Mach Learn (2016) 104:291–314
DOI 10.1007/s10994-016-5571-y

On the need for structure modelling in sequence prediction

Niall Twomey[1] · **Tom Diethe**[1] · **Peter Flach**[1]

Received: 13 March 2016 / Accepted: 21 June 2016 / Published online: 21 July 2016

Abstract There is no uniform approach in the literature for modelling sequential correlations in sequence classification problems. It is easy to find examples of unstructured models (*e.g.* logistic regression) where correlations are not taken into account at all, but there are also many examples where the correlations are explicitly incorporated into a—potentially computationally expensive—structured classification model (*e.g.* conditional random fields). In this paper we lay theoretical and empirical foundations for clarifying the types of problem which necessitate direct modelling of correlations in sequences, and the types of problem where unstructured models that capture sequential aspects solely through features are sufficient. The theoretical work in this paper shows that the rate of decay of auto-correlations within a sequence is related to the excess classification risk that is incurred by ignoring the structural aspect of the data. This is an intuitively appealing result, demonstrating the intimate link between the auto-correlations and excess classification risk. Drawing directly on this theory, we develop well-founded visual analytics tools that can be applied a priori on data sequences and we demonstrate how these tools can guide practitioners in specifying feature representations based on auto-correlation profiles. Empirical analysis is performed on three sequential datasets. With baseline feature templates, structured and unstructured models achieve similar performance, indicating no initial preference for either model. We then apply the visual ana-

Editors: Thomas Gaertner, Mirco Nanni, Andrea Passerini, and Celine Robardet.

Electronic supplementary material The online version of this article (doi:10.1007/s10994-016-5571-y) contains supplementary material, which is available to authorized users.

✉ Niall Twomey
niall.twomey@bristol.ac.uk

Tom Diethe
Tom.Diethe@bristol.ac.uk

Peter Flach
Peter.Flach@bristol.ac.uk

[1] Intelligent Systems Laboratory, University of Bristol, Bristol, UK

lytics tools to the datasets, and show that classification performance in all cases is improved over baseline results when our tools are involved in defining feature representations.

1 Introduction

Structure modelling permits target variables to collaborate so that 'informed' decisions about a set of random variables are based on a collection of beliefs linked together in a graphical structure (Lafferty et al. 2001; Sutton and McCallum 2011). In such frameworks, instances can be a list of vectors each relating to a single target variable in the graph. For classification problems, a localised belief about a particular target variable is influenced by the beliefs of neighbouring nodes, which, in turn, have been informed by their own neighbours. Marginal distributions in a structured model, therefore, are explicitly influenced by all possible target permutations over the graph. Intrinsically, this can be expensive to compute, but, in some applications, superior classification performance admonishes time complexity.

Some applications which have benefited from structural modelling include: computer vision (Zhang 2012) (*e.g.* scene recognition, item tracking), Activity Recognition (AR) (*e.g.* energy expenditure estimation), Natural Language Processing (NLP) (Collins 2002) (*e.g.* text chunking, information extraction), biomedical signal processing (Temko et al. 2011) (*e.g.* seizure detection) *etc.* With many of these, structure would be employed to model the temporal/sequential (and sometimes spatio-temporal) aspect of the data. In many of these example applications, however, the authors do not incorporate structure in the modelling pipeline, and so the structure is assumed to be approximated by the extraction of expressive features, although few researchers make this statement explicitly. The abandonment of structure might be considered sub-optimal for many of these applications, yet some are considered 'solved' with the unstructured model choice.

We can loosely view the structured and unstructured classifiers as being model- and data-driven respectively; model-driven can be seen to fit dynamics of the problem and data-driven can be seen to estimate a predictor for the problem with less attention given to modelling structure. The choice of approach is largely subjective and various practitioners approach the problem with both techniques. For example 'tracking by detection' (Andriluka et al. 2008) is a technique in computer vision where each frame in a video is considered independent, whereas filtering techniques, *e.g.* the Kalman filter (1960), can be applied to a history of predictions to estimate a trajectory to project tracking to future frames.

Despite the number of researchers that study structured problems, we cannot find specific studies where the efficacy and utility of both choices are compared over multiple classification domains. Some communities are satisfied with using static models, while others seem to insist on using structured models (*e.g.* many NLP applications).

This paper makes the following contributions. In Sect. 3 we discuss our methodology and approach. In this section we also demonstrate how Logistic Regression (LR) may be interpreted as a special case of Conditional Random Field (CRF) models. These are expanded upon with theoretical analyses in Sect. 4 where we derive bounds on the excess risk introduced when applying unstructured models to sequential problems. In Sect. 5 we describe the datasets and features used in our analyses, and we present our results in Sect. 6 where, for a number of datasets, we show that equivalent classification performance is achieved for structured and unstructured models alike. The theoretical and practical details of modelling sequences are both emphasised in detail throughout this paper. In particular Sect. 6 will introduce methods which relate our theoretical findings with practical experiments and we demonstrate that these can guide feature extraction routines to obtain improved classification performance for the datasets we considered. Finally, we discuss our contributions further and conclude in Sect. 7.

2 Related work

In this section, we discuss work that relates the use of structured and unstructured classification tasks which we outline from both practical and theoretical perspectives. In general, CRFs can be applied to any number of application domains, including NLP, bioinformatics, activity recognition, computer vision, *etc.* yet many practitioners in these areas have found that unstructured classification models can perform adequately. Examples of such applications include activity recognition (Twomey et al. 2010) (with specific reference to the Microsoft Kinect; Zhang 2012) and biomedical applications (*e.g.* brain (Temko et al. 2011) and heart (Twomey et al. 2014)). This is a principal motivation of our work, as methods have been derived for a number of application domains that are both structured and unstructured.

In Hoefel and Elkan (2008), the authors propose a two-stage CRF model. Their approach first learns LR or Support Vector Machine (SVM) models which are subsequently used as feature functions (see later) for the eventual CRFs. This work shows that CRFs with such feature functions tend to converge quickly due to the embedding of their discriminative characteristics.

Theoretical analysis of statistical learning has largely focused on identically distributed (iid) datasets, and this is a feature of many publications (Cristianini and Shawe-Taylor 2000). In Steinwart and Anghel (2009), the authors proposed the use of SVMs for forecasting on unknown ergodic systems. It was proved that with noisy observations, SVMs that incorporate Radial Basis Function (RBF) kernels will learn the best forecaster under alpha-mixing constraints when the decay of correlations for Lipshitz-continuous functions is summable.

In Sinn and Poupart (2011a), asymptotic theory relating to the consistency of linear-chain CRFs is introduced (with the assumption that the feature functions are known and that the weight parameters are not). This is used to describe parameter learning convergence of a sequence as its length tends towards infinity with maximum likelihood estimators. This required a definition of CRFs for infinite sequences that are defined by the limit distributions of conventional linear-chain CRFs. One of the main questions the authors answer is the quality of model identification in the presence of noisy data, and bounds were derived.

The investigation of infinite-sequence CRFs was continued in Sinn and Poupart (2011b) where theoretical considerations for online prediction are discussed. The work is motivated by the observation that marginal probability estimates can only be computed once a full data sequence has been observed, and this implies that the computation of exact marginal probabilities in online settings is not possible. The authors introduced methods of approximating the marginal distribution and provide theoretical bounds on the error rates on the approximations that can be calculated at run-time.

Rich notions of structure that can be captured by a first-order logical language are employed in relational learning and inductive logic programming. There, the idea of capturing structure in features rather than in models is called propositionalisation (Kramer et al. 2001; Krogel et al. 2003). This is a more general setting than ours as it can involve an unlimited range of structure including spatial structure (Appice et al. 2016), network structure (Schulte et al. 2016) and molecular structure (Kaalia et al. 2016). The advantage of our focus on sequential data is that it facilitates a more in-depth analysis of auto-correlation than would be possible with unrestricted logical structure (see also Jensen and Neville (2002) for a study on the effect of auto-correlation in relational learning).

Finally, we point the interested reader to a recent, complementary study on the tractability and optimality of structured prediction for 2D grids as commonly used in machine vision applications using a generative probabilistic model (Globerson et al. 2015).

3 Concepts and notation

3.1 Notation

In this work, we focus on non-iid sequential data. Each observation is a sequence of length N_m and each position of the sequence is a D-vector, *i.e.* $\mathbf{x}_m \in \mathbb{R}^{N_m \times D}$. Given a target label space, $\mathscr{Y}$, every sequence has an associated target vector, $\mathbf{y}_m \in \mathscr{Y}^{N_m}$. A dataset then consists of M observation-target pairs, $\mathscr{D} = \{(\mathbf{x}_m, \mathbf{y}_m)_{m=1}^{M}\}$. For the m-th observation, its n-th position is selected with $\mathbf{x}_{m,n}$ ('tokens') and the corresponding label for this position ('tags') is identified by $\mathbf{y}_{m,n}$.

Concretely, taking natural language as an example, $\mathbf{x}_m$ may represent a sentence of N_m words, $\mathbf{x}_{m,n}$ is a word in a sentence with the tag $\mathbf{y}_{m,n}$. In general, $\mathbf{x}_{m,n} \in \mathscr{V}$, for a fixed vocabulary $\mathscr{V}$. However, our analyses are not limited to this and our theoretical and empirical results hold with more general observation classes. Indeed, in all cases, predictive performance is more a function of the issued set of feature functions (see later) than the raw observations.

3.2 Models

3.2.1 Conditional random fields

Conditional Random Fields (CRFs) (Lafferty et al. 2001; Sutton and McCallum 2011) constitute a structured classification model of the distribution of $\mathbf{y}_m$ conditional on $\mathbf{x}_m$. The most common form of CRF is the linear-chain CRF which are applied to sequential data, *e.g.* natural language, but more general CRFs can be learnt on trees and indeed arbitrary structures. In general the probability distribution over the n-th node is influenced by the neighbouring nodes with graphical models, and this influence is propagated over the structure using algorithms based on message passing (Pearl 1982). In this section, we will show how marginal probability estimates are computed in the linear chain CRF framework efficiently, and we will also depict the message passing algorithm graphically.

