SUPERVISED PRINCIPAL COMPONENT ANALYSIS VIA MANIFOLD OPTIMIZATION

Alexander Ritchie[1], *Clayton Scott*[1,2], *Laura Balzano*[1], *Daniel Kessler*[2,3], *Chandra S. Sripada*[3,4]

University of Michigan
EECS[1], Statistics[2], Psychiatry[3], Philosophy[4]
correspondence: `aritch@umich.edu`

ABSTRACT

High dimensional prediction problems are pervasive in the scientific community. In practice, dimensionality reduction (DR) is often performed as an initial step to improve prediction accuracy and interpretability. Principal component analysis (PCA) has been utilized extensively for DR, but does not take advantage of outcome variables inherent in the prediction task. Existing approaches for supervised PCA (SPCA) either take a multi-stage approach or incorporate supervision indirectly. We present a manifold optimization approach to SPCA that simultaneously solves the prediction and dimensionality reduction problems. The proposed framework is general enough for both regression and classification settings. Our empirical results show that the proposed approach explains nearly as much variation as PCA while outperforming existing methods in prediction accuracy.

1. INTRODUCTION

Consider the problem of linear prediction of a set of q outcome variables $y_1, y_2, \dots, y_q$ from a set of p predictor variables $x_1, x_2, \dots, x_p$ based on n observations, i.e., in terms of matrices, predicting $Y \in \mathbb{R}^{n\times q}$ from $X \in \mathbb{R}^{n\times p}$. In the interest of both interpretability and prediction error, it is often desirable to perform dimension reduction (DR) to learn a low-dimensional representation of X. Such a low-dimensional representation enhances prediction accuracy by combating irrelevant directions and the curse of dimensionality (1; 2). At the same time, DR aids interpretation of X by concisely representing the structure in X. Perhaps the most widely used DR technique is Principal Component Analysis (PCA), which finds directions in X of maximal variation. PCA has become especially popular in neuroscience and other fields where high dimensional data abound, owing to ease of computation and geometric interpretability (3; 4).

PCA is unsupervised in that it does not leverage the training output Y. The goal of supervised PCA (SPCA) is to learn a low-dimensional representation that explains much of the total variation of X while also being predictive of Y. Such a low-dimensional representation should naturally lead to improved prediction accuracy, and could also offer increased interpretability by discovering components that explain the output variables. In general, the two goals of SPCA are in competition. Therefore, some notion of trading off between these two goals is desirable for any SPCA approach.

Several approaches to SPCA have been proposed (5; 6; 7; 8). These works often adopt a multi-stage approach, first performing supervised dimensionality reduction before learning a predictor. Existing methods also lack a means of specifying the trade-off between explaining variation in X and prediction on Y.

This work proposes a straightforward approach to SPCA that jointly solves the dimensionality reduction and prediction problems. We formulate SPCA as optimization of an objective function over the Grassmann manifold; the proposed objective is a weighted sum of the PCA objective and an empirical risk associated with the prediction problem. This formulation is general enough to handle classification and regression problems by changing the loss. We evaluate the proposed approach using squared error loss and find that it dominates existing methods for SPCA in terms of both variation explained and prediction accuracy. Omitted for brevity, we have also explored using the logistic loss and found results to be similarly compelling.

2. BACKGROUND

PCA has been used extensively for dimensionality reduction (DR) due to its effectiveness, ease of computation and interpretation. However, PCA is unsupervised, and therefore DR via PCA may not be optimal for classification or regression tasks. Existing approaches for SPCA either take a multi-stage approach, lack a direct means of specifying the trade-off between learning data structure and predictivity, or both. The method we propose governs this trade-off directly through regularization, at the expense of adding a tuning parameter. We begin by describing some classical supervised DR methods that attempt to solve a similar problem to the one we have described. We then describe some modern methods that attempt to supervise PCA more directly.

Linear discriminant analysis (LDA) is arguably the canonical example of supervised dimensionality reduction in the classification setting. LDA finds a DR of X such that separability among classes is maximized. Though it may seem that LDA is generally preferable to PCA in this setting, this has been shown not always to be the case, especially when the number of training samples is small (9). Another shortcoming of LDA is that the number of projection dimensions is limited to $c - 1$, where the number of classes in the dataset is c.

Perhaps the simplest approach to SPCA is principal component regression (PCR), which simply performs ordinary least squares regression on data that has been reduced by PCA.

