Time-varying Linear Regression with Total Variation Regularization

Matt Wytock
Machine Learning Department
Carnegie Mellon University
mwytock@cs.cmu.edu

Abstract

We consider modeling time series data with time-varying linear regression, a model that allows the weight matrix to vary at every time point but penalizes this variation with the (multivariate) total variation norm. This corresponds to simultaneously learning the parameters of multiple linear systems as well as the change points that describe when the underlying process switches between linear models. Computationally, this formulation is appealing as parameter estimation can be done using convex methods; we derive a fast Newton-like algorithm by considering the dual problem and by exploiting sparsity with an active set approach. We also develop an extension for prediction using the learned parameters with a kernel density estimator that exploits recurrent behavior in the time series. Our motivating example is the problem of modeling and predicting energy consumption—specifically learning models of home appliances which tend to be well-described as switched linear systems. On synthetic data we demonstrate that our algorithm is significantly faster than a straightforward implementation using ADMM; however, we also observe that the total variation norm often over-segments suggesting that some applications may require an additional polishing step. On real data we show that our method learns a sparse set of parameters describing the energy consumption of a refrigerator and enables us to predict future consumption significantly better than standard methods.

1 Introduction

We consider modeling time series data in which we are given output variables $y_1, \ldots, y_T \in \mathbb{R}^p$ and input features $a_1, \ldots, a_T \in \mathbb{R}^n$. In our setting, where the data are naturally observed as function of time, we adopt the convention of referring to T both as the sample size and the length of the time series—also, we use a_t to refer to the input features as opposed to the usual machine learning notation of x_t which we reserve for the parameters of the model. If we were to assume that y_t is a time-invariant (linear) function of a_t, the standard approach

would be to estimate a mapping $X \in \mathbb{R}^{p \times n}$ by solving the least squares problem

$$\underset{X}{\text{minimize}} \frac{1}{2} \sum_{t=1}^{T} \|Xa_t - y_t\|_2^2, \tag{1}$$

with a solution representing the best time-invariant X (under squared loss). However, for many types of time series data this is a strong assumption—for example, y_t may depend not just on a_t but also on some unobserved latent state or other factors that change over time—thus we are interested in models that allow the function mapping a_t to y_t to be time-varying.

In particular, we replace the static linear map X with one that is allowed to change at each time point, resulting in a high-dimensional model with parameters growing with the sample size. In this setting, the model is poorly specified without additional regularization; therefore, we augment the objective by adding a *total variation* penalty on the differences of the parameters

$$\underset{X_1,\ldots,X_T}{\text{minimize}} \frac{1}{2} \sum_{t=1}^{T} \|X_t a_t - y_t\|_2^2 + \sum_{t=1}^{T-1} \lambda_t \|X_{t+1} - X_t\|_F \tag{2}$$

where $\lambda_t \geq 0$ are regularization parameters and $\|\cdot\|_F$ denotes the Frobenius norm. Intuitively, total variation regularization on the differences $X_{t+1} - X_t$ encourages these differences to be *sparse*—due to the nonsmoothness of the Frobenius norm, each difference will tend to be either all zero or all nonzero. Computationally, sparsity is appealing as it allows us to derive efficient estimation procedures that scale to large problems by exploiting sparsity in the differences of X_t. This is especially the case when the number of change points is much less than the sample size.

Our approach to solving the time-varying linear regression optimization problem is a fast Newton method which scales to large (sparse) problems while finding highly accurate solutions. This approach is similar to the one derived in our recent work [8] which considers the simpler problem of finding a direct reconstruction of y_t by minimizing squared reconstruction loss and the multivariate total variation norm. In particular, that method could be used as a subroutine for computing the proximal operator of the total variation norm in an algorithm for solving the linear regression problem. However, we find it advantageous to derive a fast method specifically for this problem and our approach for doing so follows a similar outline to that of [8]—we begin by deriving the dual; next, we modify the problem to make the constraint smooth and consider the dual of that problem (the dual dual); finally, we design an efficient projected Newton method for the dual dual which exploits sparsity in the change points with an active set approach.

What is particularly interesting about the time-varying linear regression formulation is that the inclusion of the a_t covariates enables a new class of applications. In particular, while applying total variation regularization in reconstructing the output signal itself leads to a piecewise constant approximation, the total variation norm applied to the linear regression weights results in piecewise constant estimates of those coefficients. In the language of dynamical systems, this corresponds to a time-varying process switching between linear systems with the optimization problem adaptively picking the appropriate change points.

The time-varying linear regression problem was first proposed in this context in [6], which focuses on segmentation and system identification for time-varying linear systems. In this work, we study this model and consider fast algorithms for problems involving time-varying processes that undergo relatively few changes relative to the number of observations.

Our work on this method is inspired by applications in the energy domain which frequently involve the modeling and forecasting of time series data. In particular, we consider modeling the energy consumption of a single device with high resolution data (1 data point per minute). First, we consider the the modeling task and find that devices can typically be described as regularly switching between a small number of distinct linear systems. For example, we find that the energy consumption of a refrigerator switches between several states—on, off and an initial state characterized by a power spike as the compressor switches from off to on. Second, observing that devices are often characterized by recurrent behavior, we develop a model that predicts future energy consumption as a function of the current parameters and the amount of time since the last change point. Concretely, we employ kernel density estimation to model $y_{t+f}|X_t, z_t$ where z_t denotes the time spent in the current state. We evaluate this prediction method over a range of future time windows from 1 minute to 3 hours and show that our method outperforms a number of reasonable baselines.

The rest of the paper is organized as follows. We begin with a review of related work on time series analysis in Section 2 followed by an illustration of the properties of time-varying linear regression using two simple examples, a random Fourier basis and a time-varying autoregressive model in Section 3. In Section 4, we derive our fast Newton method by deriving the "dual dual" problem and an active set method exploiting sparsity in the change points. In Section 6 we turn to empirical evaluations, comparing our fast algorithm to a standard ADMM approach as well as considering model selection on synthetic data. In Section 7, we consider the modeling and prediction of device energy consumption at 1 minute granularity and conclude in Section 8.

2 Related work

To the best of our knowledge, [6] was the first to propose the time-varying linear regression optimization problem that we consider here, referring to the regularization penalty as the *sum of norms*. In that work, they consider applications of system identification and show that on various tasks the proposed method recovers both the correct parameters of the underlying systems as well as identifying the correct change points when (for example) a time-varying system switches between linear systems. In addition to solving the time-varying linear regression problem, their proposed method also includes the use of an iterative reweighting followed by a polishing step—we also discuss the use of polishing in Section 6, finding that the total variation penalty tends to over-segment with polishing leading to accurate, sparse solutions. In terms of the optimization algorithm, they employ standard software (CVX) and consider somewhat smaller examples than in our work.

Algorithmically, recent work has considered the multivariate total variation norm under the name of the *group fused lasso* including [3, 1] and our own work in [8] with applications

in computational biology and computer vision. These efforts do not consider the additional linear regression term a_t, but the method developed in [8] inspires the algorithmic approach we develop here. The development of an optimization method specific to the linear regression problem achieves significantly better convergence than is possible with existing methods.

Historically, perhaps the most common approach to modeling time series data with latent states has been through the use of hidden Markov models (HMMs). In fact, for the specific application of modeling sources of energy consumption, a factorial HMM was proposed in [5]. While such models are very expressive, estimation is difficult due to the nonconvexity introduced by the explicit representation of the unobserved state. Instead, the time-varying linear regression method discussed here implicitly allows for state changes (by allowing X_t to shift over time) and benefits from estimation with efficient convex methods. Comparison between time-varying linear regression and HMM-based models is an interesting topic for future work.