The general equation for estimating the probability of a sequence is given by:

$$P_{\text{CRF}}(\mathbf{y}_m | \mathbf{x}_m) = \frac{1}{Z_{\text{CRF}}} \prod_{n=1}^{N_m} \exp\{\boldsymbol{\lambda}^{\top} \mathbf{f}(\mathbf{y}_{m,n-1}, \mathbf{y}_{m,n}, \mathbf{x}_m, n)\} \tag{1}$$

where N_m denotes the length of the m-th instance and n iterates over the sequence. The model requires specification of feature functions that are (often binary) functions of the current and previous labels, and (optionally) the sequence $\mathbf{x}_m$. We will discuss the curation of these feature functions later, but let us assume that a set of J feature functions exist. Both unigram feature functions that depend on the current label y_n ($\mathbf{f}_u(\emptyset, y_n, \mathbf{x}, n)$) and bigram feature functions that depend on the previous and current labels, y_{n-1}, y_n ($\mathbf{f}_b(y_{n-1}, y_n, \mathbf{x}, n)$) are allowed. We concatenate these into one vector $\mathbf{f}$ of length J for notational convenience. In many applications the set of non-zero feature functions is sparse for any position n allowing for fast computation even for large J. The set of feature functions has a corresponding set of parameters ($\boldsymbol{\lambda} \in \mathbb{R}^J$); these are learnt from data and are shared across potentials, meaning that the dynamics of the model do not change over time. Finally, Z_{CRF} is termed the partition function, and this normalises the output of the model to follow a true distribution. The graphical model for the CRF is shown for a short sequence in Fig. 1.

We will use the vectors $\boldsymbol{\alpha}_n$, $\boldsymbol{\beta}_n$, $\boldsymbol{\gamma}_n$, $\boldsymbol{\psi}_n$ and matrices $\boldsymbol{\Psi}_n$ during inference in CRFs. Subscripts are used to denote the position along the sequence, *e.g.* $\boldsymbol{\alpha}_n$ is a vector that pertains

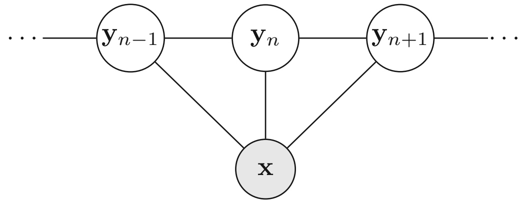


Fig. 1 This figure shows the graphical model for linear chain CRFs. Observed nodes are filled in *grey*, and this image shows how each node can depend on the whole sequence **x**

to the n-th position of the sequence, and parentheses are used to specify an element in the vectors, *e.g.* the y-th value of the n-th alpha vector is given by $\boldsymbol{\alpha}_n(y)$. Matrices are indexed by two positions, and the (i, j)-th element of $\boldsymbol{\Psi}_n$ is specified by $\boldsymbol{\Psi}_n(i, j)$.

In order to reduce the time complexity of inference, we describe a dynamic programming routine based on belief propagation here. We first calculate localised 'beliefs' about the target distributions, and these are called potentials. The accumulation of local potentials at node n is termed the 'node potential'. This $|\mathscr{Y}|$-vector where the y-th position is defined as $\boldsymbol{\psi}_n(y) = \exp\{\sum_{j=1}^{J} \boldsymbol{\lambda}_j \mathbf{f}_j(\emptyset, y, \mathbf{x}, n)\}$, where $\mathbf{f}_j$ is the j-th feature function. Similarly, the accumulation of local potentials at the n-th edge is termed the 'edge potential'. This is a matrix of size $|\mathscr{Y}| \times |\mathscr{Y}|$ where the (u, v)-th element is given by $\boldsymbol{\Psi}_n(u, v) = \exp\{\sum_{j=1}^{J} \boldsymbol{\lambda}_j \mathbf{f}_j(u, v, \mathbf{x}, n)\}$. Node potentials are depicted as the edges between observation and targets in Fig. 1, while in the same figure, edge potentials are depicted by edges between pairs of target nodes.

Given these potentials, we can apply the forward and backward algorithm on the CRFs chain. By defining the intermediate variables $\boldsymbol{\gamma}_{n-1} = \boldsymbol{\alpha}_{n-1} \odot \boldsymbol{\psi}_{n-1}$, and $\boldsymbol{\delta}_{n+1} = \boldsymbol{\beta}_{n+1} \odot \boldsymbol{\psi}_n$ (where $\odot$ denotes the element-wise product between vectors) the forward and backward vectors are recursively defined as:

$$\boldsymbol{\alpha}_n = \boldsymbol{\Psi}_{n-1}^{\top} \boldsymbol{\gamma}_{n-1} \tag{2}$$

$$\boldsymbol{\beta}_n = \boldsymbol{\Psi}_n \boldsymbol{\delta}_{n+1} \tag{3}$$

with the base cases $\boldsymbol{\alpha}_1 = \mathbf{1}$ and $\boldsymbol{\beta}_N = \mathbf{1}$. The un-normalised probability of the n-th position in the sequence can be calculated with

$$\widehat{P}(Y_n) = \boldsymbol{\alpha}_n \odot \boldsymbol{\psi}_n \odot \boldsymbol{\beta}_n. \tag{4}$$

Finally, in order to convert this to a probability distribution, values from (4) must be normalised by computing the 'partition function'. This is a real number, and may be calculated at any position n with $Z_{\text{CRF}} = \sum_{y' \in \mathscr{Y}} \widehat{P}(Y_n = y')$. The partition function is a universal normaliser on the sequence, and its value will be the same when computed at any position in the sequence. With this, we can now calculate the probability distribution on the n-th position

$$P(Y_n) = \frac{\widehat{P}(Yn)}{Z_{\text{CRF}}}. \tag{5}$$

Example 1 (Inference in CRFs) In Fig. 2 we show a graphical representation of inference for CRFs. We have overlaid the variables that we defined in this section on the graph, and, where appropriate, we also give the equations for the variables. From this image we can see that because $\boldsymbol{\alpha}_n$ is a function of $\boldsymbol{\alpha}_{n-1}$ (and indeed all elements in the set $\{\boldsymbol{\alpha}_m : 1 \leq m < n\}$), and because $\boldsymbol{\beta}_n$ is a function of all elements in $\{\boldsymbol{\beta}_m : n < m \leq N\}$, that probability estimation for node n is influenced by all node and edge potentials from the graph.

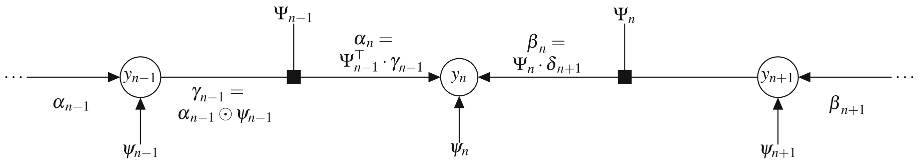

Ψ_{n-1}
$\alpha_n = \Psi_{n-1}^\top \cdot \gamma_{n-1}$
$\beta_n = \Psi_n \cdot \delta_{n+1}$
Ψ_n
y_{n-1}
y_n
y_{n+1}
α_{n-1}
$\gamma_{n-1} = \alpha_{n-1} \odot \psi_{n-1}$
β_{n+1}
ψ_{n-1}
ψ_n
ψ_{n+1}

Fig. 2 In this figure we show how marginal inference is performed over node y_n with CRFs models, where we have related the theoretical foundations of CRFs described in this section to a graphical representation of a short sequence. Note, the CRF is an undirected graphical model, and the arrows shown in this image indicate the direction of the passed messages when performing inference on y_n

3.2.2 Logistic regression

We can employ LR in a similar manner as CRFs to predict sequences. LR is formulated as follows:

$$P_{\text{LR}}(\mathbf{y}_{m,n}|\mathbf{x}_n) = \frac{1}{Z_{\text{LR}}} \exp\{\boldsymbol{\lambda}^\top \mathbf{f}(\mathbf{y}_{m,n}, \mathbf{x}_m, n)\} \tag{6}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^J$ are the parameters of the model that are associated with the unigram feature functions, $\mathbf{f}$, and Z_{LR} is the normalising constant. The set of feature functions for LR will be the same as that for CRFs with the exception that all bigram feature ($\mathscr{F}_b$) functions are zero.

Incorporating LR for sequence prediction assumes that position n of a sequence is unconditionally independent of all other positions of that sequence. This is clearly naïve assumption as neighbouring positions should provide additional information for probability estimates in sequences. However, if the order of the data is preserved and feature extraction captures the sequential nature of the data, LR may be capable of approximating the marginal distribution. Probabilities will be approximate, but can be computed with a significant reduction of computational complexity than CRFs.

To help us understand the use of LR for sequence prediction, we show in Theorem 1 that given certain conditions on transition potentials of CRFs, unconditional independence can be proved between adjacent nodes.

Theorem 1 *(Effect of rank-1 transition potentials on linear chains) Rank-1 transition potentials at any position $n(1 \leq n \leq N-1)$ of a chain induces unconditional independence between the portions of the chain preceding and following position n,* i.e. $P(Y_1, Y_2, \ldots Y_n) \perp\!\!\!\perp P(Y_{n+1}, Y_{n+2}, \ldots Y_N)$*. In the special case where all transition potentials are of rank 1, the joint probability of the chain may be exactly computed with $P(Y_1, Y_2, \ldots, Y_N) = P(Y_1)P(Y_2)\ldots P(Y_n)$.*

Proof Assuming a rank-1 incoming transition potential at position $n-1$, Singular Value Decomposition (SVD) can be employed to decompose $\boldsymbol{\Psi}_{n-1} = \sigma_{n-1}\mathbf{u}_{n-1}\mathbf{v}_{n-1}^\top$, where σ_{n-1} is the first singular value of $\boldsymbol{\Psi}_{n-1}$, and $\mathbf{u}_{n-1}$ and $\mathbf{v}_{n-1}$ are the first left and right singular vectors respectively. Given this decomposition, we can re-write the forward vectors as $\boldsymbol{\alpha}_n = (\sigma_{n-1}\mathbf{u}_{n-1}^\top \boldsymbol{\gamma}_{n-1})\mathbf{v}_{n-1}$. By noting that $(\sigma_{n-1}\mathbf{u}_{n-1}^\top \boldsymbol{\gamma}_{n-1})$ is a scalar which we will denote as c_{n-1}, un-normalised probability estimates can now be re-written as

$$\widehat{P}(Y_n) = c_{n-1}\mathbf{v}_{n-1} \odot \boldsymbol{\psi}_n \odot \boldsymbol{\beta}_n \tag{7}$$

and the partition function can be computed by marginalising over all possible labels, *i.e.* $Z_{\text{CRF}} = c_{n-1} \sum_{y' \in \mathscr{Y}} \mathbf{v}_{n-1}(y')\boldsymbol{\psi}_n(y')\boldsymbol{\beta}_n(y')$. The probability distribution over the labels at position n can now be written as