Partial Least Squares (PLS) regression is an iterative procedure that attempts to find directions in the input space that account for a high amount of variation of the input data, but are also highly correlated with the dependent variables. In this sense, the goal of PLS is very similar to the goal of supervised PCA. However, it should be noted that well known PLS procedures do not provide a means of trading off between the two competing objectives. Furthermore, PLS tends to put preference on directions that account for high variation rather than high correlation, causing it to behave similarly to PCR (10).

To our knowledge, the earliest of the modern SPCA approaches

978-1-7281-0708-0/19/$31.00 ©2019 IEEE

is due to (5). This approach, herein referred to as Bair's method, is a simple two stage procedure: 1) perform feature selection based on univariate regression coefficients, and 2) perform PCA on the data matrix consisting only of the selected features. From here the learned principal components are used for prediction purposes. One shortcoming of this method is that it only applies to regression problems with a single dependent variable. Another is that this method takes a two stage approach to the prediction problem. Therefore, it is not straightforward how the number of features selected or the threshold for selecting them governs the trade-off between prediction and variation explained. Recent work elaborates on Bair's method by iteratively performing a similar procedure, only taking the first supervised principal component, subtracting the variation explained by this principal component from X, and repeating (8). This method is referred to as Iterative Supervised Principal Component Analysis (ISPCA).

Supervised probabilistic principal component analysis (7) (SPPCA) uses the probabilistic principal component analysis (PPCA) framework to approach the SPCA problem. One substantial drawback of this approach is that it places the same amount of emphasis on the dependent and independent variables. Therefore SPPCA does not offer a way to trade-off between prediction and variation explained. It is also sensitive to the relative dimensions of $\boldsymbol{x}$ and $\boldsymbol{y}$.

The method of (6), herein referred to as Barshan's method, approaches the supervised PCA problem by means of the Hilbert-Schmidt Independence Criterion (HSIC), a measure of independence in reproducing kernel Hilbert spaces (RKHSs). In a universal RKHS, the HSIC of two random variables is zero if and only if the random variables are independent. Barshan's method attempts to maximize an empirical measure of the HSIC. The connection to PCA is drawn through the similarity of the empirical HSIC and the trace maximization formulation of PCA. It can also be shown that their approach reduces to PCA if there is no supervisory data available. This approach again leads to a two stage approach for the prediction problem. Due to the structure of the objective, there is no way to trade-off between learning the structure in X and its predictivity of Y.

3. SUPERVISED PCA AS MANIFOLD OPTIMIZATION

Our approach to SPCA is to solve an optimization problem whose objective is a weighted sum of the PCA objective and an empirical risk associated with the prediction problem. This makes the trade-off between prediction and variation explained explicit, at the expense of adding a tuning parameter. In particular, we propose to solve

$$\begin{aligned} &\underset{L,\beta}{\text{minimize}} \ \sum_{i=1}^{n} \ell(\boldsymbol{y}_i, L\boldsymbol{x}_i, \beta) + \lambda \|X - XL^TL\|_F^2 \qquad (1) \\ &s.t. \ LL^T = I_k, \end{aligned}$$

where $\ell(\cdot)$ is a loss function in terms of the dependent variables and the dimension-reduced data, n is the number of observations, k a user specified dimension of the subspace to be learned, $\lambda > 0$ is a trade-off parameter, and the remaining quantities are described in Table 1. The constraint on L^T in (1) is over the Stiefel manifold, i.e., the set of all matrices with orthonormal columns.

The key feature of the proposed approach is that the predictor, parametrized by β, operates directly on the low-dimensional representation $L\boldsymbol{x}_i$. The variable L ties the two terms together and enables simultaneous DR and prediction, with λ affecting a trade-off between these two goals. We also remark that, unlike other approaches to high-dimensional prediction, there is no need to regularize β because the predictor acts on a low-dimensional space.

In the following sections we explore applications of the proposed general methodology to regression by taking $\ell(\cdot)$ to be the squared error loss.