3 Time-varying linear models

In this section we give two illustrative examples of processes which are well-suited to the time-varying linear model. In essence, these are processes which "switch" between linear systems at relatively few change points—behavior that is captured by a piecewise constant X_t. In the first example, the covariates are smoothly varying functions of time (a Fourier basis) showing that although the *weights* are piecewise constant, the output signal, which is a linear combination of an arbitrary set of input features, can be smooth. In the second example, autoregressive features are used to construct a process switching between arbitrary linear systems. While these examples are simple, we can consider them as two examples of atoms which combine to form a rich class of time-varying processes amenable to convex estimation with the proposed method.

3.1 Random Fourier basis

Our first example considers input features containing a random Fourier basis of the form

$$a_t = \begin{bmatrix} \cos(\omega_1 t + \phi_1) \\ \vdots \\ \cos(\omega_n t + \phi_n) \end{bmatrix} \tag{3}$$

where $\omega_i, \ldots, \omega_n$ are angular frequencies and $\phi_i, \ldots, \phi_n$ are phase angles. Note that in general the covariates a_t can be arbitrary and (as in any regression problem) are typically chosen based on the application—we use the Fourier basis here as an example of covariates which are smooth functions of time in order to illustrate the difference between piecewise constant structure in the weights X_t and the smoothness of the reconstructed output prediction $X_t a_t$.

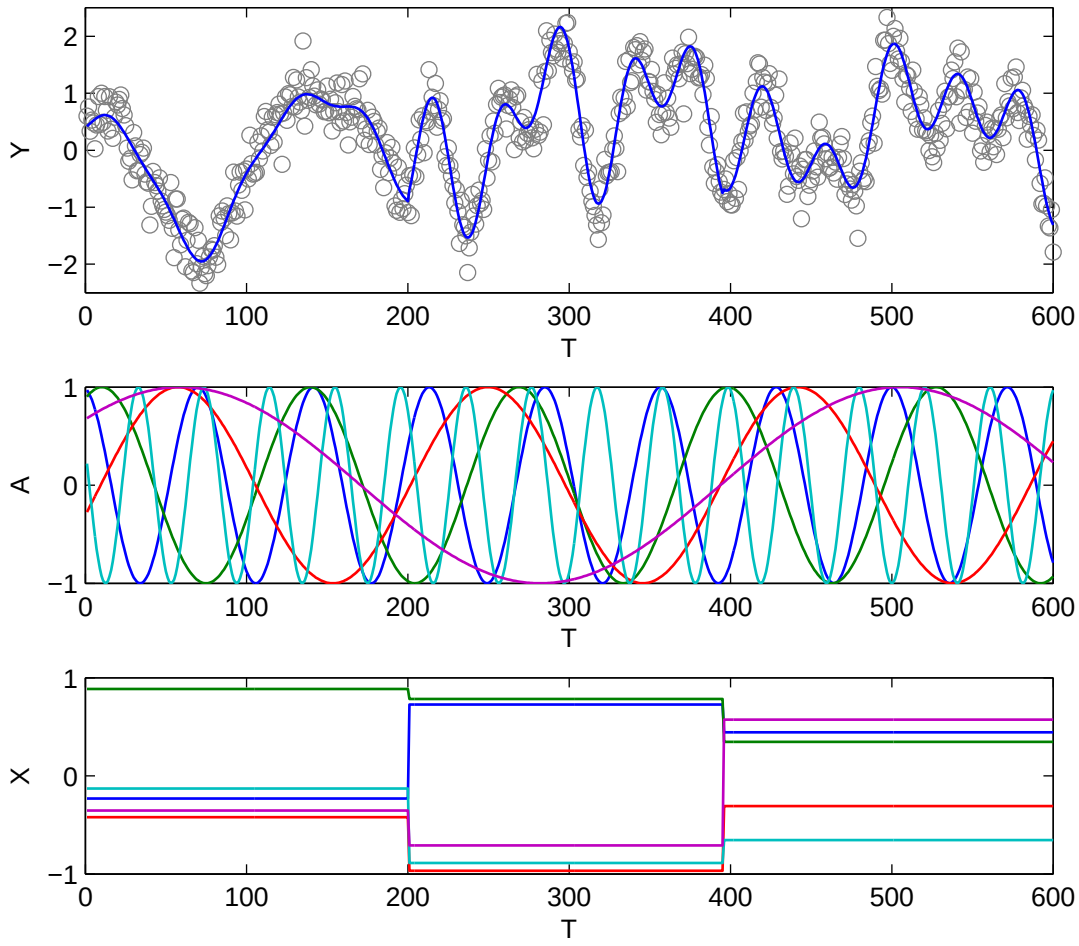

Figure 1: An example of a time-varying process generated from a random Fourier basis of the form $\cos(\omega t + \phi)$ (middle) and a piecewise constant set of weights (bottom). These combine to form a smooth signal containing arbitrary combinations of frequencies, which we observe with a small amount of Gaussian noise ($\sigma = 0.3$) (top).

In this example, we have a set of weights composed of a small number of piecewise constant segments leading to a smooth output due to the Fourier basis.

We see this in Figure 1, where the output signal jumps between three different smoothly varying signals. Typically, we would observe the signal with the addition of noise (as shown in the figure) but given the appropriate input features we expect our method to recover the correct weighting of this basis resulting in the observed signal. For many applications there may be a natural basis—for example, in modeling signals in the electrical grid we would expect a strong 60Hz component.

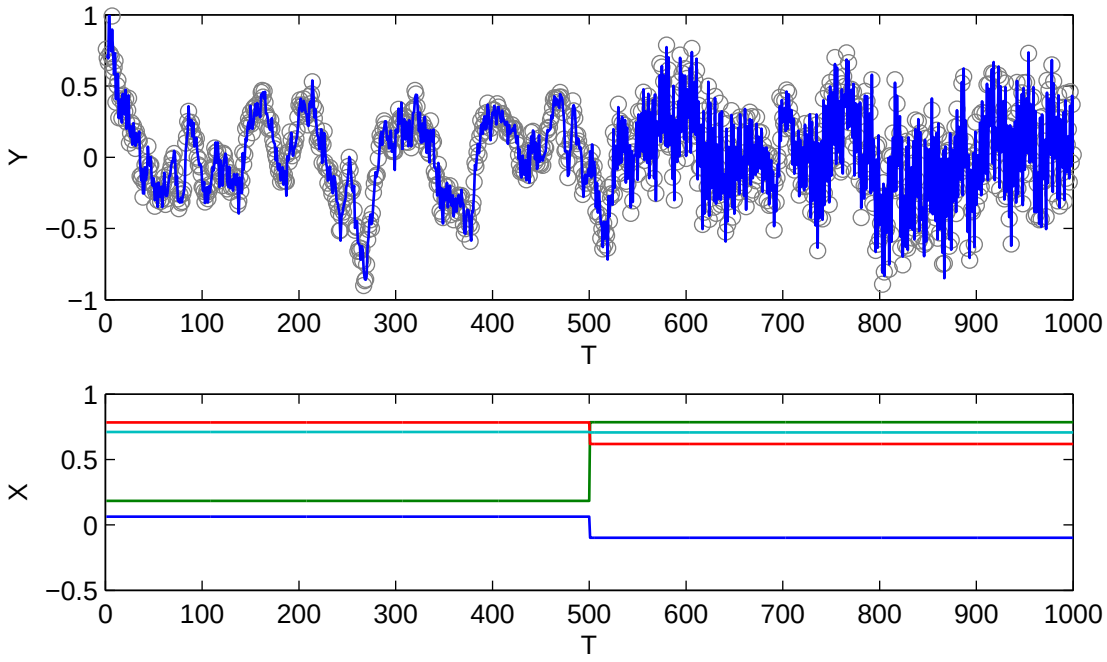

Figure 2: An example of switching between two AR(4) processes with weights (bottom). Each point is formed from a linear combination of the previous 4 points and a small amount of Gaussian noise resulting in the output signal (top).

