The Signature-Based Learning and its Application

Hao Ni[1,5]
joint work with Terry Lyons[2,5], Weixin Yang[3] Lianwen Jin[3], Zecheng Xie[3], Cordelia Schmid[4] and Jiawei Chang[2]

[1]University College London, [2]University of Oxford
[3]South China University of Technology [4]THOTH team, INRIA Grenoble, France
[5]Alan Turing Institute

LMS-EPSRC Durham Symposium Stochastic Analysis,
July 19, 2017

Outline

1. Introduction and Motivation
2. The Signature-Based Framework
   - Regression on the finite dimensional case
   - Regression on the Path Space
   - The Signature/Log-Signature Feature Sets
3. RNN and the log-signature over sub-time intervals
4. Applications

Supervised Learning on the Paths Space

Input: $X \in \mathcal{V}_p([0, T], E) \sim$ Data Stream (**path**).
Output: $Y \in W \sim$ Effects of Data stream.
Interaction: $Y = f(X) + \varepsilon$.
Goal: Estimate $\mathbb{E}[Y^*|X^*] = f(X^*)$ or f from samples of (X, Y).

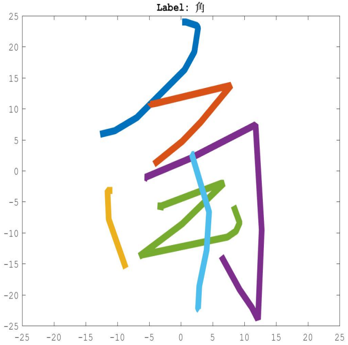
Machine Learning (v.s. Statistics)

1. Rich Dataset;
2. High dimensionality of the input;
3. Relatively small noises, but very complicated f.

Question

How to design a robust and effective algorithm for estimating f?

Online Chinese handwritten Character Recognition



CASIA-OLHWDB1 Dataset:

1. 4,037 categories (3,866 Chinese characters and 171 symbols)
2. 420 writers and 1,694,741 samples.

video

Outline

1. Introduction and Motivation
2. The Signature-Based Framework
   - Regression on the finite dimensional case
   - Regression on the Path Space
   - The Signature/Log-Signature Feature Sets
3. RNN and the log-signature over sub-time intervals
4. Applications

Regression on the Finite Dimensional Space

Dataset: $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ such that $y_i = f(x_i) + \varepsilon_i$, where ε_i is iid with zero mean and $\mathbb{E}[\varepsilon_i | x_i] = 0$.
Question: How to estimate f?

General Framework

$$\begin{aligned}
\text{Model :} \quad & y = f_\theta(x) + \varepsilon \\
\text{Loss function :} \quad & L(\theta|\mathcal{D}) \to \text{Minimize(e.g. } L := \sum_{i=1}^N (y_i - f_\theta(x_i))^2) \\
\text{Optimization :} \quad & \theta^* = \min_\theta(L(\theta|\mathcal{D})) \\
\text{Prediction :} \quad & y_* = f_{\theta^*}(x_*).
\end{aligned}$$

Linear Regression

Model : $f_\theta(x) = \theta_0 x + \theta_1.$

Loss function : $L(\theta|\mathcal{D}) = \sum_{i=1}^{N}(y_i - f_\theta(x_i))^2.$

Solution : $\hat{\theta}_0 = \frac{\sum_{i=1}^{N} x_i y_i}{\sum_{i=1}^{N} x_i^2}, \hat{\theta}_1 = \bar{y} - \hat{a}\bar{x}.$

Nonlinear Regression

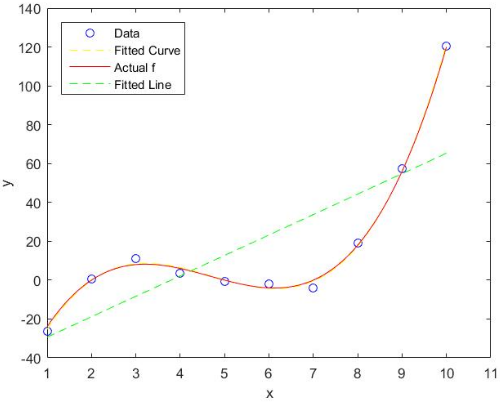


Figure: Polynomial Regression

Basis Expansion

$$y = f(x) + \varepsilon;$$

$$f(x) \approx L(\phi_1(x), \ldots, \phi_n(x)) = \sum_{i=1}^{n} \theta_i \phi_i(x), x \in \mathbb{R}^d.$$

1. Polynomial basis: $x^0, x^1, x^2, \ldots, x^n$;
2. Spline basis...

Remark

There are two crucial ideas about function approximation for $f_\theta(x)$ behind the basis expansion:

1. Features of x denoted by $\mathcal{F}(x)$:

$$f_\theta(x) \approx g_\beta(\mathcal{F}(x)),$$

   where g_β has much simpler form than f_θ.
2. f_θ: non-linear functions, e.g. neural network.