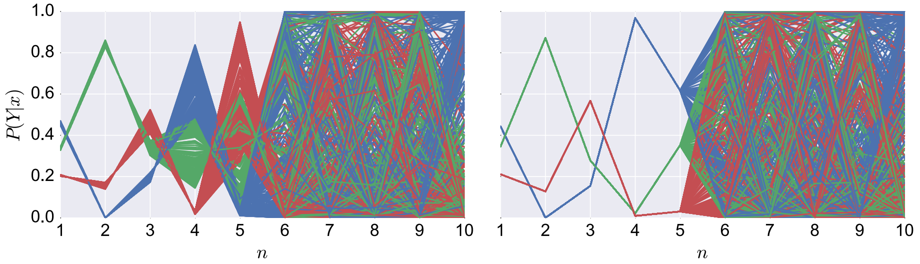


Fig. 3 This figure shows the effect of full rank transition potentials (*left*) and rank-1 transition potentials (*right*) on marginal probability estimation on CRFs. We can see that probability estimates at positions 1–5 are unaffected by this permutation when the rank of $\boldsymbol{\Psi}_5$ is 1

$$P(Y_n) = \frac{\mathbf{v}_{n-1} \odot \boldsymbol{\psi}_n \odot \boldsymbol{\beta}_n}{\sum_{y' \in \mathscr{Y}} \mathbf{v}_{n-1}(y')\boldsymbol{\psi}_n(y')\boldsymbol{\beta}_n(y')} \tag{8}$$

which no longer depends on the previous incoming forward vectors ($\boldsymbol{\alpha}$).

A similar approach will show that $P(Y_{n-1})$ is independent of all backward vectors when the $\mathtt{rank}(\boldsymbol{\Psi}_n) = 1$. Finally, if both $\mathtt{rank}\ (\boldsymbol{\Psi}_{n-1}) = 1$ and $\mathtt{rank}\ (\boldsymbol{\Psi}_n) = 1$, it follows that

$$P(Y_n) = \frac{\mathbf{v}_{n-1} \odot \boldsymbol{\psi}_n \odot \mathbf{u}_n}{\sum_{y' \in \mathscr{Y}} \mathbf{v}_{n-1}(y')\boldsymbol{\psi}_n(y')\mathbf{u}_n(y')} \tag{9}$$

which we can is independent of all forward and backward vectors due to the absence of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. □

The purpose of this analysis is to motivate the use of LR for sequence modelling by viewing it as a special case of CRFs. We do not necessarily advocate the use of SVD during learning/inference as it has time complexity $\mathscr{O}\left(|\mathscr{Y}|^3\right)$, while belief propagation requires $\mathscr{O}\left(|\mathscr{Y}|^2\right)$. Instead, the decomposition of Theorem 1 allows us to understand the connection between nodes and in particular the conditions where non-trivial transition potentials induce unconditional independence. Finally, we note that the use of SVD to detect unconditional independence with large (possibly loopy) graphs of binary variables may be advisable for non-active transition potentials (*i.e.* transition potentials that do not depend on $\mathbf{x}$; 'bias' transitions). In this case, the presence of rank-1 transition potentials may allow inference to be performed on a simpler graph that depicts equivalent marginal properties, and this condition can be encouraged by nuclear-norm regularisation (Recht and Fazel 2010).

Example 2 (Rank-1 transition potentials) In Fig. 3 we visually demonstrate effect of rank-1 transition potentials on a sequence. In both subplots in this figure, the log potentials beyond position 5 were then randomly permuted 250 times. When $\boldsymbol{\Psi}_5$ is of full rank (left), probabilities at positions 1–5 are dependent on the rest of the sequence. However, when the transition potential $\boldsymbol{\Psi}_5$ is compelled to have rank 1 (right) we can see that probability estimates at positions 1–5 are unaffected by potentials at positions 6–10.

4 Theoretical analysis

In this section we provide some theoretical results regarding learning on weakly dependent sequences, and where possible we give examples of definitions and concepts. The goal of this analysis is to show that under certain conditions, regularised Empirical Risk Minimisation (ERM) classifiers, of which LR is an example, are capable of achieving expected risk of forecasting on stochastic processes comparable to the risk achievable in a standard iid setting.

4.1 Preliminaries

We first introduce some basic definitions that will aid the following analysis.[1]

First we introduce the concept of a measure preserving dynamical system that will be used throughout. Let $\bigotimes = (\Omega, \Sigma, \mu)$ be a probability space for a set Ω, Σ a sigma-algebra on Ω and μ a probability measure. We similarly define a stochastic process (Ω, Σ, μ, T) where $T : \Omega \to \Omega$ is an endomorphism (measure-preserving transformation), meaning that T is surjective, measurable, and $\mu(T^{-1}A) = \mu(A)$ for all $A \in \Sigma$, where $T^{-1}(A)$ denotes the pre-image of A.

Definition 1 (Ergodicity; Walters 2000) An endomorphism T is called ergodic if it is true that $T^{-1}A = A$ implies $\mu(A) = 0$ or 1, where $T^{-1}A = \omega \in \Omega : T(\omega) \in A$.

Definition 2 (Stationarity) Let $\mathscr{X} = (X_i, Y_i)_{i \geq 0}$ be a stochastic $X \times Y$-valued process defined on the probability space (Ω, Σ, μ), with $X \subset \mathbb{R}^d$, $Y \subset \mathbb{R}$ are compact subsets, and $F_X(x_i)_{i=t_1+\tau,\ldots,t_k+\tau}$ represent the cumulative distribution function of the joint distribution of $\{X_t\}$ at times $t_1 + \tau, \ldots, t_k + \tau$. Then, $\{X_t\}$ is said to be stationary if, for all k, for all τ, and for all $t_1, \ldots, t_k$,

$$F_X(x_{t_1+\tau}, \ldots, x_{t_k+\tau}) = F_X(x_{t_1}, \ldots, x_{t_k}). \tag{10}$$

Since τ does not affect $F_X(\cdot)$, F_X is not a function of time.

Definition 3 (Regularity) Let μ be a measure on $\mathbb{R}^d$. μ is a regular Borel measure if for any two measurable sets $A, B \subset \mathbb{R}^d$, $\mu(A) = \mu(A + B) + \mu(A \setminus B)$, and if there exists a $B \in \mathbb{R}^d$ such that $A \subset B$ and $\mu(A) = \mu(B)$.

Definition 4 (Hölder continuity) A function f on $\mathbb{R}^d$ space is Hölder continuous, when there are non-negative real constants C, α, such that

$$|f(x) - f(y)| \leq C||x - y||^\alpha \tag{11}$$

for all x and y in the domain of f. α is the exponent of the Hölder condition. If $\alpha = 1$, then the function satisfies a Lipschitz condition. If $\alpha = 0$, then the function is simply bounded.

Example 3 (Regularity and Hölder continuity) In Fig. 4 we show a signal (blue, dotted) and the Hölder envelope (green, solid) at a position $x_0 = 0$. In this example, we can see that the signal never extends beyond the Hölder envelope, and consequently we can understand Hölder continuity is a measure of the regularity of a signal.

Definition 5 (Mixing) The transformation $T : X \to X$ is said to be mixing if for any two measurable sets $A, B \subset X$, one has $\mu(A \cap T^{-n}(B)) \to \mu(A)\mu(B)$ as $n \to \infty$. This property is closely related to the decay of correlations. If f is mixing, and iff correlations decay, $\text{cor}(\phi, \varphi) \to 0$ as $n \to \infty$, where

$$\text{cor}(\phi, \varphi) := \int_\Omega \phi \cdot \varphi d\mu - \int_\Omega \phi d\mu - \int_\Omega \varphi d\mu, \tag{12}$$

[1] The symbols α, β, γ, δ, and λ have alternate definitions when referenced in Sects. 3 and 4. While potentially confusing, we chose this notation in order to be consistent with the conventions of previous research in both areas.

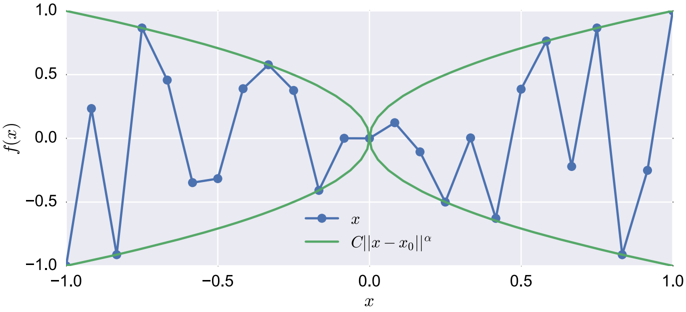


Fig. 4 A signal (*dotted blue line*) and its Hölderian envelope computed at x_0 (*solid green line*). As expected, the envelope bounds the signal, *i.e.* $|f(x) - f(x_0)| \leq C||x - x_0||^\alpha$ (Color figure online)

is the correlation of the square integrable functions $\phi, \varphi \in L_1(\mu)$ satisfying $\phi\varphi \in L_1(\mu)$.

Example 4 (Mixing) iid processes are mixing according to Definition 5 applied to finite dimensional cylinder sets (open sets of the natural topology of sequences of random variables). Ergodic Markov chains are also mixing (such as the Occasionally Dishonest Casino (ODC) example that we will analyse in Sect. 4.3). Generally, any strictly stationary, finite or countable-state aperiodic Markov chain is mixing.