Table 1: Description of Key Variables

VARIABLE	DESCRIPTION
$\underset{n\times p}{X}$	Data matrix
$\underset{n\times q}{Y}$	Dependent variables
$\underset{k\times p}{L}$	Basis for the learned subspace
$\underset{n\times k}{XL^T}$	Dimension reduced form of X
$\underset{k\times q}{\beta}$	Learned coefficients for prediction

3.1. Grassmannian Constraints for Linear Prediction

In the case of linear prediction, i.e., where the predicted output depends on $XL^T\beta$, the value of the objective function (1) only depends on the subspace spanned by the rows of L. To see this, imagine applying the same rotation to L and β. In this problem setting it is natural to consider the Grassmann manifold instead of the Stiefel manifold. The Grassmannian $\mathcal{G}(p, k)$ is the set of k dimensional subspaces in $\mathbb{R}^p$. In general many of the necessary operations for manifold optimization can be more easily performed on the Grassmannian, e.g., projection to the tangent space and geodesic step. For this reason we solve optimization problems presented in subsequent sections over the Grassmannian. It should be noted that even though points on the Grassmannian are subspaces, numerical algorithms require a representation of the subspace to be stored. Here, and in general, these representations are taken to be matrices with orthogonal columns or rows that span the subspace.

4. SUPERVISED PCA FOR LINEAR REGRESSION

In this section we consider the case where $\ell(\boldsymbol{y}_i, L\boldsymbol{x}_i, \beta) = \|\boldsymbol{y}_i - \beta^T L\boldsymbol{x}_i\|_2^2$ is the squared error loss. The resulting optimization problem, which we call least squares PCA (LSPCA), is as follows:

$$\begin{aligned} &\underset{L,\beta}{\text{minimize}} \ \|Y - XL^T\beta\|_F^2 + \lambda \|X - XL^TL\|_F^2 \qquad (2) \\ &s.t. \ LL^T = I_k. \end{aligned}$$

Note the problem is not convex in L, and L is constrained to the non-convex Stiefel manifold which, as noted previously, can be relaxed to a Grassmannian constraint. However, for any fixed L the optimal β is just the solution to the ordinary least squares problem,

$$\hat{\beta} = (XL^T)^+Y, \qquad (3)$$

where $(\cdot)^+$ denotes the Moore-Penrose pseudoinverse.

This immediately suggests substituting $\hat{\beta}$ in (3) to simplify (2) to an optimization problem with a single variable:

$$\begin{aligned} &\underset{L}{\text{minimize}} \ \|Y - XL^T(XL^T)^+Y\|_F^2 + \lambda \|X - XL^TL\|_F^2 \qquad (4) \\ &s.t. \ LL^T = I_k. \end{aligned}$$

Gradient-based optimization methods on the Grassmannian are well studied in the literature (11; 12). A gradient step on the Grassmannian for a given objective function can be calculated directly from its Euclidean gradient, or Frechét derivative, via a projection onto the tangent space at the current point.

4.1. A Gradient Algorithm for LSPCA

Elimination of β in (2) gives rise to difficulties in calculating the Frechét derivative of the objective function. We can be assured the derivative exists under the assumption that XL^T has locally constant rank over L, i.e., $\operatorname{rank}(X\tilde{L}^T) = \operatorname{rank}(XL^T)$ for all $\tilde{L} \in \Omega \subset \mathcal{R}^{k\times p}$ for some open set Ω containing L (13). We assume that the descent step of (2) does not decrease the rank of XL^T. Therefore, if we seed the proposed algorithm with L_0 such that $\operatorname{rank}(XL_0) = k$, every iterate $X_{L_i}^T$ will have rank k as well. If we take the rows of L_0 to be the first k principal components of X, and $\operatorname{rank}(X) \geq k$, clearly the preceding condition is satisfied. This implies the LSPCA objective is differentiable at every iterate of a gradient descent type algorithm (assuming proper stepsize), if the algorithm is seeded using the first k principal components of X. The derivative of the LSPCA objective function is therefore

$$\frac{\partial f}{\partial L} = -2(XL^T)^{+}YY^TP_{XL^T}^{\perp}X - 2\lambda LX^TX, \qquad (5)$$

where $P_{XL^T}^{\perp}$ is the projection matrix onto the orthogonal compliment of the span of XL^T.

The key notion for applying gradient based optimization methods to manifold optimization is that of the retraction. At any iteration, the gradient step we wish to take lies in the tangent space of the manifold. However, since the Grassmannian is not linear, the gradient step does not reside on the manifold. To account for this, a retraction is used to take a step in the direction of the negative gradient so that the resulting point lies on the manifold. If a closed form expression is available for moving between two points on the manifold while remaining on the manifold, it is possible to take the update step directly. This is called a geodesic step. For the Grassmannian, an expression for the geodesic step is given in Eqn. 2.65 of (11), which we utilize in line 7 of Algorithm 1.