3.2 Time-varying autoregressive model

In our second example, we consider an autoregressive model in which the covariates are formed from the previous time points

$$a_t = \begin{bmatrix} y_{t-1} \\ \vdots \\ y_{t-n} \end{bmatrix}, \tag{4}$$

a widely used model in time series analysis as it represents the dynamics of a (fully observed) linear system. In this example, a piecewise constant set of parameters X_t correspond directly to switching between linear systems and thus estimating parameters in our model can be thought of as a system identification task in which we jointly estimate the coefficients of the linear systems as well as identify change points between systems.

We see this in Figure 2 where at time $t = 500$ the we switch to a different linear system which has significantly different dynamics than the previous system. For some applications the order of our autoregressive process may be known but in others we can reasonably estimate this from data.

4 Active set projected Newton method

In what follows, we consider the computational aspects of the proposed method and derive an efficient optimization algorithm which exploits sparsity in the differences of $X_{t+1} - X_t$. Although the original problem contains the nonsmooth total variation norm, by deriving the dual, modifying that problem and deriving its dual (in a way that leads to a new problem) the result is an optimization problem with a smooth objective and simple nonnegative constraints. Despite the fact that this problem involves the inverse of a matrix with size $Tn \times Tn$, sparse structure can be used to compute the objective, gradient and Hessian in $O(T)$ time. Finally, we reduce the running time further by employing an active set method which fixes many of the variables and considers a significantly reduced problem with dimension typically much less than T.

We begin by writing the primal problem in more compact vector notation by defining

$$X = \left[\begin{array}{ccc} X_1 & \cdots & X_T \end{array} \right], \qquad Y = \left[\begin{array}{ccc} y_1 & \cdots & y_T \end{array} \right] \tag{5}$$

and forming the block diagonal matrix

$$A = \begin{bmatrix} | & & \\ a_1 & & \\ | & & \\ & | & \\ & a_2 & \\ & | & \\ & & \ddots \end{bmatrix} \tag{6}$$

resulting in $X \in \mathbb{R}^{p \times Tn}$, $Y \in \mathbb{R}^{p \times T}$, and $A \in \mathbb{R}^{Tn \times T}$. To handle the total variation term, we define the difference operator as the $Tn \times (T-1)n$ banded matrix

$$D_{ij} = \begin{cases} -1 & \text{if } i = j \\ 1 & \text{if } i + n = j \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

such that multiplying XD results in a matrix containing the differences $X_{t+1} - X_t$

$$XD = \left[\begin{array}{ccc} X_2 - X_1 & \cdots & X_T - X_{T-1} \end{array} \right] \tag{8}$$

and we define the matrix norm $\|\cdot\|_{1,F}$ such that

$$\|XD\|_{1,F} = \sum_{t=1}^{T-1} \|X_{t+1} - X_t\|_F \tag{9}$$

corresponds to the ℓ_1 norm of the Frobenius norm applied to the blocks of XD. With this notation, and the addition of a small amount of ℓ_2 regularization on X which will allow us to form the dual more easily, we write the primal problem (2) as

$$\underset{X}{\text{minimize}} \, \frac{1}{2}\|XA - Y\|_F^2 + + \frac{\mu}{2}\|X\|_F^2 + \|XD\Lambda\|_{1,F} \tag{10}$$

where Λ is the a diagonal matrix with $\Lambda_{tt} = \lambda_t$. This problem represents a particular form of the group fused lasso—observe that the regularization term $\|XD\Lambda\|_{1,F}$ is the sum of nonsmooth Frobenius norm, which is simply the ℓ_2 norm applied to the elements of the matrix. Optimization in the primal form is complicated by the nonsmooth regularization term; also, note that even in the case where the differences $X_{t+1} - X_t$ are sparse, X will (in general) be dense.

4.1 The dual and the dual dual

To address these issues, we proceed by forming the dual problem and its dual (the dual dual) in such a way that leads to a new problem with smooth objective and constraints. To form the first dual, we introduce a new variable Δ and the constraint $\Delta = XD$, resulting in the equivalent problem

$$\begin{array}{ll} \underset{X,\Delta}{\text{minimize}} & \frac{1}{2}\|XA - Y\|_F^2 + + \frac{\mu}{2}\|X\|_F^2 + \|\Delta\Lambda\|_{1,F} \\ \text{subject to} & XD = \Delta \end{array} \tag{11}$$

from which we form the Lagrangian

$$L(X, \Delta, U) = \frac{1}{2}\|XA - Y\|_F^2 + + \frac{\mu}{2}\|X\|_F^2 + \|\Delta\Lambda\|_{1,F} + \operatorname{tr} U^T(XD - \Delta) \tag{12}$$

which is minimized for X by

$$X = (YA^T - UD^T)(AA^T + \mu I)^{-1} \tag{13}$$

and for Δ with

$$\min_{\Delta} \|\Delta\Lambda\|_{1,F} - \operatorname{tr} U^T\Delta = \begin{cases} 0 & \text{if } \|U_t\|_F \le \lambda_t \quad \text{for } t = 1, \ldots, T-1 \\ -\infty & \text{otherwise} \end{cases} \tag{14}$$

leading to the dual problem

$$\begin{array}{ll} \underset{U}{\text{maximize}} & -\frac{1}{2}\|YA^T - UD^T\|_B^2 + \frac{1}{2}\|Y\|_F^2 \\ \text{subject to} & \|U_t\|_F \le \lambda_t \text{ for } t = 1, \ldots, T-1 \end{array} \tag{15}$$

where $B = (AA^T + \mu I)^{-1}$; to simplify notation, we have defined the quadratic matrix norm for any positive definite matrix P as

$$\|X\|_P = (\operatorname{tr} X^T P X)^{1/2}. \tag{16}$$

Observe that the first dual is a second-order cone program with a matrix variable $U \in \mathbb{R}^{p\times(T-1)n}$ and while the objective is smooth but the constraint is still nonsmooth. However,

we can modify the constraint to by replacing $\|\cdot\|_F$ with $\|\cdot\|_F^2$ which is differentiable. With this modification and dropping constant term, we have the equivalent problem

$$\begin{aligned} \underset{U}{\text{maximize}} \quad & -\frac{1}{2}\|YA^T - UD^T\|_B^2 \\ \text{subject to} \quad & \frac{1}{2}\|U_t\|_F^2 \leq \frac{\lambda_t^2}{2} \quad \text{for } t = 1, \ldots, T-1 \end{aligned} \tag{17}$$

from which we form the Lagrangian

$$L(U, z) = -\frac{1}{2}\|YA^T - UD^T\|_B^2 + \frac{1}{2}\sum_{t=1}^{T-1} z_t(\lambda_t^2 - \|U_t\|_F^2). \tag{18}$$