Definition 6 (Strong mixing) Suppose $X := (X_k, k \in \mathbb{Z})$ is a sequence of random variables on a given probability space (Ω, Σ, μ). For $-\infty \leq j \leq \ell \leq \infty$, let Σ_j^ℓ denote the σ-field of events generated by the random variables $X_k,\ j \leq k \leq \ell\ (k \in \mathbb{Z})$. For any two σ-fields $\mathscr{A}$ and $\mathscr{B} \subset \Sigma$, define the 'measure of dependence'

$$\alpha(\mathscr{A}, \mathscr{B}) := \sup_{A \in \mathscr{A}, B \in \mathscr{B}} |\mu(A \cap B) - \mu(A)\mu(B)|.$$

For the given random sequence X, for any positive integer n, define the dependence coefficient $\alpha(n) = \alpha(X, n) := \sup_{j \in \mathbb{Z}} \alpha(\Sigma_{-\infty}^j, \Sigma_{j+n}^\infty)$. By a trivial argument, the sequence of numbers $(\alpha(n), n \in \mathbb{N})$ is non-increasing. The random sequence X is said to be 'strongly mixing', or 'α-mixing', if $\alpha(n) \to 0$ as $n \to \infty$.

Definition 7 (Lipschitz Loss) Let the function $L : X \times Y \times \mathbb{R} \to [0, \infty)$ be a convex, differentiable and locally Lipschitz continuous loss function, and it also satisfies $L(x, y, 0) \leq 1$ for all $(x, y) \in X \times Y$. Moreover, for the derivative L' there exists a constant $c \in [0, \infty)$ such that for all $(x, y, t), (x', y', t') \in X \times Y \times \mathbb{R}$ we have $|L'(x, y, 0)| \leq c$ and $|L'(x, y, t) - L'(x', y', t')| \leq c \left\|(x, y, t) - (x', y', t')\right\|_2$.

Definition 8 (Linear Classifiers) Given $X \subset \mathbb{R}^d$ and $Y \subset \mathbb{R}$, and a measurable function L as defined above. For a finite sequence $T = ((x_1, y_1), \ldots, (x_n, y_n)) \in (X \times Y)_n$ and a function $f : X \to \mathbb{R}$, we define the empirical L-risk by $\mathscr{R}_{L,T}(f) := \frac{1}{n} \sum_{i=0}^{n-1} L(x_i, y_i, f(x_i))$. For a distribution P on $X \times Y$, we write $\mathscr{R}_{L,P}(f) := \int_{X \times Y} L(x, y, f(x)) dP(x, y)$ and $\mathscr{R}_{L,P}^* := \inf \left(\mathscr{R}_{L,P}(f) | f : \mathbb{R}^d \to \mathbb{R}^d\right)$ for the L-risk and minimal L-risk associated to P. Let Λ be a stable regulariser on $\mathscr{F}$, that is, a function $\Lambda : \mathscr{F} \to [0, \infty)$ with $\Lambda(0) = 0$. We will also require the following:

$$r^* := \inf \Lambda(f) + \mathscr{R}_{L,P}(\hat{f}) - \mathscr{R}_{L,P}^*, \quad r > r^* \tag{13}$$

and

$$\mathscr{F}_r := \left\{ f \in \mathscr{F} : \Lambda(f) + \mathscr{R}_{L,P}(\hat{f}) - \mathscr{R}^*_{L,P} \le r \right\}. \tag{14}$$

giving $r^* \le 1$ since $L(x, y, 0) \le 1, 0 \in \mathscr{F}$, and $\Lambda(0) = 0$. We also assume there is a monotonically decreasing sequence $(A_r)_{r\in(0,1]}$ such that

$$\|L \circ f\| \le A_r \quad \forall f \in \mathscr{F}_r, r \in (0, 1] \tag{15}$$

Because of Eq. 14 we have that $\left\| L \circ \hat{f} \right\| \le A_1 \forall f \in \mathscr{F}$ and $r \in (0, 1]$. Finally assume there exists a function $\varphi : (0, \infty) \to (0, \infty)$ and a $p \in (0, 1]$ such that, $\forall r > 0$ and $\varepsilon > 0$, we have

$$\log \mathscr{N}(\mathscr{F}_r, \|\cdot\|_\infty, \varepsilon) \le \varphi(\varepsilon) r^p. \tag{16}$$

We will first use a result regarding the consistency of SVM for forecasting the evolution of an unknown ergodic dynamical system from observations with unknown noise (Steinwart and Anghel 2009) that can easily be extended to LR. We firstly restate assumptions S1 and S2 from Steinwart and Anghel (2009) monotone sequences:

Assumption 1 For a fixed strictly positive sequence $(\gamma_i)_{i\ge0}$ converging to 0 and a locally Lipschitz continuous loss L the monotone sequences $(\lambda_n) \subset (0, 1]$ and $(\sigma_n) \subset [1, \infty)$ satisfy $\lim_{n\to\infty} \lambda_n = 0$, $\sup_{n\ge1} e^{-\sigma_n} |L|_{\sigma_n^{-1/2},1} < \infty$,

$$\sup_{n\ge1} \frac{\lambda_n \sigma_n^{4d}}{|L|_{\sigma_n^{-1/2},1}} < \infty \quad \text{and} \quad \lim_{n\to\infty} \frac{|L|^3_{\sigma_n^{-1/2},1} \sigma_n^2}{n\lambda_n^4} \sum_{i=0}^{n-1} \gamma_i = 0.$$

Assumption 2 For a fixed strictly positive sequence $(\gamma_i)_{i\ge0}$ converging to 0 and a locally Lipschitz continuous loss L the monotone sequences $(\gamma_n) \subset (0, 1]$ and $(\sigma_n) \subset [1, \infty)$ satisfy $\lim_{n\to\infty} \lambda_n \sigma_n^d = 0$,

$$\lim_{n\to\infty} \frac{\lambda_n \sigma_n^{4d}}{|L|_{\sigma_n^{-1/2},1}} = \infty \quad \text{and} \quad \lim_{n\to\infty} \frac{\sigma_n^{2+12d}}{n\lambda_n} \sum_{i=0}^{n-1} \gamma_i = 0.$$

These assumptions define two complementary conditions: the first implies that λ_n should tend to zero, and the other is that it should not decay too fast. This in turn ensures that both the approximation error and statistical error decay to zero, which is as we would expect for consistent classifiers (see Steinwart and Anghel (2009) for details).

Theorem 2 *(Consistency of LR) Let $\mathscr{X}$ be a stochastic process as defined in Definition 2. We write $P := \mu(X_0, Y_0)$ and assume that $\mathscr{X}$ has a decay of correlations of some order (γ_i). In addition, let $L : X \times Y \times \mathbb{R} \to [0, \infty)$ be the logistic loss $L(t) = \log(1+\exp(-t))$. Then for all sequences $(\lambda_n) \subset (0, 1]$ and $(\sigma_n) \subset [1, \infty)$ satisfying Assumptions S1 and S2 from Steinwart and Anghel (2009) and all $\varepsilon \in (0, 1]$ we have*

$$\lim_{n\to\infty} \mu(\omega \in \Omega : |\mathscr{R}_{L,P}(f_{T_n}(\omega), \lambda_n, \sigma_n) - \mathscr{R}^*_{L,P}| > \varepsilon) = 0$$

where $T_n(\omega) := ((X_0(\omega), Y_0(\omega)), \ldots, (X_{n-1}(\omega), Y_{n-1}(\omega)))$ and $f_{T_n}(\omega), \lambda_n, \sigma_n$ is the LR forecaster defined by Eq. 6.

Proof This is an application of Theorem 2.4 of Steinwart and Anghel (2009) to LR using the fact that LR and SVMs are both Lipschitz continuous up to a change in constants (Rosasco et al. 2004), and since the logistic loss satisfies Definition 7 as well as assumptions 1 and 2 it also satisfies Theorem 2.4. □

Using the smoothness assumptions on the map $T : M \to M, M \subset \mathbb{R}^d$ as defined in Definition 5, and restricting the measure μ to be a Lebesgue outer measure on $\mathbb{R}^d$ (*i.e.* it satisfies Definition 3, the LR is consistent in the sense of Theorem 2 and has a rate of convergence that is related to the the rate of mixing of the stochastic process $\mathscr{Z}$ (or alternatively, rate of decay of the auto-correlations). Theorem 2 applies to stochastic processes that are α-mixing with rate (γ_i). However, there are interesting stochastic processes that are not α-mixing but still have fast decay of correlations. We now introduce C-mixing processes (Hang and Steinwart 2015), which make weaker assumptions than the strong mixing used thus far.

Definition 9 (C-Mixing; Hang and Steinwart 2015) Given a semi-norm $\|\cdot\|$ on a vector space E of bounded measurable functions $f : Z \to \mathbb{R}$, we define the C-Norm by $\|f\|_C := \|f\|_\infty + \|f\|$ and denote the space of all bounded C-functions by $C(Z) := \{f : Z \to \mathbb{R} | \; \|f\|_C < \infty\}$. Some examples of semi-norms that can be used for $\|f\|$ are given in Hang and Steinwart (2015). Let (Ω, Σ, μ) be a probability space, (Z, B) be a measurable space, $\mathscr{Z} := (Z_i)_{i \geq 0}$ be a Z-valued, stationary process on Ω with a C-norm $\|\cdot\|_C$, then for $n \geq 0$ we define the C-mixing coefficients by:

$$\phi_C(\mathscr{Z}, n) := \sup \left\{ \text{cor}(\psi, h \circ Z_{k+n}) : k \geq 0, \phi \in B_{L_1(A_0^k, \mu)}, h \in B_{C(Z)} \right\} \tag{17}$$

with the time reversed coefficients

$$\phi_{C,rev}(\mathscr{Z}, n) := \sup \left\{ \text{cor}(h \circ Z_k, \varphi) : k \geq 0, h \in B_{C(Z)}, \varphi \in B_{L_1(A_{k+n}^\infty, \mu)} \right\}. \tag{18}$$

Let $(d_n)_{n \geq 0}$ be a strictly positive sequence converging to 0. Then we say $\mathscr{Z}$ is C-mixing with rate $(d_n)_{n \geq 0}$ if $\phi_{C,(rev)}(\mathscr{Z}, n) \leq d_n \forall n \geq 0$. Moreover, if $(d_n)_{n \geq 0}$ is of the form

$$d_n := c \exp(-bn^\gamma), \quad n \geq 1, \tag{19}$$

for some constants $b > 0, c \geq 0, \gamma > 0$, then $\mathscr{Z}$ is called geometrically time-reversed C-mixing.

Example 5 (Bounded variation and C-Mixing) If we take as an example of the semi-norm $\|f\| = \|f\|_{BV(Z)}$, where $BV(Z) = \sup \int f(dZ)/(dx)$, *i.e.* the total variation is bounded, then it is well know that $BV(Z)$ together with $\|f\|_\infty$ forms a Banach space, and satisfies the conditions of a C-norm. Examples of such functions are given in Fig. 5, and some further examples of C-mixing processes are given in Hang and Steinwart (2015), along with relations to well-known results on the decay of correlations of dynamical systems.