A conjugate gradient method on the Grassmannian is described by (11). We choose to implement the gradient descent algorithm described in (12), in which step size is chosen by Armijo style backtracking. Gradient descent over compact Riemannian manifolds with Armijo style backtracking converges to first order stationary points asymptotically, provided the objective is differentiable (14). Algorithm 1 satisfies these conditions. However, additional properties must be verified to guarantee convergence within a finite number of iterations (12). Note that the convergence criteria of Algorithm 1 is $\|\operatorname{grad}(L_t)\|_F < \epsilon$. Here $\operatorname{grad} f(L)$ is the Riemannian Gradient.

The method is shown in Algorithm 1, and has computational complexity $\mathcal{O}(kn(p+q) + kn^2 + k^2n + k^3)$ at each iteration. Since k is often taken to be relatively small in DR problems, e.g., $k \in \{2, 3\}$ for data visualization, the cubic complexity in k is not a major concern. Each iteration can be summarized as follows:

1. Calculate the Euclidean gradient at the current iterate L.
2. Project the negative Euclidean gradient onto the tangent space of the Grassmannian at the current iterate, to obtain the Riemannian Gradient.
3. Take the singular value decomposition (SVD) of the resulting $p \times k$ matrix.
4. Update L by taking a geodesic step along the Grassmannian, with stepsize chosen according to Armijo line search.

Algorithm 1 Gradient Descent for LSPCA

1) Column center X and Y

2) Choose the reduction dimension k, and regularization parameter λ

3) Generate an initialization, L_0 via PCA

```
 1: procedure LSPCA($X, Y, L_0, \lambda, k$)
 2:     $t = 0$
 3:     while $\|\operatorname{grad}(L_t)\|_F < \epsilon$ do
 4:         Calculate $\nabla_t = \frac{\partial f}{\partial L}|_{L=L_t}$ via equation (5)
 5:         $\operatorname{grad}(L_t)^T = (I_p - L_t^T L_t)\nabla_t^T$
 6:         $U_t, \Sigma_t, V_t = \operatorname{SVD}(-\operatorname{grad}(L_t)^T)$
 7:         $L_{t+1}^T = L_t^T V_t \cos(\eta_t\Sigma_t)V_t^T + U_t \sin(\eta_t\Sigma_t)V_t^T$    ▷
    Where $\eta_t$ is a step size chosen by Armijo backtracking line-search.
 8:         $t \leftarrow t + 1$
 9:     end while
10:     $Z = XL_t^T$                          ▷ Generate the reduced data.
11:     return $Z, L_t$
12: end procedure
```

4.2. Selecting λ

In most settings where a parameter must be tuned by cross validation, there is a clear metric by which each value of the parameter can be judged, i.e., squared error or classification accuracy on a holdout dataset. In the case of LSPCA, this is not the case since the objectives have two competing terms. We propose the following convention for selecting lambda in the regression case:

$$\hat{\lambda} = \underset{\lambda}{\operatorname{argmin}} \quad \frac{\sum_{i=1}^{n} \ell(\boldsymbol{y}_i, L_\lambda \boldsymbol{x}_i, \beta_\lambda)}{\|Y\|_F^2} + \frac{\|X - XL_\lambda^T L_\lambda\|_F^2}{\|X\|_F^2},$$

where L_λ, β_λ are the optimal L, β obtained from solving (1) with a parameter of λ. The idea is that when the two terms in the objective are normalized, they will be of similar scale. The λ which gives the lowest value of the normalized sum should perform approximately best on both terms of the objective.

5. EXPERIMENTAL RESULTS

We compare performance on three real-world regression datasets. For each experiment a random 80/20 train/test split is used. For methods that require parameter tuning, 20% of the training data is held out for parameter tuning via grid search.

