Post-Estimation Smoothing:
A Simple Baseline for Learning with Side Information

Esther Rolf
UC Berkeley

Michael I. Jordan
UC Berkeley

Benjamin Recht
UC Berkeley

Abstract

Observational data are often accompanied by natural structural indices, such as time stamps or geographic locations, which are meaningful to prediction tasks but are often discarded. We leverage semantically meaningful indexing data while ensuring robustness to potentially uninformative or misleading indices. We propose a post-estimation smoothing operator as a fast and effective method for incorporating structural index data into prediction. Because the smoothing step is separate from the original predictor, it applies to a broad class of machine learning tasks, with no need to retrain models. Our theoretical analysis details simple conditions under which post-estimation smoothing will improve accuracy over that of the original predictor. Our experiments on large scale spatial and temporal datasets highlight the speed and accuracy of post-estimation smoothing in practice. Together, these results illuminate a novel way to consider and incorporate the natural structure of index variables in machine learning.

1 INTRODUCTION

The canonical machine learning setup models pairs of features and labels as originating from some underlying distribution, $\{x_i, y_i\} \sim \mathcal{D}(x, y)$; the problem is to learn a predictor $\widehat{y}(x)$ which describes y as faithfully as possible. However, a recent narrative in machine learning is that well-annotated, large-scale datasets are rare, whereas less curated data are abundant; this has led to a taxonomy of supervision including distant-, weak-, and semi- supervision. Whether labels are

noisy by nature (distant) (Mintz et al., 2009), programmatically generated (weak) (Ratner et al., 2016), or missing altogether (semi) (Zhu, 2005), it stands that characteristics of some data necessitate making use of additional sources of constraints.

Semi-supervised methods in particular aim to leverage unlabeled data to elicit an underlying structure which can aid prediction (Singh et al., 2009). In practice, however, semi-supervised methods can be computationally expensive, and are sensitive to distribution shifts (Oliver et al., 2018). We propose to use readily-available data that is inherently structural, and apply a robust post-processing method which is independent of the original predictor to incorporate this structure.

We consider scenarios where each datum (x, y) has an associated index t with some linking or semantic meaning. We thus represent observations as triplets:

$$\{x_i, y_i, t_i\} \quad i = 1, ..., n$$

Examples of such triplets include {image, annotation, frame number} in video prediction, {house attributes, price, address} in house price prediction, and {document, sentiment, keywords} in sentiment analysis. While intuition suggests that index variables t may be *correlated* with the label values y and thus are highly informative to the prediction task, in many cases they are not well suited as *predictors* of y without major modification. For example, in object detection in videos, we may expect objects to move smoothly across frames, but the frame number itself does not carry predictive power from one video to another.

We aim to leverage the structural information encoded in t without over-relying on it. This motivates a main question of our work: *how can we utilize the dependence of x and y on t even for predictors that might ignore or underestimate such dependence?* We propose a *post-estimation smoothing* (P-ES) operator $S(t)$ that only depends on t to obtain smoothed predictions:

$$\widetilde{y} = S(t)\widehat{y}(x).$$

Decoupling smoothing $S(t)$ from the initial feature-based prediction step $\widehat{y}(x)$ allows us to efficiently

smooth any off-the-shelf model. P-ES applies to any precomputed predictions made over time or space, regardless of the original predictive model. The ease of applying P-ES facilitates robust and reproducible incorporation of index variable structure in predictions.

Problem Statement Throughout this work we consider the setting in which we have a dataset indexed by $t_i \in \mathbb{R}^l$, as well as predictions $\widehat{y_i} \in \mathbb{R}$ associated with each index. It is natural to consider that there is also a set of features $x_i \in \mathbb{R}^d$ and model $f : \mathbb{R}^d \to \mathbb{R}$ from which predictions $\widehat{y} = f(x)$ were generated; we take this as given and work with directly with $\widehat{y}$.

We study the post-prediction application of a P-ES matrix operator $S(t) : \mathbb{R}^n \to \mathbb{R}^n$ to form smoothed predictions, $\widetilde{y} := S(t)\widehat{y}$, such that the $\widetilde{y}$ are closer to the true labels y than the original unsmoothed predictions $\widehat{y}$ are. In our theoretical analysis in Section 3, this is measured using the expected mean-squared error: $\mathbb{E}\left[\frac{1}{n}\|\widetilde{y} - y\|_2^2\right] = \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(\widetilde{y}_i - y_i)^2\right]$, while experiments in Section 4 consider different accuracy metrics suitable to different contexts.

Our main contributions are:

- The formulation of a structural-index-based post-process smoothing procedure, P-ES, which is applicable to any predictor.

- Theoretical results proving that under mild conditions, P-ES will improve accuracy relative to the original predictor (Theorem 1), and characterizing when a linear smoothing operation can greatly increase predictive accuracy (Lemma 1).

- Experiments on large-scale datasets for human pose estimation and house price prediction demonstrating that P-ES improves accuracy of state-of-the-art predictors at minimal extra cost.

These contributions are made possible by incorporating the general index variables separately from the feature based predictor. This results in a fast, accurate, and robust method for local variance reduction, with the potential to change how we consider and leverage structural variables in machine learning predictions.

More broadly, we demonstrate the effectiveness of a simple method that extends and generalizes previous scholarship in locality-based semi-supervised learning and nonparametric regression, applied in a modern context of abundant but weakly predictive data. Given recent exposition of the systematic underreporting of simple baselines (Dacrema et al., 2019; Mania et al., 2018) in machine learning and especially in semi-supervised learning (Oliver et al., 2018), it is worth considering P-ES as a theoretically motivated and eas-

ily implementable baseline for semi-supervised learning and smoothing in large scale, real-data contexts.