We write this in vector notation by introducing a $(T-1)n \times (T-1)n$ diagonal matrix Z with each z_t repeated n consecutive times on the diagonal (i.e. $Z = \text{diag}(z) \otimes I_n$),

$$L(U, z) = -\frac{1}{2}\|YA^T - UD^T\|_B^2 - \frac{1}{2}\|UZ^{1/2}\|_F^2 + \frac{\lambda^2}{2}1^T z \tag{19}$$

where λ^2 is interpreted as elementwise squaring of λ. This is maximized by

$$U = YA^T BD(D^T BD + Z)^{-1} \tag{20}$$

leading to the second dual problem

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & \frac{1}{2}YA^T BD(D^T BD + Z)^{-1}D^T BAY^T + \frac{\lambda^2}{2}1^T z \\ \text{subject to} \quad & z \geq 0. \end{aligned} \tag{21}$$

The main advantage of the dual dual optimization problem is the smooth objective and simple nonnegative constraint. In addition, the primal and the first dual had $O(Tnp)$ variables while the dual dual has only $T-1$ variables, the nonzero components of which correspond directly to the change points $X_{t+1} - X_t$. In problems of interest, we expect z to be sparse at the solution, corresponding to models with sparse changes, a fact which we will exploit in our active set method.

However, the objective of the dual dual is relatively complex as it involves the inverse of a $Tn \times Tn$ matrix which is itself composed of another inverse, $B = (AA^T + \mu I)^{-1}$. Fortunately, these matrices have sparse structure. First, we observe that AA^T is block diagonal, the direct sum of $n \times n$ rank one blocks, and by taking its eigendecomposition

$$AA^T = \tilde{A}S\tilde{A}^T \tag{22}$$

we see that B is also block diagonal with the same structure

$$B = (AA^T + \mu I)^{-1} = \tilde{A}(S + \mu I)^{-1}\tilde{A}^T. \tag{23}$$

Furthermore, due to the banded structure of D, $D^T BD + Z$ is a block diagonal but unfortunately, since Z is not the identity, its inverse is *not* block diagonal and will in general be dense. Nonetheless, the Cholesky decomposition

$$R^T R = D^T BD + Z \tag{24}$$

is sparse and requires $O(Tn^3)$ operations. This decomposition forms the dominant computation cost in our approach but it is sufficient to compute the objective function and gradient as well as a subset of the Hessian.

The gradient and Hessian of the objective in (21) is given by

$$\begin{aligned} \nabla_z &= -\frac{1}{2}(U \circ U)1 + \frac{\lambda^2}{2}1 \\ \nabla_z^2 &= V \circ UU^T \end{aligned} \tag{25}$$

where 1 denotes the ones vector, $\circ$ denotes the elementwise (Hadamard) product and $V = (D^T BD + Z)^{-1}$. Although both the gradient and Hessian involve this inverse, this can be computed efficiently with the sparse Cholesky decomposition described above and backsubstitution. Similarly, we can compute a subset of the Hessian involving k rows with k backsubstitutions and $O(Tk)$ time.

Given that the gradient and Hessian (for a reduced set of coordinates) can be computed efficiently, our approach to optimization on the dual dual is a projected Newton method, a general second-order method for smooth problems with simple constraints [2]; also see [8] for a detailed description of an efficient implementation in the special case where A is the identity matrix. The main difficultly in the application of projected Newton method is due to controlling the size of the Hessian—we accomplish this with the active set method described in the next section.

4.2 Active set and reduced problem

In the previous section, we derived a fast Newton method which is linear in the number of time points T. However, for large problems (e.g. $T > 1000$) this is still relatively expensive, especially in the computation of the Newton step and subsequent line searches which require computing the objective many times. To address this, we employ an active set method which fixes many of the differences $X_{t+1} - X_t$ allowing us to solve a significantly reduced problem.

The basic intuition behind the active set method is that if we knew the correct set of change points at the solution, $c_1, \ldots, c_k$, we could equivalently solve a reduced version of (10),

$$\underset{X}{\text{minimize}}\, \frac{1}{2}\|XA' - Y\|_F^2 + + \frac{\mu}{2}\|X\|_F^2 + \|XD\Lambda\|_{1,F} \tag{26}$$

where A' is a reduced version of the original A matrix with significantly fewer rows,

$$A' = \begin{bmatrix} | & & | & & & & \\ a_1 & \cdots & a_{c_1-1} & & & & \\ | & & | & & & & \\ & & & | & & | & \\ & & & a_{c_1} & \cdots & a_{c_2-1} & \\ & & & | & & | & \\ & & & & & & \ddots \end{bmatrix} \tag{27}$$

and dimension $kn \times T$. In this reduced problem, the optimization variable X has size $p \times kn$ and is thus $O(k)$ rather than $O(T)$.

Since we do not know the change points at the solution, we use an iterative procedure that forms the active set based on the current iterate, optimizes the dual dual for that active set and then forms a new active set and repeats. This method corresponds to blockwise coordinate descent and can be shown to converge. In practice, we limit the maximum size of the active set allowing the reduced subproblems to be solved very quickly in the case where the size of the active set k is much smaller than the length of the time series T. To form the active set at the current iterate z, we use the condition

$$z_i \neq 0 \text{ or } (\nabla_z)_i < -\epsilon \tag{28}$$

for small $\epsilon > 0$, corresponding to selecting coordinates that are currently nonzero or those which following the gradient at the current iterate would make nonzero.

5 Kernel density estimation and prediction

Thus far we have focused on the task of recovering the time-varying linear map X_t from a sequence of input/output pairs $(a_1, y_1), \ldots, (a_T, y_T)$. While this is useful for analyzing past behavior, it does not explicitly result in a model that can be used to predict future time points. Furthermore, as we explore empirically in Section 6 simple heuristics such as predicting $\hat{y}_{t+1} = X_t a_{t+1}$ typically perform worse than baseline models as simple as $\hat{y}_{t+1} = y_t$. Fundamentally, while the time-varying linear regression model captures piecewise constant structure in X_t from past data, unlike (for example) an HMM, it does not model how this distribution evolves over time.

We address this issue through the use of kernel density estimation over the learned parameters $X_1, \ldots, X_T$ resulting in a two-stage procedure in which we first solve the time-varying linear regression problem followed by density estimation. To mimic HMM-like behavior, we could learn a distribution for $X_{t+1}|X_t$ and then use this to predict $\hat{X}_{t+1}$ and subsequently $\hat{y}_{t+1} = \hat{X}_{t+1} a_{t+1}$. However, in this work we instead prefer estimating $y_{t+1}|X_t$ directly, resulting in a method for predicting a future output given the current state. In addition, we augment the parameter space with an additional variable z_t representing the number of time points since the last change point. For the modeling device energy consumption, including z_t allows us