Neural Network

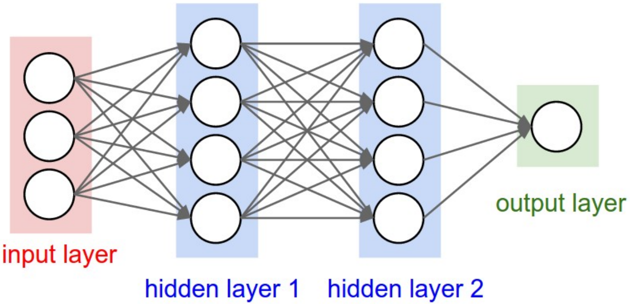


Figure: A regular 3-layer Fully Connected Neural Network.[1]

[1] This figure is retrieved from
http://cs231n.github.io/convolutional-networks/

Overfitting Issue

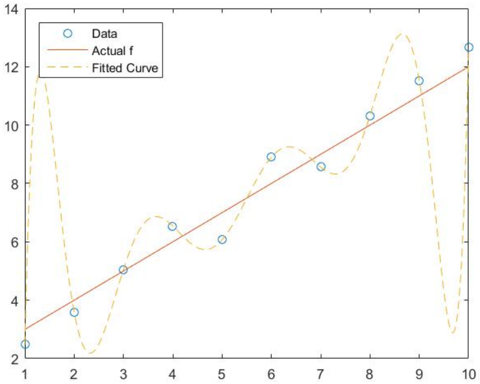


Figure: Overfitting issue

Outline

Regression on the Path Space

Curve Fitting on the Path Space

How to infer the functional f from the samples of the pair $\{(x, y)|x \in \mathcal{V}_p([0, T], \mathbb{R}^d), y \in \mathbb{R}\}$, where $y = f(x) + \varepsilon$ and $p \geq 1$?

Attempt 1

$$x \in \mathcal{V}_p([0, T], \mathbb{R}^d) \quad \leftarrow \quad x_{\mathcal{D}} = (x_{t_1}, \cdots, x_{t_N})$$

where $\mathcal{D} = \left\{(t_i)_{i=1}^N | 0 = t_1 \leq \cdots \leq t_N = T\right\}$.
$x_{\mathcal{D}} \in \mathbb{R}^{dN}$ (increment features) $\rightarrow$ features of $x_{\mathcal{D}}$ $\rightarrow$ curse of dimensionality.

Proposed Solution

Use the step-n signature of a path as feature sets of a path.

$$x \in \mathcal{V}_p([0, T], \mathbb{R}^d) \leftarrow S_n(x) \; (\leftarrow S_n(x_{\mathcal{D}}))$$

where $S_n(x_{\mathcal{D}}) \in T_n(E)$ of dimensionality $\frac{d^{n+1}-1}{d-1}$.

Outline

The Signature of a Path as a Feature set of a Path

Definition (The Signature of a Path)

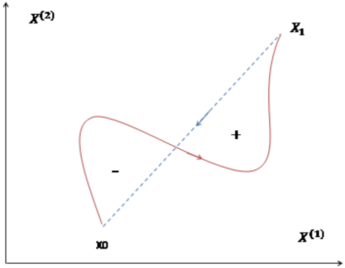
Let J denote a compact interval and $X : J \to E$ be a continuous path with finite p-variation such that the following integration makes sense. The signature of X is defined as follows:

$$S(X)_J = (1, \mathbf{X}_J^1, \ldots, \mathbf{X}_J^n, \ldots),$$

where $\mathbf{X}_J^n = \underset{\substack{u_1 < \cdots < u_n \\ u_1, \ldots, u_n \in J}}{\int \ldots \int} dX_{u_1} \otimes \cdots \otimes dX_{u_n}$ for all $n \geq 1$.

Signature - A top-down description on the path

- Level 1 - increment of a path; Level 2 - area of a path;
- Higher degree- a local structure of a path.
- Uniqueness of the signature ([Hambly and Lyons(2010)], [Boedihardjo et al.(2014)Boedihardjo, Geng, Lyons, and Yang]).



Linear Differential Controlled Equation

Let $X \in \mathcal{V}^1([0,T],\mathbb{R}^d)$ and $Y : [0,T] \to \mathbb{R}$ satisfy

$$dY_t = AY_t dX_t,\ Y_0 = y_0,$$

where $A : \mathbb{R} \to L(\mathbb{R}^d, \mathbb{R})$is a bounded linear map.