In order to evaluate LSPCA we compare performance on common datasets, specifically those used by competing methods in previous work. To compare methods in the prediction task we use squared error for regression problems. To compare how effectively each method has learned underlying structure of the data X, we use proportion of variation explained which is frequently used to evaluate the quality of dimensionality reduction by unsupervised PCA. We calculate proportion of variation explained as

$$\text{var}_{\text{ex}} = \frac{\|XL^T\|_F^2}{\|X\|_F^2} \qquad (6)$$

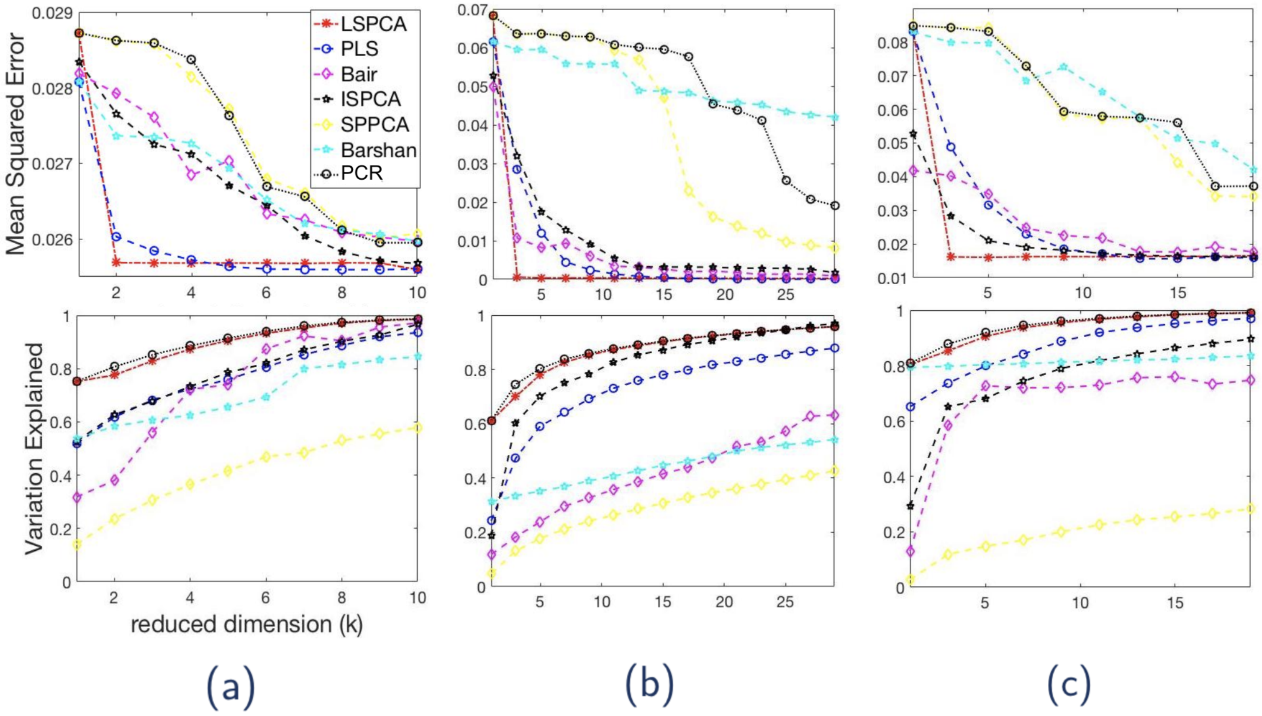

Fig. 1: Mean squared error and variation explained vs. reduced dimension for each method on various datasets. (a) Parkinsons (b) Music (c) Residential

5.1. Regression Experiments

We compare performance on three real-world regression datasets, the Music dataset[1] ($n = 1059, p = 118$) of (15), the Residential dataset[1] ($n = 372, p = 105$) of (16), and the Parkinsons telemonitoring dataset[1] ($n = 5875, p = 20$) of (17).

Figure 1 shows average performance of each method over 10 independent runs (each with random test/train splits) on each dataset. It is readily seen that LSPCA outperforms or is competitive with all competing methods on each dataset. In each case, the variation explained by LSPCA approaches that of PCA.

6. CONCLUSIONS AND FUTURE WORK

This paper has proposed a new approach to SPCA, in both the classification and regression settings. Our experimental results show that the new methods outperform existing approaches in many cases in terms of prediction error, variation explained, or both. The framework we have proposed provides many avenues for future work including other losses and nonlinear predictors. We also note that our method naturally extends to the semi-supervised setting, where unlabeled data may be used in the PCA term of our objective function. Finally, when q is large, it may be of interest to simultaneously learn low-dimensional representations of both X and Y.