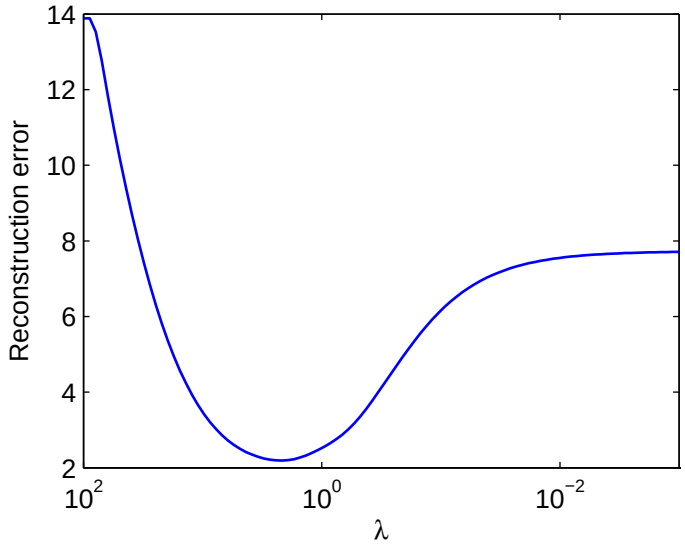

Figure 3: Reconstruction accuracy for varying λ on data generated from a time-varying linear combination of random Fourier functions described in Section 3.

to explicitly account for strongly recurrent behavior in the sequence of states in the system, e.g. the refrigerator compressor is on for roughly 10 minutes before switching off. In general, this allows our model to more easily represent non-geometric probability distributions time spent in a particular state.

Concretely, we model $y_{T+1}|X_T, z_T$ with the kernel estimator

$$\hat{y}_{T+1} = f_h(\theta) = \frac{1}{T}\sum_{t=1}^{T} \frac{y_t}{h^d} K\left(\frac{\|\theta - \theta_t\|_2}{h}\right) \tag{29}$$

where $\theta = (X_t, z_t)$, K is a smooth, symmetric kernel and $h > 0$ is the kernel bandwidth. We note that like the total variation norm, the kernel estimator is based on Euclidean distance in the parameter space, suggesting a possible tighter connection between the bandwidth h and the regularization parameter λ. However, in this work we choose h using cross validation and leave further exploration of the connection between the parameters controlling these two stages for future study.

6 Numerical results on synthetic data

6.1 Model selection and polishing

First, we consider the ability of our method to recover the true change points and parameters of the linear system generated from a random Fourier basis as in Figure 1. Given that the data is generated, we have access to the true noiseless signal which we can use to evaluate

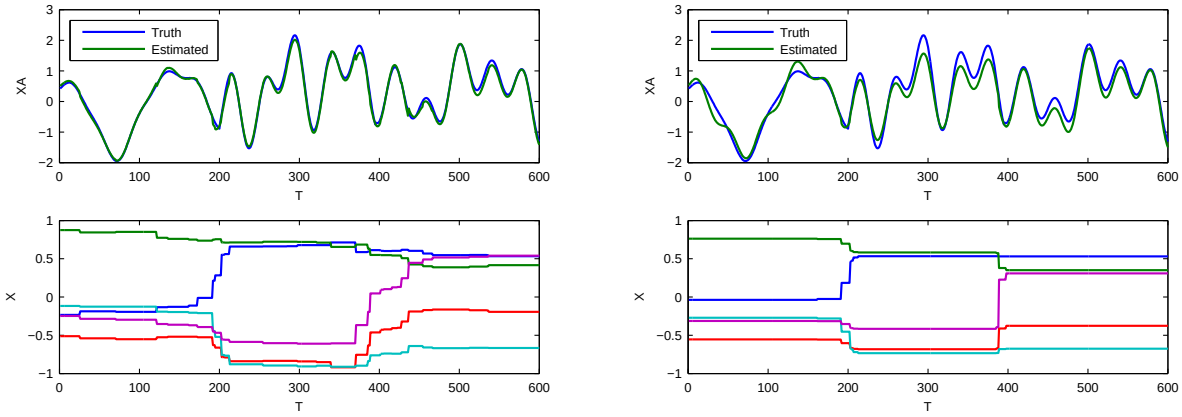

Figure 4: Shrinkage vs. change point detection with the total variation norm on random Fourier example. On left we show the model found with $\lambda = \lambda^\star$ leading to the best possible reconstruction of the noiseless signal (see Figure 3) but observe that this results in over-segmentation. On right we show $\lambda = 10\lambda^\star$ which recovers the change points more accurately but with some shrinkage of the signal.

the solution of our model; the results of this procedure are shown in Figure 3 with a very large choice of λ giving a static model and tiny choices of λ fitting noise. We refer to λ which minimizes this reconstruction error as $\lambda^\star$ and using this value we expect to achieve a highly accurate reconstruction; as can be seen in Figure 4 (top left) this is indeed the case. However, from Figure 4 (bottom left) we also see that this accurate reconstruction comes at the expense of requiring several change points as opposed to the original model which has exactly 2. In retrospect the reason for this is clear: the total variation norm tends to prefer several small changes over one large change which can lead to over-segmentation. One possible solution is to use a larger version of λ that results in a less accurate signal reconstruction but fewer change points; for example, in Figure 4 (right) we see that by choosing $\lambda = 10\lambda^\star$ we have more fewer change points, accurate identification of change points but significant shrinkage in the underlying weights.

To address this shrinkage vs. change point detection tradeoff, previous authors [6] have suggested iterative reweighting [4] and/or an additional polishing step in which the sparsity pattern is held fixed and the model is refit on the nonzero components without using regularization. On this example, we show the effect of polishing in Figure 5, finding that this two-step procedure does a good job of both identifying the correct change points as well as fitting parameters for the segments for an accurate reconstruction of the original signal. The use of polishing is somewhat application-specific as for some applications (e.g. the prediction task we consider later) this shrinkage may not create a problem. Overall, a better understanding of the benefits and costs of shrinkage with the total variation norm and time-varying linear regression is an open problem for future study.

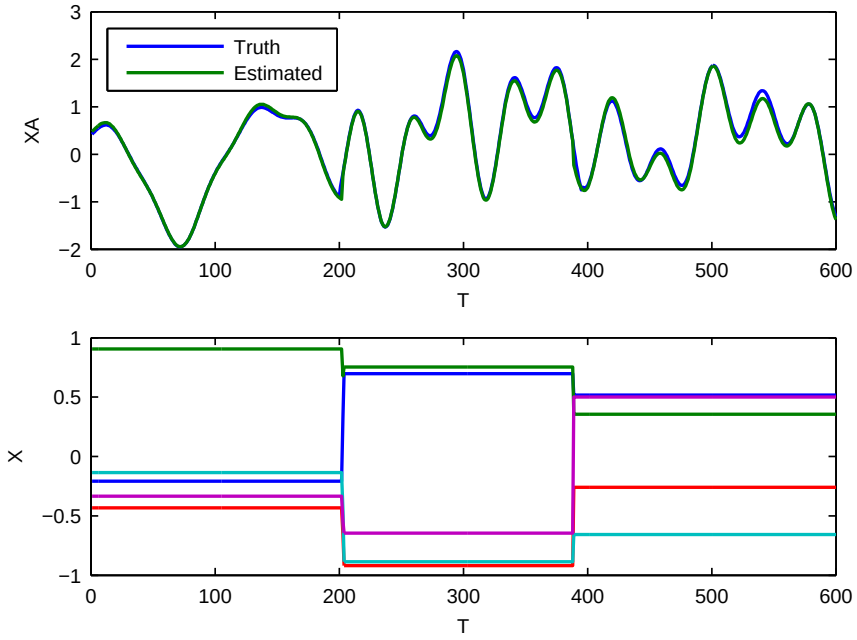

Figure 5: Accurate recovery of change points and noiseless signal on random Fourier basis example using an additional polishing step.