2 RELATED WORK

Semi-supervised learning (SSL) methods leverage large amounts of unlabeled data along with some labeled data under a local consistency assumption: instances which are near to each other should have similar label values. Distance is commonly determined with respect to an underlying manifold or graph defined by the features (Belkin and Niyogi, 2004; Zhu et al., 2003).

To encourage local consistency of predictions, Belkin et al. (2006) add a Laplacian regularization term to least squares and support vector machines, and Jean et al. (2018) add a spatial regularization to the loss function of deep neural nets. There is also a considerable amount of work incorporating additive consistency regularization and similar notions in neural nets (Tarvainen and Valpola, 2017; Bachman et al., 2014; Grandvalet and Bengio, 2005). As noted by Oliver et al. (2018), however, such methods can be sensitive to distribution shifts and require large validation sets and heavy computation to tune parameters; thus they are often poorly suited for "real-world" applications.

Unfortunately, adding a local consistency regularization term multiplies the number of parameter configurations in the optimization problem, and only works for predictors with an explicit objective function. For example, it not straightforward to add a spatial consistency term to random forests. An alternative method, Gaussian harmonic energy minimization (HEM) (Zhu et al., 2003), augments an underlying graph with noisy predictions, and solves for spatially consistent predictions on this larger graph. The local and global consistency (LCG) algorithm (Zhou et al., 2004a) solves a similar optimization problem iteratively.

Singh et al. (2009) show that unlabeled data is useful in SSL precisely when it illuminates the underlying structure of the data beyond what was discernible by the labeled data alone. However, if modeling assumptions incorrectly summarize the true structure of the data, unlabeled data can be misleading and even degrade performance (Cozman and Cohen, 2002; Oliver et al., 2018). One approach to mitigate this is to fortify semi-supervised learning methods to be robust to this mismatch (Li and Zhou, 2014); we obtain robustness by decoupling feature-based prediction from a nonparametric incorporation of structural indices.

The literature on nonparametric regression methods is extensive (Tsybakov, 2010; Györfi et al., 2006); we focus on two prominent approaches. Gaussian Process Regression (GPR) places a Gaussian prior on label co-

variances, specified by feature variables (Williams and Rasmussen, 2006). GPR has been widely adopted and extended in the geospatial statistics community, under the names of "kriging" and "inverse distance interpolation" (Babak and Deutsch, 2009; Lu and Wong, 2008). As pointed out by Banerjee et al. (2008), when applied to large datasets, GPR has large computation and memory requirements, or necessitates approximations (Williams and Rasmussen, 2006).

Kernel smoothing (Tsybakov, 2010) is another type of nonparametric regression in which predictions are locally weighted averages of observations. Of particular note is the Nadaraya-Watson estimator (Nadaraya, 1964; Watson, 1964) in which weights are determined by a kernel relation on all instances. GPR, Laplacian regularized least squares (Belkin et al., 2006), HEM (Zhu et al., 2003), and exact LCG (Zhou et al., 2004a) can all be cast as instances of linear smoothing operators for which computing the smoothing matrix involves inverting a matrix of size $n \times n$.

Lastly, recent work on the statistical optimality of data interpolation in machine learning (Belkin et al., 2018) highlights that averaging methods are quite powerful for prediction. We apply a locally-weighted average to *predictions* themselves, which are the output of some prior model. Like semi-supervised learning methods, P-ES encodes spatial consistency properties, but takes the perspective of refining given predictions with minimal restrictions on the underlying structure. In light of the previous work, P-ES can be seen as a fast and robust way to leverage structure, and to interpolate. Application-specific references are provided in Sec. 4.

3   ANALYSIS

Here we answer the questions *(i) how should we form a useful post-estimation smoothing matrix while maintaining robustness to possible distributional misspecification?* and *(ii) for what data distributions and predictors is linear smoothing beneficial?* Throughout the analysis, we model true values y, predictions $\widehat{y}$, and error residuals ε as stochastic processes indexed by t:

$$\widehat{y}(t) = y(t) + \varepsilon(t) \ . \tag{1}$$

3.1   Accuracy Increases with General Smoothing Matrices

While we may have strong intuition that there is *some* locality-based structure in certain domains, the choice of distributional priors governing this structure will most often be inexact. We use a matrix $W(t) \in \mathbb{R}^{n \times n}$, where weights W_{ij} denote how much the jth prediction should contribute to a smoothed estimate for the ith instance, depending on the values of t_i and t_j.

Theorem 1 below shows that using a reasonable weight matrix $W(t)$ which captures correlation in the underlying data can improve performance. A key insight is that shrinking W towards the identity matrix tempers potential misspecification gracefully. Therefore, we form our smoothing matrix as the convex combination:

$$S_c(t) = c \cdot W(t) + (1 - c) \cdot I \ , \tag{2}$$

where in practice $c \in [0, 1]$ can be chosen through cross-validation along with any parameters of W.

For any weight matrix, define the following quantities: $\gamma(\varepsilon, W)$, which describes the amount by which W acts as a zero operator on the errors, and $\beta(\varepsilon, W; y)$, which describes the amount by which W acts as the identity operator on the true labels, both scaled by $\mathbb{E}[\|\varepsilon\|_2^2]$:

$$\gamma(\varepsilon, W) := \mathbb{E}[\varepsilon^\top W \varepsilon] / \mathbb{E}[\|\varepsilon\|_2^2],$$
$$\beta(\varepsilon, W; y) := \mathbb{E}[\varepsilon^\top (W - I) y] / \mathbb{E}\left[\|\varepsilon\|_2^2\right] \ .$$

Intuitively, we want to use a weight matrix W such that both γ and β are small, so that W averages out erroneous error signals while decreasing correlation between y and ε. Theorem 1 shows that an imperfect W will suffice, so long as the sum $\gamma + \beta$ is controlled.

