.0012	0.0016	601.0107
50%	MSEFInSamp	0.0031	0.0060	0.0055	0.0021	0.0025	0.0035	0.0017	0.0030	0.0029
50%	MSEFTesting	0.0044	0.0073	0.4835	0.0026	0.0030	49.4704	0.0019	0.0033	372.3199
50%	MSEFCensus	0.0040	0.0069	0.3401	0.0024	0.0028	34.6304	0.0018	0.0032	260.6248
30%	MSEFInSamp	0.0033	0.0034	0.0060	0.0023	0.0028	0.0041	0.0021	0.0022	0.0035
30%	MSEFTesting	0.0048	0.0047	0.5589	0.0030	0.0037	11.4710	0.0024	0.0027	96.0842
30%	MSEFCensus	0.0045	0.0045	0.4483	0.0029	0.0035	9.1776	0.0024	0.0026	76.8680
100%	MSEYInSamp/var(Y)	38.72%	39.26%	41.75%	36.67%	37.81%	39.35%	36.25%	37.24%	38.10%
100%	MSEYTest/var(Y)	39.08%	40.68%	1.08E+04	36.89%	38.43%	5.45E+04	36.26%	37.44%	8.55E+04
100%	MSEYCensus/var(Y)	38.72%	39.26%	41.75%	36.67%	37.81%	39.35%	36.25%	37.24%	38.10%
80%	MSEYInSamp/var(Y)	41.10%	42.36%	47.45%	38.75%	39.00%	41.91%	37.54%	38.76%	39.88%
80%	MSEYTest/var(Y)	43.62%	47.77%	6.78E+02	39.19%	40.13%	1.34E+04	37.80%	39.28%	4.03E+04
80%	MSEYCensus/var(Y)	42.36%	45.07%	3.39E+02	38.97%	39.56%	6.69E+03	37.67%	39.02%	2.02E+04
50%	MSEYInSamp/var(Y)	43.13%	51.83%	48.51%	39.64%	41.45%	45.42%	39.32%	42.94%	43.20%
50%	MSEYTest /var(Y)	48.08%	57.66%	1.66E+01	41.71%	43.21%	1.66E+03	39.72%	44.51%	1.25E+04
50%	MSEYCensus/var(Y)	46.60%	55.91%	1.17E+01	41.09%	42.68%	1.16E+03	39.60%	44.04%	8.75E+03
30%	MSEYInSamp/var(Y)	46.76%	42.65%	51.13%	39.99%	41.21%	48.58%	39.50%	40.20%	45.14%
30%	MSEYTest/var(Y)	49.24%	49.12%	1.91E+01	43.41%	45.69%	3.85E+02	41.36%	42.13%	3.22E+03
30%	MSEYCensus/var(Y)	48.75%	47.83%	1.54E+01	42.73%	44.79%	3.08E+02	40.99%	41.74%	2.58E+03
100%	K (Full Census)	6.9800	2.0000	67.6400	11.9000	2.0400	142.6800	12.5400	2.0000	217.2600
100%	Time (Full Census)	4.0251	0.4753	1.3533	36.7225	1.2767	9.4859	75.9045	1.6847	30.9289

Table A18. MSE_f, MSE_{ISf}, MSE_{FSf}, MSE_y, MSE_{ISy}, MSE_{FSy}, time, and K results for d=4, sigma = 0

Appendix B. Scalability of CAP-NLS to larger datasets

To demonstrate the performance of CAP-NLS in large data sets, we revisit the DGP used in Section 3.2, specifically, $Y_i = X_{i1}^{0.4} X_{i2}^{0.3} X_{i3}^{0.2} + \varepsilon_i$, where $\varepsilon_i \sim N(0, \sigma^2)$, $\sigma = 0.1$ and $X_{ij} \sim Unif(0.1,1)$ for $j = 1,2,3,\ i = 1, \dots, n$, and $n = 500,\ 1000,\ 2000,\ 3000,$ and 5000. Table B1 reports the estimator's performance[18].