4.2 Learning rates

Thus far we have only shown that the risk of the LR solution converges to the smallest possible risk. However, for practical considerations the speed of this convergence is of great importance. In order to use the above analysis to get a rate of convergence, we need to place

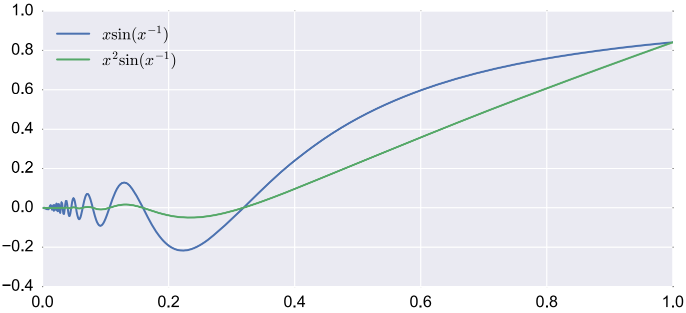


Fig. 5 On the interval [0,1], the function $x^2 \sin\left(x^{-1}\right)$ is of bounded variation, but $x \sin\left(x^{-1}\right)$ is not

additional restrictions on T and μ to give us a quantitative version of Theorem 2. We will now give learning rates for regularised ERM classifiers on C-Mixing processes.

Theorem 3 *Let $Z := (Z_n)_{n>0}$ be a Z-valued stationary geometrically (time-reversed) C-mixing process on (Ω, Σ, μ, T) with rate $(d_n)_{n\geq 0}$ and $\|\cdot\|_C$ as defined in Definition 9, and $P := \mu Z_0$. Moreover, let L be a loss satisfying Definition 7. In addition assume that there exists a Bayes decision function $f^*_{L,P}$, we have that*

$$\mathbb{E}_P\left(L(x, y, f) - L(x, y, f^*_{L,P})\right)^2 \leq 16\left(\mathbb{E}_P\left(L(x, y, f) - L(x, y, f^*_{L,P})\right)\right), \quad f \in \mathscr{F}, \tag{20}$$

where $\mathscr{F}$ is a hypothesis set with $0 \in \mathscr{F}$. Defining $r^*, \mathscr{F}_r$, and A_r by (13), (14), (15) respectively and assume that Eq. 16 holds. Finally, let $\Lambda : \mathscr{F} \to [0, \infty)$ be a regulariser with $\Lambda(0) = 0$, $f_0, f_1 \in \mathscr{F}$ be fixed functions, and $A_0, A_1, A^* \geq 0$, $B_0 \geq 1$ be constants such that $\|L(x, y, f_0(x))\| \leq A_0$, $\|L(x, y, f_1(x))\| \leq A_1$, $\left\|L(x, y, f^*_{LP})\right\| \leq A^*$ and $\|L(x, y, f_0(x))\|_\infty \leq B_0$. Then for all fixed $\varepsilon > 0$, $\delta \geq 0$, $\tau \geq 1$ and

$$n \geq n^*_0 := \max\left\{\min\left\{m \geq 3 : m^2 \geq K \text{ and } \frac{m}{(\log(m))^2/\gamma} \geq 4\right\}, e^{3/b}\right\} \tag{21}$$

with $K = 1212c(4A_0 + A^* + A_1 + 1)$, and $r \in (0, 1]$ satisfying

$$r \geq \max\left\{\frac{c_V(\log(n))^{2/\gamma}(\tau + \log(4/\varepsilon)2^p r^{2p}}{n}, \frac{20(\log(n))^{2/\gamma} B_0\tau}{n}\right\} \tag{22}$$

with $c_v = 32938\frac{2}{3}$, every learning method defined by Definition 8 satisfies with probability μ not less than $1 - 16e^{-\tau}$:

$$\Lambda(f_{D_n}) + \mathscr{R}_{L,P}(f_{D_n}) - \mathscr{R}^*_{L,P} < 2\Lambda(f_0) + 4\mathscr{R}_{L,P}(f_0) - 4\mathscr{R}^*_{L,P} + 4dr + 5d\varepsilon + 2d\delta. \tag{23}$$

Proof This is a direct application of the bound of Hang and Steinwart (2015, Theorem 4.10), with some minor modifications. Firstly, since we are interested in classification rather than regression, and we can without loss of generality shift our classifier outputs away from zero and one by some small epsilon, we are not concerned with the possibility that our predictor can incur arbitrarily large loss for any given example. This allows us to drop the clipping restriction required by the theorem, and instead fall back on linear classifiers that have Lipschitz bounded loss functions defined in Definition 7. Further to this, we are here interested in a simpler class of Hilbert spaces than that induced by the Gaussian kernel, so we can instead use the covering number of linear or polynomial kernels (which can be seen as equivalent to n-gram type features when neighbouring data points are concatenated). Since we know from Li and Wang (2009) that the covering number for a polynomial kernel is given by $\log \mathscr{N}(\varepsilon) \leq (d+1)\log(4/\varepsilon)$, $\forall \varepsilon > 0$, it is easy to see that the covering number for linear kernels is $\log \mathscr{N}(\varepsilon) \leq 2\log(4/\varepsilon)$. □

It is worth noting that these bounds are very similar to the Bernstein type bounds achievable for iid processes. The implication is that, given mild assumptions on the nature of the dependence of the underlying process (*i.e.* a geometric rate of decay of correlations), we will be able to learn a classifier that in the limit will behave as if the data were indeed iid, with the rate of convergence being directly related to the rate of decay of correlations of the underlying process. Intuitively, this makes sense: if the decay is sufficiently fast, we have a high probability that, given a 'current' example and another randomly selected example, there will be virtually no dependence between them, so the standard theory then holds. Furthermore, it implies that if we construct features that capture some notion of the context, such as the n-gram feature templates discussed herein, we can accurately capture the full dependency structure of the sequence, and that the faster the rate of decay of correlation, the smaller the resulting feature templates need to be.

In this analysis, we have considered linear measures of correlation. It is also possible to consider nonlinear measures of dependence in the time series, such as the non-parametric extension of Kendall's Tau for sequences (Ferguson et al. 2011), or other nonlinear rank-based measures (Naro et al. 2014). Whilst this is outside the scope of this work, this would be an interesting area of investigation from both theoretical and experimental perspectives.

Interestingly, however, linear measures of correlation are valid for many stochastic (including chaotic) processes that display highly nonlinear behaviour, as they will still have bounded auto-correlation. A common example is the set of Lipshitz continuous functions, which are a special case of these C-Mixing processes (*c.f.* Definition 9 and Example 5).

In the following, as a concrete example of a stochastic process to which this theory can be applied, we analyse the Occasionally Dishonest Casino (ODC), giving a method to quantify the rate of convergence of correlations based on the parameter settings used to define the sequence. We empirically analyse this setting in Sects. 5 and 6.

4.3 Auto-correlation of the occasionally dishonest casino

Markov's theorem tells us that a Markov chain is ergodic if there is a strictly positive probability to pass from any state to any other state in one step, so by construction the ODC as defined in Theorem 4 satisfies ergodicity. Furthermore, by the definition of stationarity given in Eq. 10, by construction the ODC is also a stationary system. Following on from this, we give an example of quantifying the expected auto-correlation for the ODC as:

Theorem 4 *An Occasionally Dishonest Casino (ODC) uses two kinds of die. Define the set of outcomes* S, e.g. *for a 6-sided die* $S = \{1, 2, 3, 4, 5, 6\}$. *A fair die has* $\frac{1}{|S|}$ *probability of rolling any number, and a loaded die that has* p_v *probability to roll a value* $v \in S$ *and* $p_{\sim v} = \frac{1-p_v}{|S|-1}$ *probability to roll each of the remaining numbers. We will use the notation* $\Sigma_S = \sum_{x \in S} x$ *to denote the sum of the possible outcomes in* S, *and* $\Sigma_{S\setminus v} = \sum_{x \in S\setminus\{v\}} x$. *Assume a symmetric probability* p_s *that the casino switches from fair to loaded die and back. The expected auto-correlation* R *of the discrete time process depending on* v, p_v *and* p_s, *and for a six-sided die* $S = \{1, 2, 3, 4, 5, 6\}$, *is given by:*

$$R(v, p_v, p_s) = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \left[\frac{2\,(X_t - C)\,(X_{t+n} - C)}{\frac{35}{12} + p_v(v - D)^2 + \frac{p_{\sim v}}{5} \sum_{x \in S\setminus\{v\}} (x - D)^2} \right].$$

$$\textit{where} \quad C = \frac{7 - 2D}{4}, \quad D = vp_v - \frac{21 - v}{5} p_{\sim v} \tag{24}$$

Proof See supplementary material. □

This analysis has related the learning rates of linear classifiers such as LR of the decay in correlations in the sequence, which motivates the empirical use of auto-correlation as a sensible quantity to estimate when deciding whether or not a structured model is required. There are two main factors affecting the decay of correlations in a sequence: the strength of the chaos in the underlying dynamical system $g : X \to X$, and the *regularity* of the observables F and G. Generally speaking, the correlations decay rapidly if the system is strongly chaotic and the observations are sufficiently regular (*e.g.* systems that are Hölder continuous—see Definition 4). We shall see that many real-world problems that have been considered to be sequential classification tasks, and hence 'requiring' structured models, in fact do exhibit the rapid decays in auto-correlation required by the theory.

5 Features, datasets and experiments

In this section we describe feature extraction methodology and datasets used for our empirical results.

5.1 Features

Feature are often specified with so-called 'feature templates' in sequential classification. This is a powerful framework as it allows the practitioner to abstractly define the form of features instead of manually curating them explicitly.

We extract n-gram features from our datasets as a proxy for encoding sequential information. For example, the templates $f_{\langle -1,0\rangle}$ and $f_{\langle 0,1\rangle}$ specify that, for every position n in the input sequence, the feature $f_{\langle -1,0\rangle}$ will be the concatenation of the $(n-1)$-th and n-th value in the input sequence, and the feature $f_{\langle 0,1\rangle}$ will be a concatenation of the n-th and $(n+1)$-th values. In this work we employ n-grams of up to length 5, and we also extract long-range 'skip-grams' (*i.e.* conjunctions of non-contiguous positions) which can capture long-range dependencies.