Theorem 1. *Given any predictor $\widehat{y}$ of y with error residuals satisfying $\mathbb{E}\left[\|\varepsilon\|_2^2\right] \neq 0$, and any weight matrix W satisfying $\gamma(\varepsilon, W) + \beta(\varepsilon, W; y) < 1$, there exists a constant $c \in (0, 1]$ such that the smoothing matrix $S_c = c \cdot W + (1 - c) \cdot I$ strictly reduces expected MSE:*

$$\mathbb{E}\left[\frac{1}{n}\|S_c\widehat{y} - y\|_2^2\right] < \mathbb{E}\left[\frac{1}{n}\|\widehat{y} - y\|_2^2\right] \ .$$

(Proof sketch for $\beta = 0$.) For unbiased estimators $\widehat{y}$ with errors ε that are independent of the labels, $\beta = 0$ and the objective decomposes as

$$\mathbb{E}\left[\|S_c\widehat{y} - y\|_2^2 - \|\widehat{y} - y\|_2^2\right]$$
$$\leq c^2 \left(\mathbb{E}\left[\|W\widehat{y} - y\|_2^2\right] + 2(1 - \gamma)\mathbb{E}\left[\|\varepsilon\|_2^2\right]\right)$$
$$+ 2c(\gamma - 1)\mathbb{E}\left[\|\varepsilon\|_2^2\right] \ .$$

The upper bound is a convex quadratic function in c with optimum at

$$c^* = \frac{(1 - \gamma)\mathbb{E}\left[\|\varepsilon\|_2^2\right]}{(\mathbb{E}\left[\|W\widehat{y} - y\|_2^2\right] + 2(1 - \gamma)\mathbb{E}\left[\|\varepsilon\|_2^2\right])}$$

By the theorem conditions, $\gamma < 1$, so that $c^* \in (0, 1]$. The resulting upper bound is then given by

$$\mathbb{E}\left[\frac{1}{n}\|S_{c^*}\widehat{y} - y\|_2^2 - \frac{1}{n}\|\widehat{y} - y\|_2^2\right]$$
$$\leq -\frac{(1 - \gamma)^2\mathbb{E}\left[\|\varepsilon\|_2^2\right]^2}{n(\mathbb{E}\left[\|W\widehat{y} - y\|_2^2\right] + 2(1 - \gamma)\mathbb{E}\left[\|\varepsilon\|_2^2\right])} < 0 \ . \ \square$$

We present the full proof in Appendix A.2.

Theorem 1 inverts the standard statistical smoothing analysis (Simonoff, 1998; Tsybakov, 2010)—which, provided generative processes as in Eq. (1), calculates the bias and variance of the resulting smoothed estimator—and instead characterizes properties of the underlying signal $y(t)$, the prediction errors $\varepsilon(t)$, and the weight matrix W that make smoothing beneficial. As in standard kernel smoothing regression (Wand and Jones, 1994), in P-ES we are willing to tolerate an increase in the bias of our predictor, so long as the variance decreases. Formulating the conditions in terms of γ and β allows us to assess this trade-off in terms of the conditions on the prediction errors directly.

We can guarantee that $\gamma(\varepsilon, W) \leq 1$ by ensuring $\lambda_{\max}(W) \leq 1$, for example by taking any right-stochastic matrix (here $\lambda_{\max}(\cdot)$ denotes the maximum eigenvalue). Controlling $\beta(\varepsilon, W; y)$ depends on both the true values and the errors in predictions. A sufficient condition to achieving $\beta \leq c$ is to pick W which satisfies $\mathbb{E}\left[\|(W - I)y\|_2^2\right] \leq c^2 \mathbb{E}\left[\|\varepsilon\|_2^2\right]$, encoding that we tolerate a deviation in labels due to W limited by the magnitude of errors that can potentially be reduced. See Appendix A.1 for ways to ensure $\gamma + \beta < 1$.

In general, the condition $\beta + \gamma < 1$ represents a trade-off in choosing a weight matrix that acts approximately as a zero matrix with respect to the errors (small γ), while acting close to an identity matrix with respect to the true values (small β). In order to keep the sum small, W needs to incorporate knowledge in the structure of the domain-specific labels, y, as well as the distribution of prediction errors, ε, for a given predictor. For all experiments (Section 4), we use the Nadaraya-Watson smoothing matrix with a Gaussian kernel (Eq. (4)). This matrix is right-stochastic, and the Gaussian kernel encodes the constraint that nearby data points (measured with respect to structural index variable t) should have similar label values y.

The best-case reduction in MSE attainable by P-ES is bounded in the final line of the proof of Theorem 1 (for $\beta \neq 0$, see Appendix A.1). The MSE reduction depends on covariances between W, y, and the error residuals in $\hat{y}$. This motivates us to study the form of the optimal smoothing operator and the resulting expected error reduction, when these covariances are known.

3.2 Optimal P-ES for Known Distributions

Having proposed a P-ES matrix S_c in Section 3.1, we now study the form of an optimal linear smoothing matrix S^* when the distributions governing the labels and error residuals are known. This reinforces the high-level structures we wish to capture in S_c, and provides a baseline for simulation experiments in Section 4.1. Denote the cross-correlation matrices K element-wise as $K_{xy}[t, s] = \mathbb{E}[x(t)y(s)]$.