n	500	1000	2000	3000	5000
$RMSE f_{Learn}^{CAP-NLS}$	0.025	0.022	0.025	0.025	0.024
$RMSE f_{Learn}^{CAP-NLSF}$	0.023	0.025	0.026	0.027	0.028
$MSE y_{Learn}^{CAP-NLS}$	0.009	0.010	0.011	0.011	0.011
$MSE y_{Learn}^{CAP-NLSF}$	0.010	0.011	0.010	0.011	0.011
$K^{CAP-NLS}$	5	5	7	5	5
$K^{CAP-NLSF}$	4	4	5	4	4
$Time(min)^{CAP-NLS}$	3	8	43	114	367
$Time(min)^{CAP-NLSF}$	3	5	10	11	41

Table B1. Number of Hyperplanes and Runtimes for Trivariate Input Cobb-Douglas DGP on Larger Datasets.

We first conduct standard CAP-NLS analysis and report learning errors, number of fitted hyperplanes and runtime results. Runtimes for datasets up to 2000 observations are well below the one hour threshold, but there are significant scalability challenges for datasets larger than 2000 observations. Thus, we apply the Fast CAP stopping criterion in Hannah and Dunson (2013), which measures the GCV score improvement by the addition of one more hyperplane and stops the algorithm if no improvement has been achieved in two consecutive additions. Unlike Fast CAP,

[18] Due to the increased computational burden of using larger datasets, we present results for a single replicate of the DGP for each sample size and only include learning set results. For this section we report $RMSE$ results rather than MSE ones, since the latter are very small and the differences are indistinguishable across settings.

however, we apply it directly to the learning error against observations. We denote the results for those runs with the CAP-NLSF superscript and observe that differences are minimal compared to following our standard partitioning strategy. This alternative stopping rule results in a highly scalable algorithm which can fit datasets up to 5000 observations in around 40 minutes.

Appendix C. Parametric Bootstrap algorithm to calculate expected optimism

We apply the following algorithm from Efron (2004) to compute in-sample optimism. First, we assume a Gaussian density $p(\boldsymbol{Y}) = N(\widehat{\boldsymbol{Y}}, \hat{\sigma}^2\boldsymbol{I})$, where $\widehat{\boldsymbol{Y}}$ is the vector of estimated output values of the estimator for which we are assessing the in-sample optimism. We obtain $\hat{\sigma}^2$ from the residuals of a "big" model presumed to have negligible bias. Given CNLS's high flexibility and complex description (many hyperplanes), we choose it as our "big" model. Although obtaining an unbiased estimate for σ^2 from CNLS's residuals is complicated, i.e., there are no formal results regarding the effective number of parameters CNLS uses, using MSE_{yLearn}^{CNLS} as $\hat{\sigma}^2$ results in a downward biased estimator of σ^2. This downward bias in fact results in improved efficiency for the parametric bootstrap algorithm and is an example of a "little" bootstrap (Breiman, 1992). Thus, we let $\hat{\sigma} = MSE_{yLearn}^{CNLS}$. Efron (2004) then suggests to run a large number B of simulated observations $\boldsymbol{Y}^*$ from $p(\boldsymbol{Y})$, fit them to obtain estimates $\widehat{\boldsymbol{Y}^*}$, and estimate $cov_i = cov(\widehat{Y_i}, Y_i)$ computing

$$\widehat{cov_i} = \sum_{b=1}^{B} \widehat{Y_i^{*b}}(Y_i^{*b} - Y_i^{*\cdot})/(B-1)\ ;\ Y_i^{*\cdot} = \sum_{b=1}^{B} Y_i^{*b}/B. \tag{C1}$$

We select B = 500 for all our experiments based on observed convergence of the $\sum_{i=1}^{n} \widehat{cov_i}$ quantity.