Picard's iteration

$$Y_T = y_0 + \sum_{n=1}^{\infty} A^{\otimes n} y_0 \int_0^T \int_0^{u_n} \dots \int_0^{u_2} dX_{u_1} \otimes \dots \otimes dX_{u_n}.$$

$$(1d) = y_0 + \sum_{n=1}^{\infty} A^n y_0 \frac{(X_T - X_0)^n}{n!} = y_0 \exp(A(X_T - X_0)).$$

Remark

1. The signature of a path can be thought as non-commutative monomials of a path.
2. The linear forms on signatures form an algebra - thus they are rich enough to span the space of smooth functionals on paths.
3. Uniform estimates for signatures - The linear form on the truncated signature can well approximate the original function.

Main Idea

$$
\begin{aligned}
& f(X_{[0,T]}) \\
\approx\ & \hat{f}(S(X_{[0,T]})), \text{ by uniqueness of signatures} \\
\approx\ & L(S(X_{[0,T]})), \text{ by shuffle product property of signatures} \\
\approx\ & L(S_n(X_{[0,T]})), \text{ uniform estimates of signatures}
\end{aligned}
$$

Definition (The Log Signature of a Path)

Let $\mathbf{a}$ be an element of $T((E))$. Then for $a_0 > 0$, then $\log(\mathbf{a})$ is the element of $T((E))$ defined by

$$\log(\mathbf{a}) = \log(a_0) + \sum_{n \geq 1} \frac{(-1)^n}{n} \left(\mathbf{1} - \frac{\mathbf{a}}{a_0} \right)^n.$$

The log signature of a path is the logarithm of the signature of a path where the logarithm is defined as above.

Remark

The log signature of a path $X_{[0,T]}$ provides the parsimonious description of the signature $S(X_{[0,T]})$.

The Signature-Based Model

Under the probability space $(\Omega, \mathcal{F}, P)$, $\{X_t\}_{t\in[0,T]}$ is a E-valued stochastic process and Y is a W-valued random variable, such that there exists a **linear** function L,

$$Y = L(S(X_{[0,T]})) + \varepsilon, \mathbb{E}[\varepsilon | X_{[0,T]}] = 0.$$

The Signature Approach

- Calibration: Apply linear regression on $Y^{(i)}$ against $S_n(X^{(i)}_{[0,T]})$ in the learning set and obtained the estimated linear functional $\hat{L}$.
- Goodness of Fitting: Compute the statistics for the fitting error for both the training set and backtesting set.
- Pros: Dimension reduction, non-parametric.

An Illustrative Example

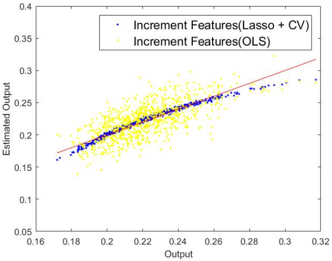
Predicting a solution to an unknown SDE

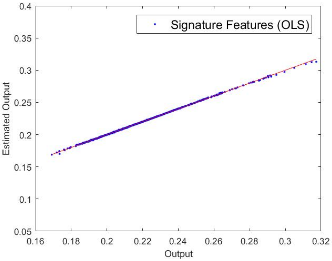
Suppose Y_t satisfies the following SDE:

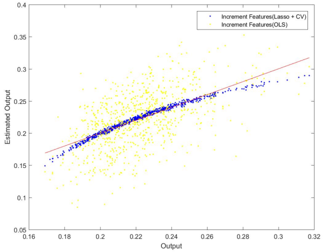
$$dY_t = (1 - Y_t)dX_t^{(1)} + 2Y_t^2 dX_t^{(2)}, Y_0 = 0.$$

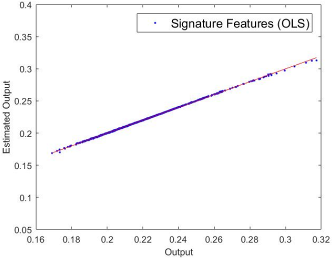
where $X_t = (X_t^{(1)}, X_t^{(2)}) = (t, W_t)$, and the integral is in the Stratonovich sense. [Papavasiliou et al. (2011)]

We generate 1600 independent samples of pairs $(X_{[0,T]}, Y_T)$ for $T = 0.25$ using Milstein's method with number of discretization steps 750. Half of the samples are used for the training set, and the rest is for the backtesting set.