Lemma 1. *For a predictor $\hat{y}$ of y with error residuals distributed as $\varepsilon(t) = \hat{y}(t) - y(t)$, when $K_{\hat{y}\hat{y}} \succ 0$, the optimal linear smoothing matrix has the form*

$$S^* = \arg\min_{S \in \mathbb{R}^{n \times n}} \mathbb{E}\left[\tfrac{1}{n}\|S\hat{y} - y\|_2^2\right]$$
$$= I - (K_{\varepsilon\varepsilon} + K_{y\varepsilon})^\top (K_{yy} + K_{y\varepsilon} + K_{\varepsilon y} + K_{\varepsilon\varepsilon})^{-1} .$$

The expected MSE reduction of applying S^* versus using the original predictions $\hat{y}$ is always non-negative, and is given by

$$\tfrac{1}{n}\mathbb{E}\left[\|\hat{y} - y\|_2^2\right] - \|S^*\hat{y} - y\|_2^2 = \tfrac{1}{n}\mathrm{tr}\left(K_{\hat{y}y}^\top (K_{\hat{y}\hat{y}})^{-1} K_{\hat{y}y}\right) .$$

We present the proof of Lemma 1 in Appendix A.3.

Lemma 1 shows that smoothing can reduce prediction error associated with $K_{\varepsilon\varepsilon}$ and $K_{y\varepsilon}$, but that the extent to which errors can be smoothed out depends on the forms of K_{yy} and $K_{y\varepsilon}$. The following example underscores this point for an illustrative data generating model and sets the stage for simulation experiments in Section 4.1. The main details of the example are given here, with more extensive exposition in Appendix A.4.

Example 3.1. *Consider zero-mean stochastic processes $x(t)$ and $y(t)$ which are dependent on a third zero-mean hidden process $z(t)$, but with independent additive Gaussian noise. In particular:*

$$z \sim \mathcal{N}(0, K_{zz}) \tag{3}$$
$$x(t) = z(t) + \omega(t), \qquad \omega(t) \sim_{i.i.d.} \mathcal{N}(0, \sigma_x^2)$$
$$y(t) = c \cdot z(t) + \mu(t), \qquad \mu(t) \sim_{i.i.d.} \mathcal{N}(0, \sigma_y^2) .$$

The autocorrelation matrices show that there is shared variation due to the "hidden" process z:

$$K_{xx} = K_{zz} + K_{\omega\omega}, \quad K_{yy} = c^2 K_{zz} + K_{\mu\mu}, \quad K_{xy} = cK_{zz} .$$

Without any specific knowledge of the covariance structure in z, this could be modeled with an "errors in variables" model, for which total least squares (TLS) gives a statistically consistent estimator of c. In the appendix, we show that as n grows large, the expected MSE of the TLS predictions approaches

$$\mathbb{E}\left[\tfrac{1}{n}\|\hat{y}_{TLS} - y\|_2^2\right] \approx \sigma_y^2 + c^2 \sigma_x^2 ,$$

whereas invoking Lemma 1, the expected smoothed performance using S^* approaches

$$\mathbb{E}\left[\tfrac{1}{n}\|S^*\hat{y}_{TLS} - y\|_2^2\right]$$
$$\approx \sigma_y^2 + c^2 \sigma_x^2 \left(1 - \tfrac{1}{n}\mathrm{tr}\left((\sigma_x^{-2}K_{zz} + I)^{-1}\right)\right) \geq \sigma_y^2 .$$

Using the first line above, the expected MSE reduction is approximately $\frac{c^2 \sigma_x^2}{n}\mathrm{tr}\left((\sigma_x^{-2}K_{zz} + I)^{-1}\right)$ which is strictly positive for $\sigma_x^2 > 0$, and increasing with σ_x^2.
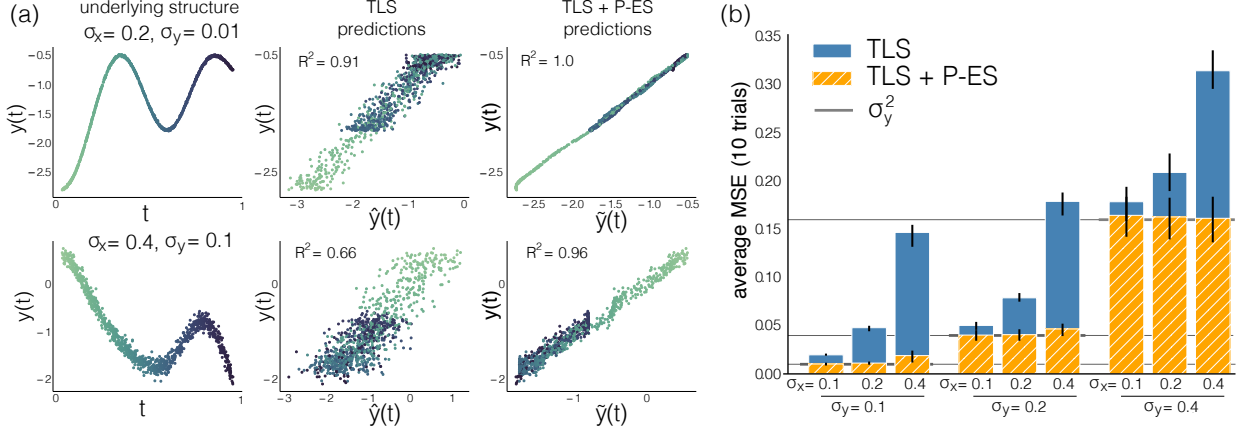
Esther Rolf, Michael I. Jordan, Benjamin Recht

Figure 1: Simulation results. (a) Two examples of structure in z (left column), where a TLS estimator recovers structure (middle column), but is improved upon using P-ES (right column). (b) Aggregate performance over different noise parameters for unsmoothed and P-ES estimates, compared to a lower bound of σ_y^2 for any linear smoother. Vertical black lines show min and max over 10 trials.