Further, we note that if the researcher is not comfortable with the assumption made about the size of MSE_{yLearn}^{CNLS} relative to σ^2, sensitivity analysis (by adding a multiplier $c > 1$, such that $p(\boldsymbol{Y}) = N(\widehat{\boldsymbol{Y}}, c\hat{\sigma}^2\boldsymbol{I})$) can be performed. Finally, we also note that non-Gaussian distributions can be used to draw the bootstrapped $\boldsymbol{Y}^*$vectors. This is especially useful when considering inefficiency, because it can include skewed distributions also.

Appendix D. Cobb-Douglas results with multiplicative residual assumption for Chilean manufacturing data

Industry Name and Code	n	Survey Size	R^2_{FS}	R^2_{CDM}	Ratio vs. Best Method
Other Metal Products (2899)	144	20%	50%	82%	CDM is Best Method
		30%	60%	85%	CDM is Best Method
		40%	64%	86%	CDM is Best Method
		50%	72%	86%	CDM is Best Method
		100%	88%	87%	CDM ties for Best Method
Wood (2010)	150	20%	35%	45%	CDM is Best Method
		30%	40%	50%	CDM is Best Method
		40%	47%	51%	CDM is Best Method
		50%	52%	53%	CDA ties for Best Method
		100%	66%	62%	0.94 vs. CAP-NLS
Structural Use Metal (2811)	161	20%	77%	79%	CDM ties for Best Method
		30%	82%	81%	CDM ties for Best Method
		40%	87%	84%	0.97 vs. CAP-NLS
		50%	90%	85%	0.94 vs. CAP-NLS
		100%	95%	92%	0.97 vs. CAP-NLS
Plastics (2520)	249	20%	54%	56%	CDM ties for Best Method
		30%	57%	56%	CDM ties for Best Method
		40%	57%	57%	CDM ties for Best Method
		50%	60%	57%	CDM ties for Best Method
		100%	64%	60%	0.94 vs. CAP-NLS
Bakeries (1541)	250	20%	72%	46%	0.64 vs. CAP
		30%	77%	50%	0.65 vs. CAP
		40%	78%	50%	0.64 vs. CAP
		50%	85%	51%	0.60 vs. CAP
		100%	99%	58%	0.59 vs. CAP

Table D1. Ratio of CDM to Best Model performance

Appendix E. Application Results for Infinite Populations

In this Appendix we explore the sensitivity of our application insights in the case when predictive ability at any point of the production function is equally important. This represents a key departure from the assumptions made in the main body of the paper, as it translates into not weighting in-sample and predictive errors. Rather, since we are interested in the descriptive ability of the fitted production function on an infinite number of unobserved input vectors, we only consider the predictive error. To illustrate the consequences of this alternative assumptions in detail, we present our results in Figure E1 which is analogous to Figure 4. In Figure E1 we show replicate-specific as well as averaged R^2 values. In this case, rather than using R^2_{FS} as our predictive power indicator, we use $R^2_{Pred} = \max(1 - E(\widehat{Err_y})/Var(Y_{FS}), 0)$.

In Figure E1, we observe that for the majority of studied industries, weighting the in-sample error with the predictive error, the direct consequence of our finite full population assumption, does not affect the diagnostic of the mean predictive power of our production function models. Using our notation, this means that for most industries the expected R^2_{Pred} for each given subsample size did not differ greatly from R^2_{FS}. However, the R^2_{Pred} figures have significantly higher variance than their R^2_{FS} counterparts for each subsample size. For all industries except for industry code 2811, we obtain at least one replicates with negligible predictive power. For some industries, such as Industry Codes 2520 and 1541, this causes the predictive power bound to be very wide (although the upper bound and mean values increase monotonically in the subsample size).