The exact form of the feature templates used in our analyses will be clarified in the next sections. Our data is generally discrete from a finite vocabulary, $\mathscr{V}$, meaning that the feature functions in this analysis return binary values. However, this is not a limitation of our framework and real-valued and continuous data (*e.g.* accelerometer, physiological signals, images) can be considered by our analysis by incorporating sparse coding techniques, for example. In general, CRFs are entirely agnostic to the operations that are performed on the data so long as real-valued numbers are returned.

5.2 Datasets

5.2.1 Word hyphenation

Word Hyphenation (WH) (or orthographic syllabification) is the process of separating words into their constituent syllables, and the boundaries between syllables are a natural position

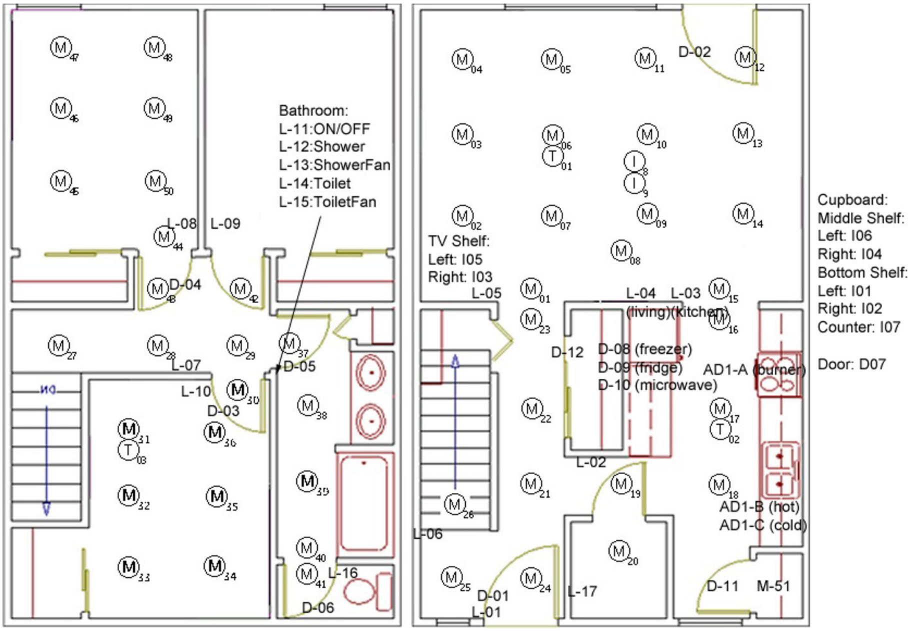


Fig. 6 Floorplan of the `twor.2009` smart home. Motion sensors (prefixed with `m`) are regularly distributed throughout the house. As residents move throught the smart home, these sensors detect motion and trigger between the 'off' (no motion) and 'on' (motion) states

for hyphens. This is a pre-processing step in a number of different tasks. Trogkanis and Elkan (2010) posed this problem as a sequential binary classification task using linear-chain CRFs. To represent the problem, the researchers used the feature template system described earlier considering all 15 contiguous sub-strings up to length 5 that include the n-th position:

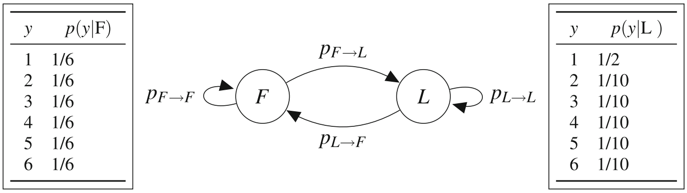
$$\mathscr{F}_{\mathrm{H}} = \{f_{\langle 0\rangle}, f_{\langle -1,0\rangle}, f_{\langle 0,1\rangle}, \ldots, f_{\langle -1,0,1,2,3\rangle}, f_{\langle 0,1,2,3,4\rangle}\}. \tag{25}$$

Excellent prediction was obtained for English and Dutch corpora. In our results section, we assess the classification performance on the English corpus (the harder task based on performance evaluation, consisting of approximately 80 % negatives) with $\mathscr{F}_{\mathrm{H}}$.

5.2.2 Smart home activity recognition

The Centre for Advanced Studies in Adaptive Systems (CASAS) research group focus on many aspects of Activity Recognition (AR) in smart environments, and provide a number of annotated datasets. We consider the hand-segmented data from the `twor.2009`[2] multiresident dataset (Cook and Schmitter-Edgecombe 2009) to allow a focus on activity recognition instead of other AR challenges, such as activity segmentation. This dataset was recorded 'in the wild' where various sensors placed in the home (*e.g.* motion, temperature, door sensors were present in all rooms in the house) were activated when a resident performed Activities of Daily Living (ADL), and activities are predicted based on the patterns sensor activation, see Fig. 6. Annotations were applied retrospectively by domain experts.

[2] http://casas.wsu.edu/datasets/twor.2009.zip



y	$p(y\|\text{F})$
1	1/6
2	1/6
3	1/6
4	1/6
5	1/6
6	1/6

y	$p(y\|\text{L})$
1	1/2
2	1/10
3	1/10
4	1/10
5	1/10
6	1/10

Fig. 7 The ODC drawn as a graphical model. Here, we have fair and loaded states, and the probability distribution over a die is shown for both in a conditional probability table. Data is generated by randomly walking through the automaton

Fifteen activities are labelled in this dataset, and labels with fewer than two occurrences were removed. Some ADLs will 'look' the same from a sensor activation point of view (*e.g.* meal preparation and washing up). Therefore, the principal difficulty of AR lies in discriminating between 'similar' activities and in identifying the correct resident to the predicted activity.

We represent this data in an atomic state-change manner, *e.g.* $\mathbf{x}_n = (\texttt{m16\_on}, \texttt{m15\_on}, \texttt{m17\_on}, \texttt{m15\_off}, \ldots)$ would be a sequence of sensor activations that would be predomenently in the kitchen (lower right hand side of Fig. 6), and we make predictions for all events. With this representation we can readily apply feature templates discussed earlier. Because 'breakfast' sensor activities will resemble 'lunch' and 'dinner' activities, we further adjust feature specification by adding 1-of-24 categorical hour of day features.

5.2.3 Occasionally dishonest casino

The Occasionally Dishonest Casino (ODC) is a well-known hypothetical scenario in which a die can transition between fair (F) and loaded (L) states. When in the fair state, a uniform discrete probability distribution is imposed on the die, and when in the loaded state the die will roll to its biased face with probability p_b, and to its remaining faces with probability $(1 - p_b)/5$.

The ODC is depicted by the automaton in Fig. 7 in which the biased face is selected as 1. The task we choose is to predict when the die is in a fair state given only a sequence of face observations. To generate an instance, we randomly walk through the automaton according to the 'transition' and 'emission' probabilities. Each walk consists of M_n 'rolls' ($M_n \backsim \text{Poisson}(\lambda)$), and we set $\lambda = 100$ arbitrarily. We reduce the degrees of freedom of this model to one by imposing symmetric transition probabilities. A dataset consists of N random walks, and we have set $N = 2\,000$, $p_t = 0.05$, and $p_b = 0.5$ (following Murphy (2012). The class distribution is balanced due to the symmetric transition probabilities.

5.3 Performance evaluation

Given a set of ground truth labels and classifier predictions, we can define predictions as being True Positives (TPs), True Negatives (TNs), False Positives (FPs), or False Negatives (FNs). By accumulating these over a dataset, we can compute various accuracy metrics, including precision, recall as follows:

$$\text{precision} = \frac{\#TP}{\#TP + \#FP} \tag{26}$$

$$\text{recall} = \frac{\#TP}{\#TP + \#FN} \tag{27}$$

Precision and recall are accuracy metrics, and these averaged by calculating their harmonic mean, which yields the F-Score:

$$\text{F}_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{28}$$

The F-Score relates to classification accuracy and ignores the effect of the true negative examples, and its utility as an accuracy measure is well documented (Provost et al. 1998).

Finally, we also compute the Brier score (Brier 1950):

$$BS = \frac{1}{N} \sum_{n=1}^{N} \sum_{c=1}^{C} w_c \left(p_{n,c} - y_{n,c}\right)^2 \tag{29}$$

where N is the number of test sequences, C is the number of classes, w_c is the weight for each class, $p_{n,c}$ is the predicted probability of instance n being from class c, and $y_{n,c}$ ground truth label. Lower Brier score values indicate better performance, with optimal performance achieved with a Brier score of 0.

We perform 10-fold cross validation on all experiments, and results report the mean and standard deviation calculated on the test-folds. Hyperparameters are kept at their default values for all experiments.

Generally, it is assumed that for sequential tasks CRFs will perform significantly better than non-sequential models, such as LR. In our results, we compare the performance of CRF and LR models with statistical hypothesis testing. Note that for experiments that yield insufficient evidence to reject the null hypothesis indicate that we should not prefer CRFs over LR.

5.4 Experiments conducted

Our first experiments assess the difference in classification performance between LR and CRF models over the Word Hyphenation (WH), Activity Recognition (AR) and Occasionally Dishonest Casino (ODC) datasets (described previously). With the ODC dataset, we also show how LR models can approximate the 'smoothing' behaviour that one can achieve in sequential models (*e.g.* CRFs).

We demonstrate relationship between our main theoretical results with practical experiments. We have already shown the empirical effect of rank-1 transition potentials on inference with CRFs in Example 2. Second, we show how the auto-correlation and its rate of decay can be employed to glean insight into the characteristics of sequential data. This is then used to guide the specification of feature templates in a manner that demonstrably improves classification performance.

Finally, we perform analyses that investigate classification performance with increasingly expressive feature representations (which we term 'incremental performance assessment'). To do this, we will assume that the feature templates are ordered by increasing expressivity (n-gram templates are more expressive than $(n-1)$-gram templates). The incremental subsets will consider up to c templates, and are denoted by $\mathscr{F}_{\mathrm{H}}^{c} \subseteq \mathscr{F}_{\mathrm{H}}$. Using the WH feature templates as an example, $\mathscr{F}_{\mathrm{H}}^{2} = \{f_{\langle 0 \rangle}, f_{\langle -1,0 \rangle}\}$, and $\mathscr{F}_{\mathrm{H}}^{4} = \{f_{\langle 0 \rangle}, f_{\langle -1,0 \rangle}, f_{\langle 0,1 \rangle}, f_{\langle -1,0,1 \rangle}\}$.