4 EXPERIMENTS

We first use simulated experiments to study situations in which smoothing is beneficial and to demonstrate that a simple instantiation of Eq. (2) achieves close to optimal accuracy in these settings. We then apply P-ES to predictions on real-world datasets with temporal and spatial structure: human-pose prediction in video (Sec. 4.2) and house-price prediction over space (Sec. 4.3). P-ES improves performance of all predictors we consider, including some that already incorporate locality. P-ES compares favorably to statistical smoothing and SSL methods, both in predictive accuracy and computation time. As a simple local-averaging weight matrix, all experiments use as W the Nadaraya-Watson smoothing matrix with squared exponential kernel on t, where $D_{ij}(t;\sigma) = e^{-\frac{1}{2\sigma^2}\|t_i - t_j\|_2^2}$:

$$S_c(t;\sigma) = c \cdot \operatorname{diag}^{-1}\left(D\vec{1}\right)D + (1-c)\cdot I . \quad (4)$$

4.1 Simulations

We return to the distribution defined in Eq. (3) in Example 3.1, where the processes $x(t)$ and $y(t)$ are influenced by a third 'hidden' process $z(t)$. We now make a specific assumption for the covariance of z:

$$\vec{z} = \mathcal{N}\left(\vec{0}, \Sigma(t)\right), \quad \Sigma_{ij}(t) = e^{-\frac{1}{2\sigma_z^2}(t_i - t_j)^2} .$$

We take $t = [0, 1/n, 2/n, \ldots, (n-2)/n, (n-1)/n]$ with $n = 2000$, and $\sigma_z = 0.2$. Half of the points are chosen at random to form a training set from which we learn the total least squares (TLS) estimator $\hat{y}(x)$. The remaining 1000 points are used to evaluate performance with and without P-ES. To show the expressiveness of

the matrix S_c, in simulations we pick the parameters c, σ of S_c so as to maximize performance on the evaluation set. In Sections 4.2 and 4.3, we pick parameters on a validation set before applying to a holdout set.

Figure 1(a) shows the process of P-ES as local variance reduction. Each row shows a different setting of σ_x, σ_y. The leftmost column shows observed labels $y(t)$ as a function of indices t. The middle and right columns show the TLS predictions, without and with P-ES, respectively. Errors in $\hat{y}$ that are made in the horizontal axis are reducible by smoothing, as $S_c(t)$ gives more weight to pairs closer in t (similar hue in Figure 1(a)). The smoothed predictions $\tilde{y}$ exhibit a similar structure to the original predictions, with significantly reduced horizontal error bands. The difference in the performance of the TLS estimator with and without P-ES (Figure 1(b)) indicates that P-ES reduces prediction errors that are uncorrelated with the index variable t.

4.2 Human Pose Prediction in Video

Recent work has shown that improvements in human pose estimation (Zhang et al., 2019; Kanazawa et al., 2019; Dabral et al., 2018) and object detection and classification in videos (Prest et al., 2012; Yucer et al., 2015; Zhu et al., 2017) can be obtained by encoding temporal consistency as part of a larger predictive pipeline. The intuition is that exploiting continuity of motion over video frames can reduce the noise in per-frame predictions. For example, a recent state-of-the-art method for pose estimation (Kanazawa et al., 2019) learns both a temporal encoder and temporal human dynamics as part of the predictive pipeline. In the following experiment, we apply P-ES to predictions from this model as well as to predictions from a per-frame baseline model (Kanazawa et al., 2018).

Table 1: Holdout set performance for human pose estimation in video. Arrows indicate the direction of desired performance; bold numbers indicate the best performance for each metric. For all metrics, P-ES predictions (italicized methods) have the best performance of all methods considered. Other methods are attributed as "temporal, temporal + dynamics:" Kanazawa et al. (2019), "per-frame:" Kanazawa et al. (2019, 2018).

	3DPW				Penn Action	
method	PCK ↑	MPJPE ↓	PA-MPJPE ↓	Acc. Err. ↓	PCK ↑	Accel ↓
per-frame	84.06	129.95	76.68	37.41	73.17	79.91
per-frame, with P-ES	84.46	128.44	**75.84**	20.46	73.74	48.22
temporal	82.59	139.19	78.35	15.15	71.16	29.30
temporal + dynamics	86.37	127.08	80.05	16.42	77.88	29.66
temporal + dynamics with P-ES	**86.57**	**126.14**	79.73	**8.14**	**78.07**	**4.96**

We use the same validation and holdout splits for the 3D Poses in the Wild (3DPW) (von Marcard et al., 2018) and the Penn Action datasets (Zhang et al., 2013) as in Kanazawa et al. (2019). Before testing results on the holdout set, smoothing parameters were chosen from $\sigma \in [0.5, 1, 2, 3, 4]$ frames, and $c \in [0.0, 0.2, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]$ to maximize average validation accuracy measured by the key-point accuracy (PCK) metric (see Appendix B for details).

Holdout test set performance is given in Table 1. PCK is an accuracy metric on key-points, MPJPE and PA-MPJPE measure error over predicted pose joints, and acceleration error penalizes high acceleration predictions (see Appendix B for a discussion of the metrics). While we optimized according to PCK, P-ES improves performance in both models, across all metrics.

Smoothing confers greater gains in the time-agnostic per-frame model than the temporal dynamics model. Interestingly, the "temporal model" without human dynamics does worse in almost all metrics than the "per-frame" model that ignores frame number. This underscores our motivation, that temporal information must be encoded with care, as well as our claim that P-ES is a suitable baseline for such tasks.

The optimal hyperparameter pairs chosen are given in Table 2. For both models, the optimal σ was around 2 frames, but the optimal c for the per-frame predictions was much smaller for the per-frame model (avg. 0.45) than for the model that already incorporated temporal structure (avg. 0.9). This may be because predictions for the temporal model are smoother, so that we do not alter the signal in predictions as much with P-ES.