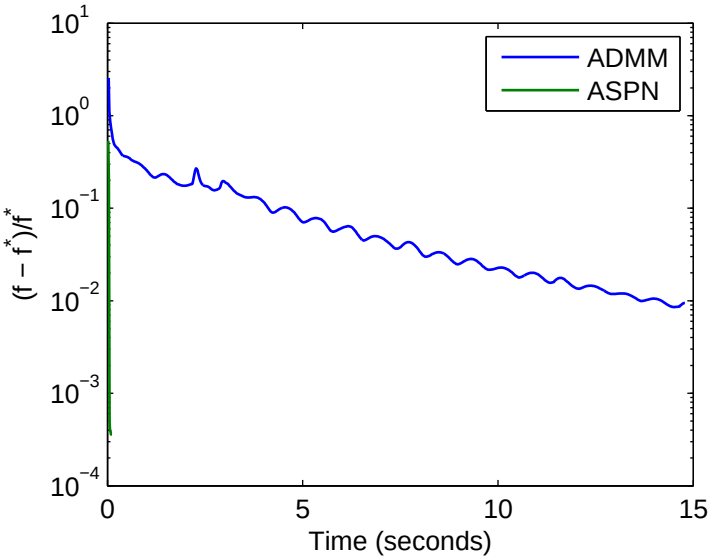

Figure 6: Solution suboptimality vs. time for ASPN and ADMM algorithms on the random Fourier basis synthetic example with λ chosen to minimize reconstruction error. ASPN converges to 10^{-3} relative accuracy in 0.11 seconds whereas ADMM has not reached this level after 15 seconds.

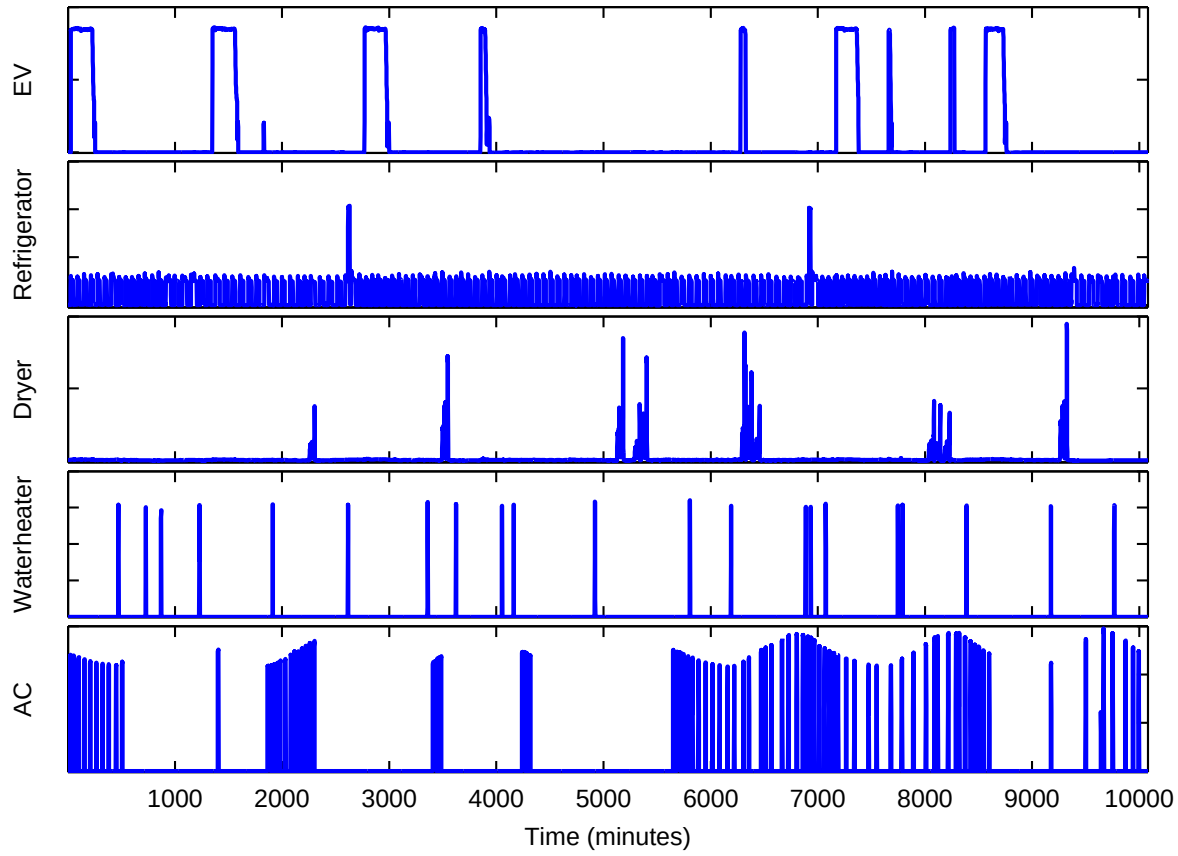

Figure 7: Typical device energy consumption for for from the Pecan Street dataset measured in kilowatts once per minute over one week. Some devices display strongly recurrent behavior, for example the waterheater and refrigerator which we can model with a time-varying autoregressive process.

6.2 Timing results

Next, we turn to an evaluation of our proposed active set projected Newton (ASPN) algorithm with a comparison to a standard ADMM approach to the algorithm. Applying ADMM to this problem is straightforward once this problem is in the form of (11). In particular, we decompose the objective into two simple subproblems which can be handled efficiently as least squares and soft-thresholding; however, first-order methods such as ADMM tend to require a significant number of iterations to converge to accurate solutions and we observe this empirically in Figure 6. In this example ADMM indeed requires more than 100 outer iterations while ASPN converges in less than 10 resulting in a significantly faster running time for ASPN.

Algorithm	RMSE
Mean	0.0536
Last	0.0191
AR(1)	0.0182
TVLR AR(1)	**0.0172**

Table 1: Performance on modeling the energy consumption of a refrigerator under various models measured with held-out data.

7 Modeling and predicting energy consumption

In this section, we evaluate our model on the task of modeling and predicting sources of energy consumption at the individual device level with high frequency measurements (1 per minute). Understanding load profiles is useful for many applications and is (for example) a natural subproblem of energy disaggregation, the task of separating an aggregate whole-home energy signal into its source components. Past work on energy disaggregation has proposed the use of hidden Markov models for this task [5] as the overall consumption can be modeled as an additive combination of several devices. Here we employ time-varying linear regression to build models for devices enabling us to characterize their energy profile.

In addition to modeling, we also use our learned representation in combination with a kernel density estimator to develop a prediction method as discussed in Section 5. We show that this method significantly outperforms reasonable baselines especially in predicting energy usage over the next 10-60 minutes. Predicting device energy consumption for devices these time scales (next few minutes to hour) is a critical component in matching real-time supply and demand, an important effort in the development of the smart grid with the potential to significantly improve efficiency by reducing the dependence on expensive fast-ramping power plants needed to handle peak load.

Our data comes from the Pecan Street project (`http://www.pecanstreet.org/`), collected with current sensors installed at the circuit level inside the home; typical device profiles with power measurements once per minute over one week are shown in Figure 7. In this work we focus on modeling devices with recurrent behavior (e.g. refrigerator, waterheater) as purely autoregressive processes; nonetheless, our time-varying linear regression method can easily incorporate additional features such as time-of-day, outside temperature, etc. to improve modeling and prediction for all devices under consideration.