6 Results and discussion

6.1 LR and CRF classification performance

6.1.1 WH/AR/ODC

Table 1 shows the averaged F-Score and Brier score of the CRF and LR models for the WH and AR datasets. LR models performed marginally better than CRF models on both F-Score and Brier score metrics on the WH task. We conducted two-way Analysis of Variance (ANOVA) to determine if any of the differences reached statistical significance, but found that all main effects failed to reach significance at the $p < 0.01$ level. Indeed, the lack of statistical significance is suggestive that neither model should be preferred. We obtained similar performance to those from the original paper (Trogkanis and Elkan 2010) with both LR and CRF classification models.

Table 1 summarises the results for the AR dataset. It is worth noting that our results are competitive with those obtained by a number of AR researchers with $\mathscr{F}_{\mathrm{H}}$ feature templates even though these were not designed for AR. Classification performance is improved substantially against the majority-class classifier. We also note that variance of predictions is quite large for the AR results, and this is due to sample size and the sparsity of some labels.

Figure 8 shows the probability estimates obtained by CRF and LR models for a particular sub-sequence of die rolls of the ODC dataset. The upper bar chart shows the faces that were rolled (blue fill indicates the fair state whereas red fill indicates the biased state). In the lower image, the red line gives the probability estimates from the CRF model, and the blue line gives those of the LR model. The biased face is 1.

While we are attempting to estimate the probability of bias in this example, we do not necessarily desire 'responsive' changes in these probability estimates. Such changes would likely be indicative of overfitting because realisations of the biased face are always possible in both biased and unbiased states. Instead, we wish for probability estimates between neighbouring

Table 1 Classification performance of CRF and LR models on WH dataset and AR datasets

Dataset	Model	F-Score	Brier score
WH	CRF	0.963 ± 0.0014	0.012 ± 0.0004
	LR	0.966 ± 0.0012	0.011 ± 0.0003
AR	CRF	0.7980 ± 0.0445	0.0222 ± 0.0044
	LR	0.7876 ± 0.0439	0.0209 ± 0.0039

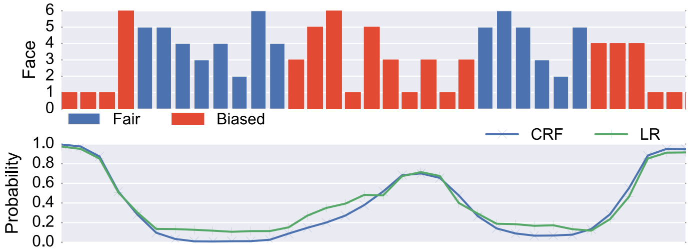


Fig. 8 Marginal probability estimates of bias with LR and CRF models for a sequence of die rolls. Both models follow similar general trends indicating that LR probability estimation approximates CRF smoothing

positions to be correlated, due to the parameterisation of p_t in the ODC (*c.f.* Sect. 5.2.3). For applications that require decisions rather than probability estimates, we would recommend computing the Viterbi path (Viterbi 1967) through the sequence rather than thresholding these probability estimates as the Viterbi path depicts the most likely path through the sequence.

We can see that CRF and LR predictions exhibit similar dynamics, though the CRF estimates are smoother due to the message passing routine used within the internal structure of the model. LR probability estimates appear to not fall below a value of 0.15 whereas CRF models can assign lower probability estimates because sequences of agreeing beliefs will support one another.

6.1.2 Comments on results

We evaluated our classification performance on F-Score and Brier scores. Statistical testing did not yield sufficient evidence for rejecting the null hypothesis for all datasets, *i.e.* CRFs should not be preferred. However, upon deeper investigation, we consistently found for all experiments on all datasets that LR models out-performed CRFs on precision, and CRF models out-performed LR models on recall; in other words, CRFs predict more actual positives, but positive predictions from LR models are more likely to be true positives. CRF models make predictions with influence of the beliefs at neighbouring positions (which themselves have been influenced by their neighbours) so that all nodes in a sequence affect the marginal probabilities calculated at all positions, whereas LR models can only rely on features extracted from local regions of a sequence.

For the task of WH, Trogkanis and Elkan (2010) explicitly stated that false positive predictions are less desirable than false negatives. In prediction, therefore, the authors thresholded probability estimates at a high value (*e.g.* 0.99) and were able to reduce the false positive rate significantly. We have found that LR models can naturally achieve this on the WH dataset without having to threshold the probability estimates at such high values. Therefore, based on the consistency of our observations in all of our experiments, if LR and CRF performance is equivalent, we believe that practitioners informed on the relative costs of false positive and false negative predictions in the application domain may wish to pick the model that best suits these costs; *e.g.* LR if false positives are more costly than false negatives.

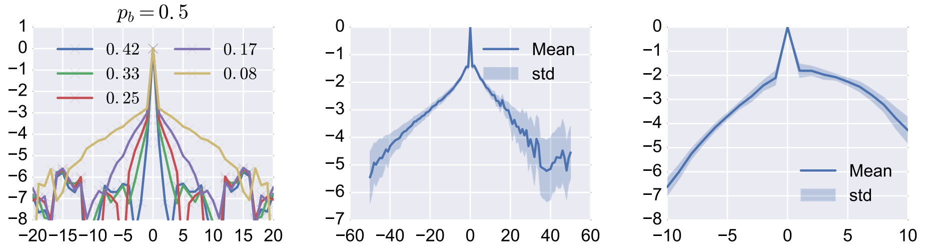
The ODC dataset has been used extensively as an exemplar of ‘smoothing’ probability estimates over sequences. The task we investigate here is to learn the conditional distribution of the fair and biased states with expressive feature templates. Interestingly we demonstrated that LR probability estimates resemble the smoothed probabilities (Fig. 8), which indicates that the smoothing behaviour of structured models can be approximated with a rich set of features. However, in some applications the number of parameters required for unstructured models to approximate the smoothed estimates may be greater than the number of parameters required to train a ‘simpler’ CRF model. We will look at this in more detail in Sect. 6.3.3.

6.2 Comparison to theory

6.2.1 Analysis of auto-correlation

The theory discussed in Sect. 4 related the excess classification risk imposed by ignoring the sequential nature of data to the auto-correlation of the examples. We show the log of auto-correlations of the features as calculated on the three datasets considered in Sect. 5.2.

Figure 9a shows the log of the auto-correlation of the ODC dataset for a sample of 1 000 000 die rolls with various values of p_t (as shown in the legend). This image shows that smaller



(a) **(b)** **(c)**

Fig. 9 Auto-correlation of ODC/AR/WH datasets. The x-axis is the range of lages that were considered, and the y-axis marks the log of the auto-correlation. The rate of decay of the auto-correlation is dataset-dependent, with AR depicting the slowest rate, and WH depicting the fastest. **a** ODC dataset **b** AR dataset **c** WH dataset

p_t yield correlations which persist for longer, as one might expect, although these decay exponentially to a baseline value ($\approx 10^{-6}$).

The auto-correlation of the AR data is shown in Fig. 9b. A wide range of lags was considered here as the average sequence length was long. In this image we can observe a high auto-correlation over the set of lags considered, with a slower rate of decay in comparison to that shown in the ODC. This trend in this figure is visually reminiscent of the trend shown in Fig. 9a for small p_t.

Figure 9c shows the auto-correlations for the WH dataset. Interestingly, this image shows asymmetric auto-correlation is obtained, and that the values obtained at positive lags are greater than those obtained for negative lags.

6.3 Relating theoretical results to practical experiments

6.3.1 Improving WH performance

We noted greater auto-correlation values at positive lags for WH which suggested that more contextual information about hyphenation is available at positive lags than at negative lags. We constructed a new set of feature templates which placed more emphasis on conjunctions of ‘future’ sub-strings ($\mathscr{F}_{\mathrm{H+}}$) to determine whether performance would improve. Using $\mathscr{F}_{\mathrm{H+}}$, we obtained higher F-Scores to 0.965 and 0.971 respectively for CRF and LR models. While this is a modest and statistically insignificant improvement, the use of the $\mathscr{F}_{\mathrm{H+}}$ features yielded improved results on all 10 test folds for LR and CRF models. Furthermore, that these templates *should* improve prediction is not altogether obvious, but the potential for improvement was unveiled by an a priori analysis of the auto-correlation.

$\mathscr{F}_{\mathrm{H+}}$ defines feature templates that place increased emphasis on ‘future’ data. We also performed experiments with $\mathscr{F}_{\mathrm{H-}}$ which increase emphasis on past data. We found that all performance metrics with $\mathscr{F}_{\mathrm{H-}}$ feature templates degraded when compared to $\mathscr{F}_{\mathrm{H}}$ and $\mathscr{F}_{H+}$.

6.3.2 Improving AR performance

By considering the auto-correlation in Fig. 9b, we can see that the auto-correlation remains high over the range of lags shown. We postulate that wide-spanning feature templates may capture context which may improve classification performance. We tested this hypothesis by defining the following skip-gram feature templates

$$\mathscr{F}_{\text{AR}} = f_{\langle 0 \rangle} \cup \{f_{\langle -i,0,i \rangle}\}_{i=1}^{N_{\text{AR}}} \tag{30}$$

where we have set $N_{\text{AR}} = 25$ as, for this range, the auto-correlations remained approximately symmetric in Fig. 9b. Using these feature templates we obtained an improvement of 5 % with CRF and 6 % with LR models, yielding a micro F-Score of $\approx$ 84 % for both models.

While modest improvements were made in predicting ADLs on average, we have made particular improvements on ‘bed to toilet’ activities achieving relative improvements of ≈ 0.25 with LR and CRF models for both residents. It should be noted that we achieved these improvements using new feature templates that were inspired by analysis of auto-correlation trends rather than explicit curation.

6.3.3 Incremental performance assessment

We previously stated that is reasonable to assume that $(n + 1)$-gram features are more expressive than n-gram features, so by taking subsets of $\mathscr{F}_{\text{H}}$ (as described in Sect. 5) we can demonstrate classification performance as the feature representation becomes more and more expressive.