In summary, P-ES confers performance gains to both the per-frame model and the more accurate temporal and dynamics model, showing that P-ES can improve performance even when the base estimator is a complex model incorporating locality.

Table 2: Optimal hyperparameter values on validation set per predictor for both datasets.

	per-frame		temporal + dynamics	
	3DPW	PA	3DPW	Penn Action
σ	2	3	2	2
c	0.5	0.4	0.8	1.0

4.3 Predicting House Price from Attributes

The usefulness of applying semi-parametric techniques merging feature-based prediction and spatial regularization in predicting house prices has been documented from many perspectives (see, e.g., Caplin et al., 2008; Dubin, 1998; Clapp et al., 2002; Can, 1992). This motivates house price prediction as a domain in which to compare the performance of P-ES and other methods exploiting spatial consistency. Using data on house sales from the Zillow Transaction and Assessment Database (ZTRAX) (Zillow, 2018), we first demonstrate the effectiveness of P-ES on various machine learning regression methods (Figure 2), and then in comparison to standard semi-supervised learning techniques (Table 3).

In this experiment we predict sale prices y of single family homes, given features x about the homes (e.g., number of bedrooms, year of home sale, etc.). Location t is the latitude and longitude of the homes. After preprocessing (see Appendix B), the dataset contains roughly 600,000 home sales at unique locations. We test three diverse machine learning models: ridge regression, random feature regression (Rahimi and Recht, 2009), and gradient boosted decision trees (XGB) (Chen and Guestrin, 2016), with and without post-estimation smoothing. For all three resulting models, we chose parameters jointly over the model parameters and P-ES parameters to maximize validation set accuracy, measured in R^2, the percent of label variation explained by the predictions.

Table 3: Comparison with nonparametric and semi-supervised methods, for 10 random trials with train, validation, and holdout sets of size $n = 10,000$.

| method	model function (f) or post-processing (pp) form	average holdout accuracy mean (std) in r^2	average runtime mean (std) in secs
kernel smoothing	$f(t, y)$	0.277 (0.185)	20.6 (0.2)
GPR (Kriging)	$f(t, y)$	0.386 (0.011)	1336.1 (4.6)
LapRLS (Belkin et al., 2006)	$f(x, t, y)$	0.452 (0.012)	1683.6 (12.4)
XGB	$f(x, y)$	0.458 (0.014)	7.8 (0.1)
XGB + shrinkage	$pp(t, \widehat{y}_{\mathrm{XGB}})$	0.457 (0.014)	+ 0.0 (0.0)
XGB + P-ES	$pp(t, y, \widehat{y}_{\mathrm{XGB}})$	0.526 (0.015)	+ 27.0 (0.2)
HEM (Zhu et al., 2003)	$f(t, y, \widehat{y}_{\mathrm{XGB}})$	0.544 (0.015)	+ 898.6 (4.8)
HEM (Zhu et al., 2003) + P-ES	$pp\left(t, y, f(t, y, \widehat{y}_{\mathrm{XGB}})\right)$	0.546 (0.015)	+ 287.8 (8.0)
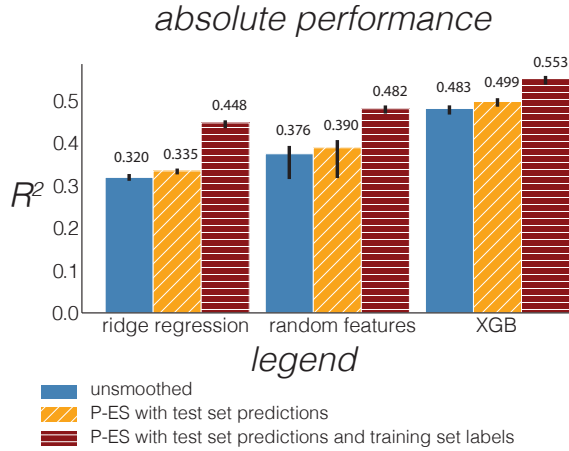
Figure 2: P-ES performance for various base predictors. Bars denote average performance; vertical black lines show min and max over 10 trials. All six average relative differences (unsmoothed - smoothed) are positive with p-value $< 1e^{-4}$, those that include the training set have p-value $< 1e^{-7}$.

Figure 2 shows the holdout test set performance of smoothed and unsmoothed models for the three machine learning algorithms. Training, validation, and test sets are of size $n = 20,000$ each. We performed this experiment over 10 random data draws. Smoothing is performed with respect to just the predictions, as well as with respect to the concatenated training set labels and test set predictions (validated on training set labels and validation set predictions). See Appendix B for details on hyperparameter settings.

For all three methods, applying P-ES with the test set predictions improves accuracy over the original predictions. Smoothing with the training points, in the spirit of semi-supervised learning, boosts accuracy further, as we might expect since there is no estimation error for the training labels. Figure 5 in Appendix B.3 shows a similar trend holds across sample set sizes (n).

Table 3 shows a comparison to alternative methods for reducing variance or inducing spatial consistency: kernel smoothing based only on the training set labels (without predictions), Gaussian process regression (GPR), Laplacian regularized least squares (LapRLS), a variance reducing shrinkage estimator ($S = \delta \cdot (\frac{1}{n} \vec{1}\vec{1}^{\top}) + (1 - \delta)I$), and Gaussian harmonic energy minimization (HEM) (see Sec. 2).

Timing results underscore that P-ES is a fast way to incorporate spatial structure (it incurs an $\mathcal{O}(n^2)$ additional runtime as opposed to $\mathcal{O}(n^3)$ for GPR, LapRLS, and HEM). The high variance in performance of kernel smoothing alone may be explained by inherent difficulties in choosing hyperparameters in semi-supervised settings, as discussed by Oliver et al. (2018). Accuracy of post-processing with P-ES is within 1.2 standard deviations of the HEM method which takes roughly 30× as long to run in this instance over the chosen set of hyperparameters (see Appendix B).