7.1 Modeling

First, we consider the modeling task of learning a succinct representation for device energy consumption from historical data. Concretely, given a sequence of $y_1, \ldots, y_T$ representing the power draw of a device in kilowatts (kW), we consider an AR(1) model

$$a_t = \begin{bmatrix} y_{t-1} \\ 1 \end{bmatrix} \tag{30}$$

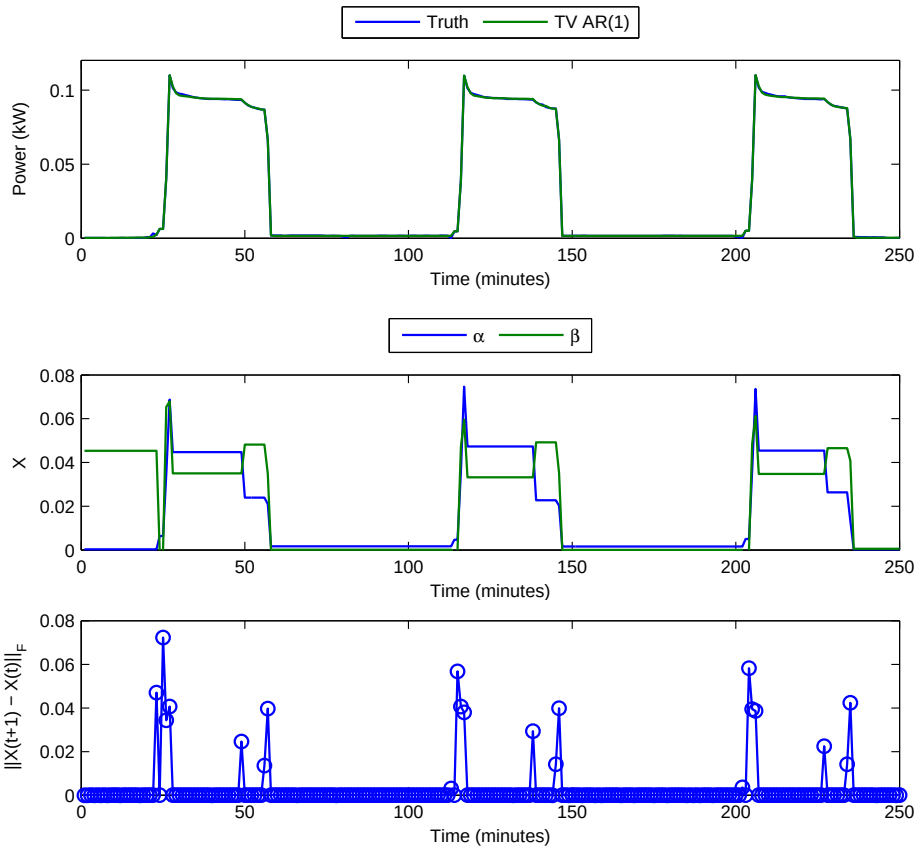

Figure 8: The energy consumption of a typical refrigerator and a time-varying linear model described by an AR(1) process with $y_t = \alpha y_{t-1} + \beta$. On top we see that this model fits the data well and in the middle/bottom we see that the change points found by time-varying linear regression correspond to switching between linear systems with different dynamics.

which captures persistency and exponential decay. Learning this model using the TVLR method allows us to capture switches in dynamics of the system as devices transition states, for example from off to on.

Figure 8 (top) shows the typical load profile of a refrigerator which cycles between the idle state consuming very little energy, and the state when the compressor switches on causing a quick increase in the power draw. For ease of discussion we have explicitly written out the parameters of our time-varying linear model as $y_t = \alpha_t y_{t-1} + \beta_t$; this model captures the behavior of the refrigerator with a solution characterized by (roughly) 4 sets of unique coefficients that capture the dynamics of this process with high accuracy. One set of parameters for representing the refrigerator in the off state (characterized by $\beta_t = 0$) and 3 sets characterizing different regimes of the on state. We see from Figure 9 that the parameters found by this model tend to live in relatively well-defined regions of the parameter space. Evaluating the model on held out data points in Table 1 we see that it outperforms static models autoregressive models that are unable to transition between states.

Finally, we note that the model presented in Figure 8 makes use of a second polishing step

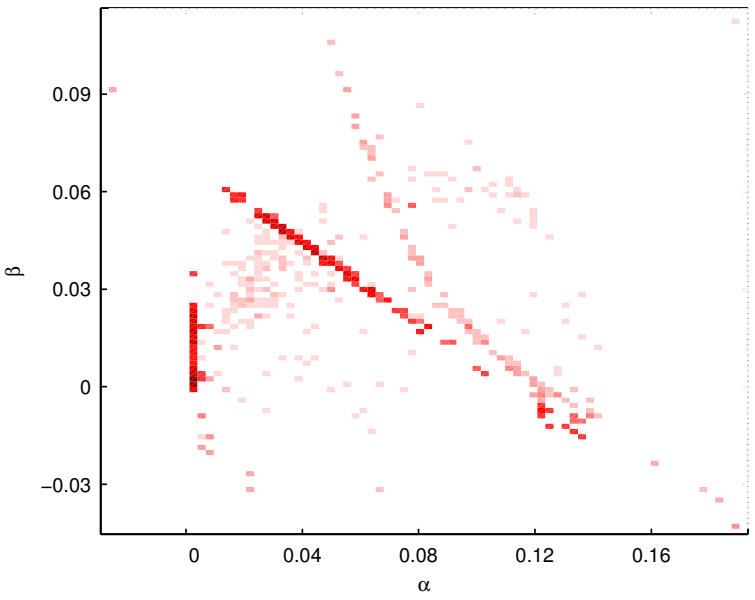

Figure 9: Parameter heat map for time-varying AR(1) model of refrigerator from Pecan Street trained on one week worth of data at 1 minute granularity.

Algorithm	**RMSE**
Mean	0.0520
Last	0.0167
AR(1)	0.0165
Kernel on $y_{t+1}\|y_t$	0.0172
TVLR with X_t	0.0166
TVLR with kernel on $y_{t+1}\|X_t, z_t$	**0.0138**

Table 2: Refrigerator prediction results

which was also discussed in Section 6. We can also consider models fit without polishing. However, on this example this results in models with significantly more change points for the same level of accuracy—2178 vs. 645.