For WH and AR datasets, optimal classification performance is obtained with the full set of feature templates, and so incremental performance only demonstrates that CRF models achieve better performance with more features. With the ODC dataset we notice that CRF models begin to overfit the data quickly, as shown in Fig. 10. We believe the cause for this is due to using complex features to model the simple generative process that underlies the ODC. Interestingly, maximal performance is achieved with the CRF using only three feature templates (*i.e.* $\mathscr{F}_{\text{H}}^3 = \{f_{\langle 0 \rangle}, f_{\langle -1,0 \rangle}, f_{\langle 0,1 \rangle}\}$).

To investigate the effect of encoding long-range dependencies into the sequences, we applied the $\mathscr{F}_{\text{AR}}$ feature templates to the ODC prediction problem. With reference to Fig. 9a we selected $N_{\text{AR}} = 12$ (as this is approximately the point at which the auto-correlations decay to their minimal value). With these feature templates we obtained F-Scores of ≈ 0.83 with both LR and CRF models. Interestingly, the span of these features is 24 instances, which approximately corresponds to the expected run-length of the model since $p_t = 0.05$.

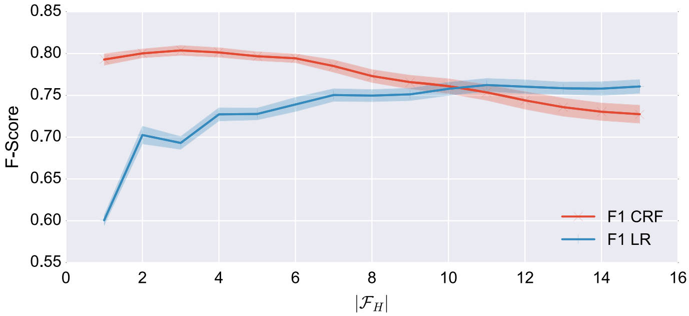


Fig. 10 Incremental F-Scores obtained from LR and CRF models on the ODC dataset. We can see here that the CRF model achieves its best results with a simpler representation (owing to the propagation of beliefs over the sequence), while LR models require a more complicated representation in order to capture the context of the data

7 Conclusions and future work

The ultimate aim of this work is to lay the foundations to determine whether structure needs to be modelled in sequence prediction. Since no unified theoretical and practical assessment of this important question has been considered before, the decision of incorporating structure into a classification problem in much of the applied work that deals with sequences can be considered arbitrary. This paper makes the first steps towards rationalising this decision in general settings.

We demonstrated that structured and unstructured classification models can both achieve equivalent performance on sequential prediction problems. It is remarkable that all sequential datasets investigated in this work could be equivalently modelled by simpler, unstructured models that ignore the sequential nature of the data and instead use features to capture the temporal dependencies. However, we provide an explanation for this in our theoretical analysis of these problems and show that classification risk is intimately linked to the rate of decay of auto-correlations, and the features used in unstructured models cases capture the context with features.

For applications where statistical significance favours neither CRF nor LR models, we would submit to Occam's razor and recommend the selection of the simpler model (*i.e.* the model with fewer parameters) as these should reduce the risk of overfitting and because they offer (potentially significant) reduction in training time. Indeed, from a computational perspective, LR requires optimisation over $|\mathscr{Y}|^2$ fewer parameters than linear-chain CRFs, and therefore may be a favourable model choice based on savings in time and space complexity. This point is of particular interest for streaming applications using CRFs as exact marginal distributions are only available once the full sequences have been obtained (see Sinn and Poupart (2011b) for further discussion). Conversely, exact marginal distributions may be calculated in real-time with LR models.

We used visual analytics tools by leveraging the results of our theoretical analyses. These tools operate on the auto-correlations of dataset sequences a priori to learning classification models, and naturally guided us to specify feature templates that, when incorporated into the classification model, improved classification performance over all datasets.

We speculate that, in general, sequential datasets may have a 'fundamental bandwidth' property, that is related to the jurisdiction over which a particular instance has marked influence. We are encouraged by the variety of auto-correlation profiles that we obtained in our experimental section as these lead us to define different feature templates that improved classification performance. Defining a means of automatically computing this would yield many advantages in sequential modelling, and this work lays the theoretical and practical foundations for the automated discovery of this property.

Future work will seek to extend this work in the following manners. First, we will attempt to automate the specification of (potentially localised) structure based on the auto-correlation profiles that were described in this paper. Furthermore, we will seek to generalise the theoretical and practical analyses outlined in this paper over, for example, nonlinear auto-correlation measures, and to arbitrary graphical structures.

Acknowledgments This work was performed under the Sensor Platform for HEalthcare in Residential Environment (SPHERE) Interdisciplinary Research Collaboration (IRC) funded by the UK Engineering and Physical Sciences Research Council (EPSRC), Grant EP/K031910/1. We gratefully acknowledge the suggestions from the anonymous reviewers which helped us to improve the paper.

References

Andriluka, M., Roth, S., & Schiele, B. (2008). People-tracking-by-detection and people-detection-by-tracking. *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, (pp. 1–8). IEEE.

Appice, A., Guccione, P., & Malerba, D. (2016). Transductive hyperspectral image classification: Toward integrating spectral and relational features via an iterative ensemble system. *Machine Learning*, *103*(3), 343–375.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review, 78*(1), 1–3.

Collins, M. (2002). Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, (Vol. 10, pp. 1–8). Association for Computational Linguistics.

Cook, D. J., & Schmitter-Edgecombe, M. (2009). Assessing the quality of activities in a smart environment. *Methods of Information in Medicine, 48*(5), 480–485.

Cristianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge: Cambridge University Press.

Ferguson, T., Genest, C., & Hallin, M. (2011). *Kendall's tau for autocorrelation*. Los Angeles: Department of Statistics, UCLA.

Globerson, A., Roughgarden, T., Sontag, D., & Yildirim, C. (2015). How hard is inference for structured prediction? *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)* (pp. 2181–2190).

Hang, H., & Steinwart, I. (2015). A bernstein-type inequality for some mixing processes and dynamical systems with an application to learning. *Tech. Rep. 2015-006, Fakultät für Mathematik und Physik, Universität Stuttgart*.

Hoefel, G., & Elkan, C. (2008). Learning a two-stage SVM/CRF sequence classifier. *Proceedings of the 17th ACM conference on Information and knowledge management* (pp. 271–278), ACM.

Jensen, D., & Neville, J. (2002). Linkage and autocorrelation cause feature selection bias in relational learning. *Proceedings of the Nineteenth International Conference on Machine Learning* (pp. 259–266). Morgan Kaufmann Publishers Inc.

Kaalia, R., Srinivasan, A., Kumar, A., & Ghosh, I. (2016). Ilp-assisted de novo drug design. *Machine Learning*, *103*(3), 309–341.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, *82*(1), 35–45.

Kramer, S., Lavrač, N., & Flach, P. (2001). Propositionalization approaches to relational data mining. In S. Džeroski & N. Lavrač (Eds.), *Relational data mining* (pp. 262–291). Berlin, Heidelberg: Springer. doi:10.1007/978-3-662-04599-2_11.

Krogel, M.-A., Rawles, S., Zelezny, F., Flach, P., Lavrac, N., & Wrobel, S. (2003). Comparative evaluation of approaches to propositionalization. *Proceedings of the 13th International Conference on Inductive Logic Programming* (pp. 197–214) . Springer.

Lafferty, J. D., McCallum, A., & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML*, *1*, 282–289.

Li, B. Z., & Wang, G. (2009). Learning rates of least-square regularized regression with polynomial kernels. *Science in China Series A: Mathematics*, *52*(4), 687–700.

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. Cambridge: MIT press.

Naro, D., Rummel, C., Schindler, K., & Andrzejak, R. G. (2014). Detecting determinism with improved sensitivity in time series: Rank-based nonlinear predictability score. *Physical Review E*, *90*(3), 032913.

Pearl, J. (1982). Reverend bayes on inference engines: A distributed hierarchical approach. In *AAAI*, 133–136.

Provost, F. J., Fawcett, T., & Kohavi, R. (1998). The case against accuracy estimation for comparing induction algorithms. *ICML*, *98*, 445–453.

Recht, B., & Fazel, M. (2010). Pablo a parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, *52*(3), 471–501.

Rosasco, L., Vito, E., Caponnetto, A., Piana, M., & Verri, Alessandro. (2004). Are loss functions all the same? *Neural Computation*, *16*(5), 1063–1076.

Schulte, O., Qian, Z., Kirkpatrick, A. E., Yin, X., & Sun, Yan. (2016). Fast learning of relational dependency networks. *Machine Learning*, *103*(3), 377–406.

Sinn, M., & Poupart, P. (2011a). Asymptotic theory for linear-chain conditional random fields. *International Conference on Artificial Intelligence and Statistics*, *15*, 679–687.

Sinn, M., & Poupart, P. (2011b). Error bounds for online predictions of linear-chain conditional random fields: Application to activity recognition for users of rolling walkers. *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on* (vol. 2, pp. 1–6), IEEE.

Steinwart, I., & Anghel, M. (2009). Consistency of support vector machines for forecasting the evolution of an unknown ergodic dynamical system from observations with unknown noise. *The Annals of Statistics*, *37*, 841–875.

Sutton, C., & McCallum, A. (2011). An introduction to conditional random fields. *Machine Learning*, *4*(4), 267–373.

Temko, A., Thomas, E., Marnane, W., Lightbody, G., & Boylan, G. (2011). EEG-based neonatal seizure detection with support vector machines. *Clinical Neurophysiology*, *122*(3), 464–473.

Trogkanis, N., & Elkan, C. (2010). Conditional random fields for word hyphenation. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 366–374). Association for Computational Linguistics.

Twomey, N., Faul, S., & Marnane, W. P. (2010). Comparison of Accelerometer-Based Energy Expenditure Estimation Algorithms. *4th International Conference Pervasive Computing Technologies for Healthcare (PervasiveHealth)* (1–8), IEEE.

Twomey, N., Temko, A., Hourihane, J. O., & Marnane, W. P. (2014). Automated detection of perturbed cardiac physiology during oral food allergen challenge in children. *IEEE Journal of Biomedical and Health Informatics*, *18*(3), 1051–1057.

Viterbi, Andrew J. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on*, *13*(2), 260–269.

Walters, P. (2000). *An introduction to ergodic theory* (Vol. 79). Berlin: Springer Science & Business Media.

Zhang, Z. (2012). Microsoft kinect sensor and its effect. *MultiMedia, IEEE*, *19*(2), 4–10.