Runtimes for post-processing procedures are reported as the additional time compared to not running the post-processing procedure (on average, computing P-ES predictions takes 27 seconds on top of the 7.8 seconds to run XGB over multiple hyperparameter configurations). The runtime numbers reported in Table 3 are for solving the exact HEM and LapRLS problems, using the inverse and with as much shared computation as possible. We omit experimental comparison to LGC (Zhou et al., 2004a,b) as the adaption from multi-class classification and ranking problems to regression problems is nontrivial, but we note that it is an iterative metho where each iteration is $\mathcal{O}(n^2)$. The first iteration of the LGC algorithm is very similar to P-ES, so that similarity of accuracy of HEM and P-ES with the smoothing matrix defined in Eq. (4) suggests that a one-iteration approximation of these algorithms can be sufficient in some cases.

The last line in table Table 3 confirms that we get a very small increase in accuracy by smoothing the
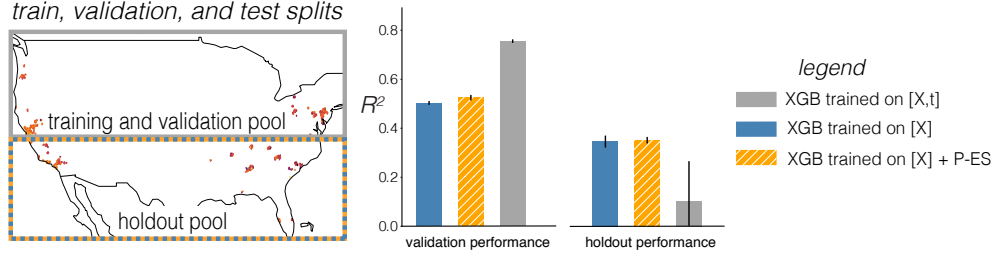
Figure 3: In-sample (unsmoothed) validation performance and out-of-sample holdout performance for different methods of incorporating spatial index variables. Lines show min and max over 10 trials.

best SSL method; this is consistent with our understanding that this smoothing operator acts similarly to the graph-defined HEM operator. The reduction in computation to apply P-ES significantly reduces the barrier to comparing to this family of algorithms as a baseline. Additionally, the computational speed of P-ES makes it much easier to explore different choices of the weight matrix $W(t; \sigma)$ from which to form the smoothing matrix; we consider this future work since such a matrix will likely be domain-specific.

We conclude our experiments with a final example of the danger of incorporating index variables with unique characteristics as predictive features. We compare the out-of-sample generalizability of a model $\widehat{y}(x, t)$ trained on the concatenated set of home attributes and geographic location as features with that of the model from previous experiments $\widehat{y}(x)$ trained only on attributes with smoothing applied.

Figure 3 shows the results of these two approaches for a spatial extrapolation experiment, where a training set and validation set of size $20,000$ each are sampled from the northern U.S., and a holdout set of size $20,000$ is sampled from a disjoint southern segment.[1]

We might expect that incorporating latitude and longitude as features could cause the first model to overfit, whereas distribution shift in neighborhood structure might negatively impact the learning of parameters for P-ES. Indeed, while the XGB predictor trained with locations and attributes (XGB trained on $[X, t]$) has better validation performance than the location-agnostic predictor (XGB trained on $[X]$), it performs much worse on the holdout set (0.10 vs 0.35 average R^2). However, applying P-ES does not degrade performance (0.002 average R^2 increase). In this scenario, the distribution shift with respect to t is so severe that it is possible to overfit by using t as a feature. Incorporating structure in t with P-ES, on the other hand, is much more robust to this distribution shift.

5   CONCLUSION

We introduce post-estimation smoothing as a method for incorporating structural indices like time or location as valuable information sources in machine learning predictions. Theory and experiments underscore that P-ES is an effective and robust way to incorporate structured index variables in prediction, at much less cost than traditional semi-supervised methods.

The performance of P-ES depends on the accuracy of the original predictions. If predictions $\hat{y}$ are very far from y, smoothing is unlikely to remedy this. While decoupling smoothing from the original prediction may be limiting, we have shown that it can be advantageous when viewing P-ES as a diagnostic method or a baseline with which to compare more complex methods.

This work opens a door for extensions to applications where index variables satisfy less physical notions of distance (e.g., word embeddings), and to analysis characterizing when decoupling local consistency and prediction can be close to the optimal integrated approach. Future work could also consider multivariate labels y with correlation among their elements. Lastly, it will likely be worthwhile to investigate more structured weight matrices W (other than the Nadaraya-Watson estimator) which form the basis of the smoothing matrix S. Due to the decoupling of smoothing from the original prediction, practitioners can try domain-specific weight matrices with marginal extra cost.

When the goal is to obtain accurate predictors, no data should be overlooked. However, index variables such as time and space should be incorporated with care. We propose that post-processing is a natural and effective way to utilize this structure, and show it is robust to different tasks, predictors, and sampling patterns.

Acknowledgements

---

[1]A figure with all possible data pipelines including smoothing $[X, t]$ is given in Appendix B.4.

References

Olena Babak and Clayton V Deutsch. Statistical approach to inverse distance interpolation. *Stochastic Environmental Research and Risk Assessment*, 23(5):543–553, 2009.

Sudharshan Chandra Babu. Human-pose-estimation-101. https://github.com/cbsudux/Human-Pose-Estimation-101, 2019.

Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *Advances in Neural Information Processing Systems*, pages 3365–3373, 2014.