6.1. Acknowledgements

The authors would like to acknowledge the UCI Machine Learning repository (18), from which many datasets used in this paper were obtained.

A. Ritchie, C. Scott, and L. Balzano were partially supported by NSF grant 1838179. A. Rithchie was also partially supported by R01MH107741 (Sripada PI), Michigan Alzheimer's Disease Center Pilot Grant (Sripada PI), and DARPA grant 16-43-D3M-FP-037. L. Balzano was also partially supported by ARO YIP award W911NF1910027 and DARPA grant 16-43-D3M-FP-037. Daniel Kessler was supported by NSF grant DMS-1646108. Chandra Sripada was supported by R01MH107741 and a grant from the Dana Foundation David Mahoney Neuroimaging Program.

References

[1] David L Donoho et al., "High-dimensional data analysis: The curses and blessings of dimensionality," *AMS math challenges lecture*, vol. 1, no. 2000, pp. 32, 2000.

[2] Richard E Bellman, *Adaptive control processes: a guided tour*, vol. 2045, Princeton university press, 2015.

[3] John P Cunningham and M Yu Byron, "Dimensionality reduction for large-scale neural recordings," *Nature neuroscience*, vol. 17, no. 11, pp. 1500, 2014.

[4] John P Cunningham and Zoubin Ghahramani, "Linear dimensionality reduction: Survey, insights, and generalizations," *The Journal of Machine Learning Research*, vol. 16, no. 1, pp. 2859–2900, 2015.

[5] Eric Bair, Trevor Hastie, Debashis Paul, and Robert Tibshirani, "Prediction by supervised principal components," *Jour-*

[1] https://archive.ics.uci.edu/ml/datasets.html

nal of the American Statistical Association, vol. 101, no. 473, pp. 119–137, 2006.

[6] Elnaz Barshan, Ali Ghodsi, Zohreh Azimifar, and Mansoor Zolghadri Jahromi, "Supervised principal component analysis: Visualization, classification and regression on subspaces and submanifolds," *Pattern Recognition*, vol. 44, no. 7, pp. 1357–1371, 2011.

[7] Shipeng Yu, Kai Yu, Volker Tresp, Hans-Peter Kriegel, and Mingrui Wu, "Supervised probabilistic principal component analysis," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2006, pp. 464–473.

[8] Juho Piironen and Aki Vehtari, "Iterative supervised principal components," *arXiv preprint arXiv:1710.06229*, 2017.

[9] Aleix M Martínez and Avinash C Kak, "Pca versus lda," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, , no. 2, pp. 228–233, 2001.

[10] Jerome Friedman, Trevor Hastie, and Robert Tibshirani, *The elements of statistical learning*, vol. 1, Springer series in statistics New York, 2001.

[11] Alan Edelman, Tomás A Arias, and Steven T Smith, "The geometry of algorithms with orthogonality constraints," *SIAM journal on Matrix Analysis and Applications*, vol. 20, no. 2, pp. 303–353, 1998.

[12] Nicolas Boumal, Pierre-Antoine Absil, and Coralia Cartis, "Global rates of convergence for nonconvex optimization on manifolds," *IMA Journal of Numerical Analysis*, 2016.

[13] Gene H Golub and Victor Pereyra, "The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate," *SIAM Journal on numerical analysis*, vol. 10, no. 2, pp. 413–432, 1973.

[14] P-A Absil, Robert Mahony, and Rodolphe Sepulchre, *Optimization algorithms on matrix manifolds*, Princeton University Press, 2009.

[15] Fang Zhou, Q Claire, and Ross D King, "Predicting the geographical origin of music," in *2014 IEEE international conference on data mining (ICDM)*. IEEE, 2014, pp. 1115–1120.

[16] Mohammad Hossein Rafiei and Hojjat Adeli, "A novel machine learning model for estimation of sale prices of real estate units," *Journal of Construction Engineering and Management*, vol. 142, no. 2, pp. 04015066, 2015.

[17] Athanasios Tsanas, Max A Little, Patrick E McSharry, and Lorraine O Ramig, "Accurate telemonitoring of parkinson's disease progression by noninvasive speech tests," *IEEE transactions on Biomedical Engineering*, vol. 57, no. 4, pp. 884–893, 2010.

[18] Dua Dheeru and Efi Karra Taniskidou, "UCI machine learning repository," 2017.