7.2 Prediction

Next, we consider the task of predicting future energy consumption of an individual device; given past observations $y_1, \ldots, y_T$ we would like to predict the amount of energy used at the next time point $\hat{y}_{T+1}$ or more generally at some future time point $\hat{y}_{T+f}$ for $f > 0$. The devices under consideration exhibit a strong amount of persistence such that the naive approach of predicting $\hat{y}_{T+1} = y_T$ does reasonably well and in fact better than the more complicated heuristic of predicting $\hat{y}_{T+1} = X_T a_{T+1}$ which is shown in Figure 10 (left). However, this approach provides little insight into the behavior and fundamentally any method for $\hat{y}_{T+1} = f(y_T)$ cannot hope to do much better as we can see from Figure 10 (right). Intuitively,

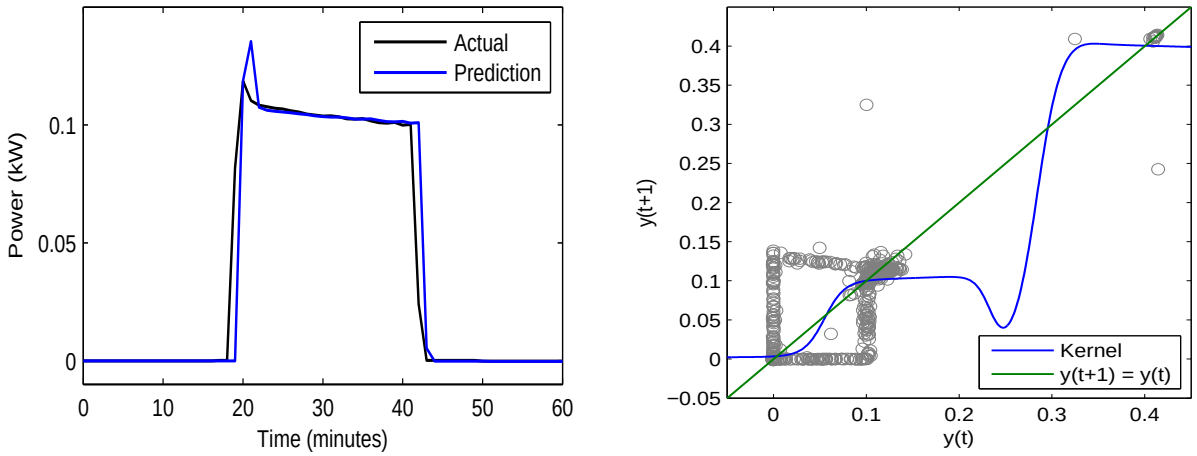

Figure 10: Problems with naive prediction methods, on we show the results of predicting $\hat{y}_{T+1} = X_T a_{t+1}$ which does worse than predicting $\hat{y}_{T+1} = y_T$ and on right we see that it is difficult to do better for any choice of $\hat{y}_{T+1} = f(y_T)$.

although the last time point is on average a good prediction for the next one, it does not provide information as to when the device may change states.

Instead, our approach is to augment the prediction model with additional data provided by the learned parameters from TVLR. In particular we build a model that predicts $\hat{y}_{T+1} = f(X_T, z_t)$ using nonparametric regression, the kernel density estimator described in Section 5). The intuition behind this approach is that for recurrent behavior such as that exhibited by the refrigerator, we model future energy consumption dependent on (the parameters of) the current state, and the amount of time spent in this state. In Table 2 we compare this method to several reasonable baselines including a static AR(1) method and see that in does significantly better at predicting $\hat{y}_{T+1}$. Also, on longer time horizons shown in Figure 11 using features from the TVLR method as a basis or prediction significantly outperforms static models, especially for predictions over the future 1-60 minutes. On this example, both methods converge to the accuracy of predicting the mean value for time horizons greater than 3 hours reflecting the inherently stochastic nature of the device's energy consumption.

8 Conclusions and discussion

The application that we have considered in this work, modeling and predicting device energy consumption, is a critical subtask for many open problems in energy. For example, better models of device energy consumption can inform energy disaggregation efforts, such as those considered in [5] as well as form the building block for real-time matching of supply and demand, an increasingly important problem due to the desire of incorporating variable renewable energy sources (e.g. wind, solar) into the grid. Future solutions to these problems will likely benefit from large amounts of data as infrastructure is now coming online for large-scale data collection. One such example is the recent deployment of consumer smart

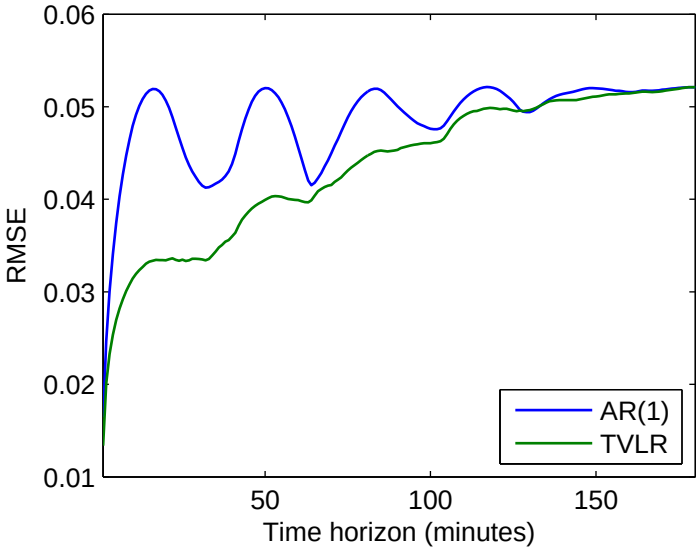

Figure 11: Comparison of prediction error vs. time horizon for the kernel density estimator with TVLR features and a static AR(1) model. The TVLR model significantly outperforms for predictions in the 1-60 minute window.

meters enabling real-time monitoring and collection of data at the single home level and resulting in time series datasets with millions of simultaneous points observed over several years.

This deluge of data motivates our desire to develop analysis methods for time series data that are both expressive and scalable. In this work, we consider the use of the total variation norm in building time-varying linear models that remain tractable to learn with convex methods. Convexity enables a rich class of algorithmic approaches and here we show that our problem, after appropriate analysis via the dual and the dual dual, is amenable to the classic projected Newton method [2]. Furthermore, we extend this approach by devising an active set method to exploit a high degree of sparsity that we expect to encounter in problems of interest.

More generally, time-varying linear regression can be seen as an example of a particular choice of the observational distribution $p(y_t|a_t; x_t)$—a standard Normal with fixed variance. We could potentially consider other more sophisticated observational models, an idea that we explore in a recent submission [7] which generalizes this work in proposing the use of the total variation norm as a convex surrogate for latent variable models such as HMMs. In addition, although the segmentation provided by by the total variation norm does not explicitly model recurrence, we augment the method with an additional step based on kernel density estimation similar to our approach for prediction in this work. Overall, we believe that the total variation norm and convex methods can lead to the development of fast, scalable algorithms for a large class of problems in time series analysis which have previously been addressed through the use of latent variables.

References

[1] C. M. Alaız, Á. Barbero, and J. R. Dorronsoro. Group fused lasso. In *International Conference on Artificial Neural Networks and Machine Learning–ICANN 2013*. Springer, 2013.

[2] D. P. Bertsekas. Projected newton methods for optimization problems with simple constraints. *SIAM Journal on control and Optimization*, 20(2):221–246, 1982.

[3] K. Bleakley and J.-P. Vert. The group fused lasso for multiple change-point detection. *arXiv preprint arXiv:1106.4199*, 2011.

[4] E. J. Candes, M. B. Wakin, and S. P. Boyd. Enhancing sparsity by reweighted 1 minimization. *Journal of Fourier analysis and applications*, 14(5-6):877–905, 2008.

[5] J. Z. Kolter and T. Jaakkola. Approximate inference in additive factorial hmms with application to energy disaggregation. In *International Conference on Artificial Intelligence and Statistics*, pages 1472–1482, 2012.

[6] H. Ohlsson, L. Ljung, and S. Boyd. Segmentation of ARX-models using sum-of-norms regularization. *Automatica*, 46(6):1107–1111, 2010.

[7] M. Wytock and J. Z. Kolter. Probabilistic segmentation vis total variation regularization. *In submission*.

[8] M. Wytock, S. Sra, and J. Z. Kolter. Fast Newton methods for the group fused lasso. *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence*, 2014.