Sudipto Banerjee, Alan E Gelfand, Andrew O Finley, and Huiyan Sang. Gaussian predictive process models for large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(4):825–848, 2008.

Mikhail Belkin and Partha Niyogi. Semi-supervised learning on riemannian manifolds. *Machine learning*, 56(1-3):209–239, 2004.

Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7(Nov):2399–2434, 2006.

Mikhail Belkin, Alexander Rakhlin, and Alexandre B Tsybakov. Does data interpolation contradict statistical optimality? *arXiv preprint arXiv:1806.09471*, 2018.

Ayse Can. Specification and estimation of hedonic housing price models. *Regional science and urban economics*, 22(3):453–474, 1992.

Andrew Caplin, Sumit Chopra, John Leahy, Yann LeCun, and Trivikraman Thampy. Machine learning and the spatial structure of house prices and housing returns. 2008.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM, 2016.

John M Clapp, Hyon-Jung Kim, and Alan E Gelfand. Predicting spatial patterns of house prices using lpr and bayesian smoothing. *Real Estate Economics*, 30(4):505–532, 2002.

Fabio Gagliardi Cozman and Ira Cohen. Unlabeled data can degrade classification performance of generative classifiers. In *Flairs conference*, pages 327–331, 2002.

Rishabh Dabral, Anurag Mundhada, Uday Kusupati, Safeer Afaque, Abhishek Sharma, and Arjun Jain. Learning 3d human pose from structure and motion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 668–683, 2018.

Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. Are we really making much progress? a worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 101–109. ACM, 2019.

Robin A Dubin. Predicting house prices using multiple listings data. *The Journal of Real Estate Finance and Economics*, 17(1):35–59, 1998.

Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*, pages 529–536, 2005.

László Györfi, Michael Kohler, Adam Krzyzak, and Harro Walk. *A distribution-free theory of nonparametric regression*. Springer Science & Business Media, 2006.

Sabine Van Huffel. *The Total Least Squares Problem: Computational Aspects and Analysis (Frontiers in Applied Mathematics)*. Society for Industrial and Applied Mathematics, 1991. ISBN 0898712750. URL https://www.xarg.org/ref/a/0898712750/.

Neal Jean, Sang Michael Xie, and Stefano Ermon. Semi-supervised deep kernel learning: Regression with unlabeled data by minimizing predictive variance. In *Advances in Neural Information Processing Systems*, pages 5322–5333, 2018.

Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018.

Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Yu-Feng Li and Zhi-Hua Zhou. Towards making unlabeled data never hurt. *IEEE transactions on pattern analysis and machine intelligence*, 37(1):175–188, 2014.

George Y Lu and David W Wong. An adaptive inverse-distance weighting spatial interpolation technique. *Computers & geosciences*, 34(9):1044–1055, 2008.

Horia Mania, Aurelia Guy, and Benjamin Recht. Simple random search provides a competitive approach to reinforcement learning. *arXiv preprint arXiv:1803.07055*, 2018.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics, 2009.

Elizbar A Nadaraya. On estimating regression. *Theory of Probability & Its Applications*, 9(1):141–142, 1964.

Avital Oliver, Augustus Odena, Colin Raffel, Ekin D Cubuk, and Ian J Goodfellow. Realistic evaluation of semi-supervised learning algorithms. 2018.

Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari. Learning object class detectors from weakly annotated video. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3282–3289. IEEE, 2012.

Ali Rahimi and Benjamin Recht. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in neural information processing systems*, pages 1313–1320, 2009.

Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data programming: Creating large training sets, quickly. In *Advances in neural information processing systems*, pages 3567–3575, 2016.

H Schneeweiss. Consistent estimation of a regression with errors in the variables. *Metrika*, 23(1):101–115, 1976.

Jeffrey S. Simonoff. *Smoothing Methods in Statistics (Springer Series in Statistics)*. Springer, 1998.

ISBN 0387947167. URL `https://www.amazon.com/Smoothing-Methods-Statistics-Springer/dp/0387947167?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=0387947167`.

Aarti Singh, Robert Nowak, and Xiaojin Zhu. Unlabeled data: Now it helps, now it doesn't. In *Advances in neural information processing systems*, pages 1513–1520, 2009.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, pages 1195–1204, 2017.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation (Springer Series in Statistics)*. Springer, nov 2010. ISBN 9781441927095. URL `https://www.xarg.org/ref/a/1441927093/`.

Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, sep 2018.

Matt P Wand and M Chris Jones. *Kernel smoothing*. Chapman and Hall/CRC, 1994.

Geoffrey S Watson. Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 359–372, 1964.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT Press Cambridge, MA, 2006.

Kaan Yucer, Oliver Wang, Alexander Sorkine-Hornung, and Olga Sorkine-Hornung. Reconstruction of articulated objects from a moving camera. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 28–36, 2015.

Jason Y. Zhang, Panna Felsen, Angjoo Kanazawa, and Jitendra Malik. Predicting 3d human dynamics from video. In *International Conference on Computer Vision (ICCV)*, 2019.

Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2248–2255, 2013.

Dengyong Zhou, Olivier Bousquet, Thomas N Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *Advances in neural information processing systems*, pages 321–328, 2004a.

Dengyong Zhou, Jason Weston, Arthur Gretton, Olivier Bousquet, and Bernhard Schölkopf. Ranking on data manifolds. In *Advances in neural information processing systems*, pages 169–176, 2004b.

Xiaojin Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.

Xiaojin Zhu, Zoubin Ghahramani, and John D Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International conference on Machine learning (ICML-03)*, pages 912–919, 2003.

Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *CVPR*, volume 1, page 3, 2017.

Zillow. ZTRAX: Zillow Transaction and Assessor Dataset, 2018. URL `http://www.zillow.com/ztrax/`.