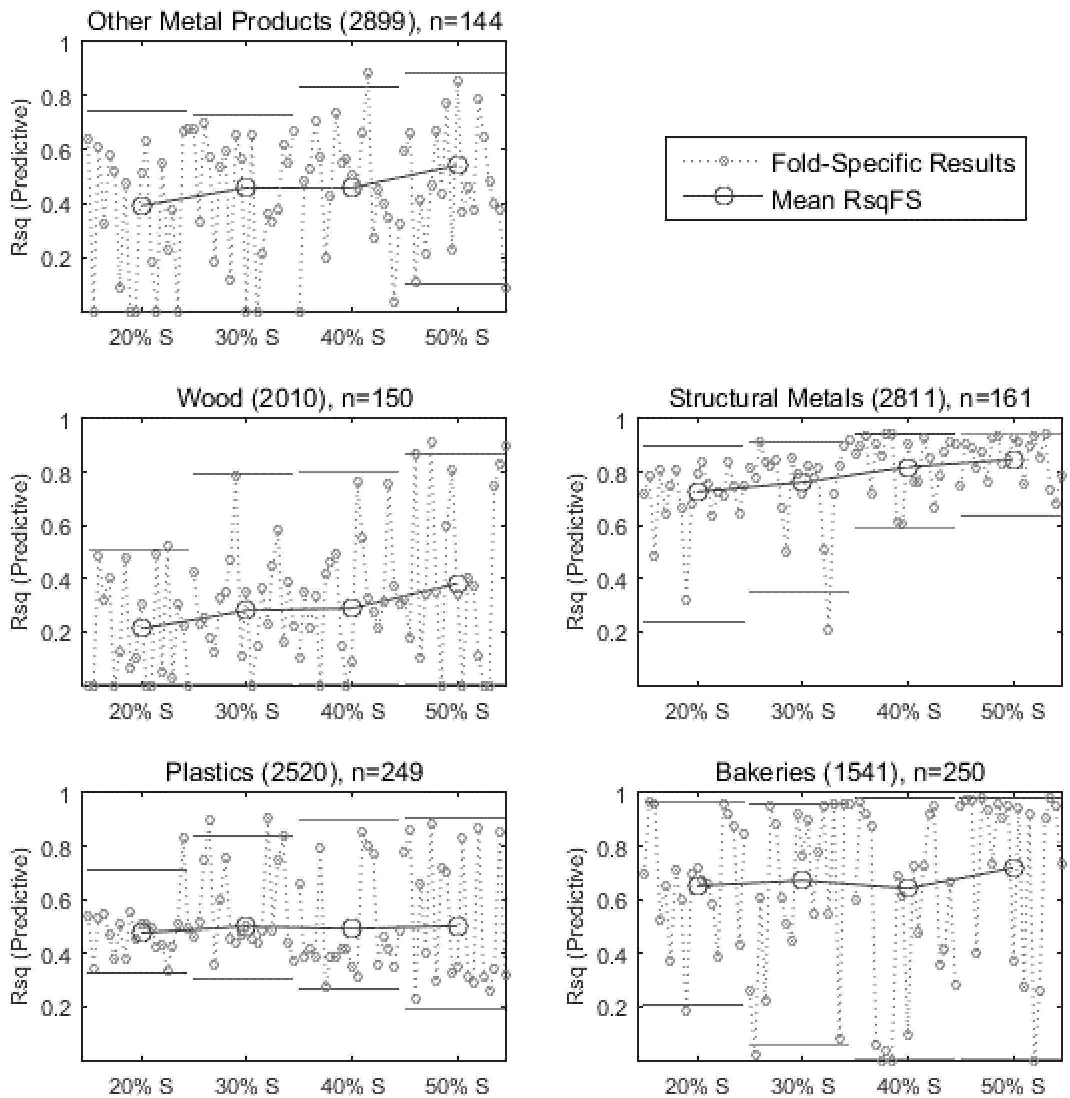

Figure E1. Best Method's R^2_{Pred} as function of relative subset size for selected industries. CAP-NLS was chosen as Best Method for industry codes 2899, 2010, 2811 and 2520, while CDA was chosen for industry code 1541.