Increment Features(Lasso + CV)
Increment Features(OLS)
Estimated Output
Output

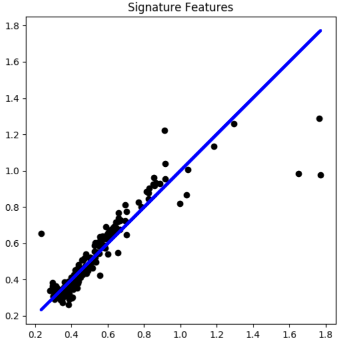

Signature Features (OLS)
Estimated Output
Output

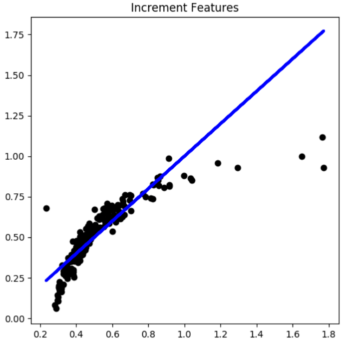

Increment Features(Lasso + CV)
Increment Features(OLS)
Estimated Output
Output


Signature Features (OLS)
Estimated Output
Output

How about $T = 1.0$?

	Signature	Increments
R^2	0.759342070697	0.623664235527





Caution

To achieve certain accuracy of fitting, the truncated signature of high degree might be required, but it might cause the curse of dimensionality.

Possible Solutions

1. Feature sets (Dimension Reduction): the variants of signature features(e.g. log-signatures, signature of paths over sub-time intervals) ;
2. Non-linear function form for f_θ.
3. Regularization.

Predicting a solution to an unknown differential equation

Under the probability space, X_t and Y_t are two stochastic processes. Suppose that Y_t is the solution to the controlled differential equation driven by X_t, i.e.

$$dY_t = f(Y_t)dX_t; Y_0 = y_0$$

where $X : [0, T] \to E$, $f : E \to L(E \to \mathbb{R})$ is a smooth vector field.

Taylor Expansion

$$Y_t - Y_s \approx \sum_{k=1}^{N} f^{\circ k}(Y_s) \int_{s<s_1<\cdots<s_k<t} dX_{t_1} \otimes \cdots \otimes dX_{s_k}$$

where $f^{\circ m} : E \to L(E^{\otimes m}, \mathbb{R})$ is defined recursively by

$$\begin{aligned} f^{\circ 1} &= f; \\ f^{\circ k+1} &= D(f^{\circ k})f. \end{aligned}$$

Theorem ([Boedihardjo et al.(2015)Boedihardjo, Lyons, Yang, et al.])

Let $p \geq 1$. Let $X = (1, X^1, \ldots, X^{\lfloor p \rfloor})$ be a p-weak geometric rough path. Let f be a $Lip(\gamma - 1)$ vector field where $\gamma > p$. Let Y satisfy

$$dY_t = f(Y_t)dX_t.$$

Then there exists a constant C_p depending only on p such that

$$\left| Y_t - Y_s - \sum_{k=1}^{\lfloor \gamma \rfloor} f^{\circ k}(Y_s) X^n_{s,t} \right| \leq \frac{1}{\left(\frac{\lfloor \gamma \rfloor}{p}\right)!} \beta^{\lfloor \gamma \rfloor} M_{p,\gamma} ||f||_{\circ\gamma} ||X||^{\gamma}_{p-var,[s,t]},$$

where $\beta = p\left(1 + \sum_{r=2}^{\infty} \left(\frac{2}{r-1} \wedge 1\right)^{\frac{\lfloor p \rfloor + 1}{p}}\right)$ *and*

$$\begin{aligned} M_{p,\gamma} &= 2C_p \left(|f|_{Lip(\gamma-1)\wedge\lfloor p \rfloor} \vee 1\right)^{\lfloor p \rfloor + 1} \left(||X||_{p-var} \vee 1\right)^{\lfloor p \rfloor + 1} \\ ||f||_{\circ\gamma} &= \max_{\lfloor \gamma \rfloor - \lfloor p \rfloor + 1 \leq m \leq \lfloor p \rfloor} \left| f^{\circ m} \right|^{\min(\gamma - m, 1)}_{Lip(\min(\gamma - m, 1))}. \end{aligned}$$

Numerical Approximation

Let $\mathcal{D} = \{0 = u_0 < u_1 < \cdots < u_N = T\}$. Define $\{\hat{Y}_{u_i}^{\mathcal{D}}\}$ as follows:

$$\begin{aligned} \hat{Y}_{u_0}^{\mathcal{D}} &= y_0, \\ \hat{Y}_{u_{i+1}}^{\mathcal{D}} &= \hat{Y}_{u_i}^{\mathcal{D}} + \sum_{k=1}^{M} f^{\circ k}(\hat{Y}_{u_i}^{\mathcal{D}}) X_{u_i,u_{i+1}}^{k} := g(\pi^{M}(\log S(X_{u_i,u_{i+1}})), \hat{Y}_{u_i}^{\mathcal{D}}), \end{aligned}$$

where $i \in \{1, \cdots, N\}$.

Remark

For any given arbitrage error tolerance ε, when Δu is small enough and d_{ls} is large enough, there exists certain non-linear function g, such that

$$||Y_T - \hat{Y}_{u_N}|| \leq \varepsilon.$$

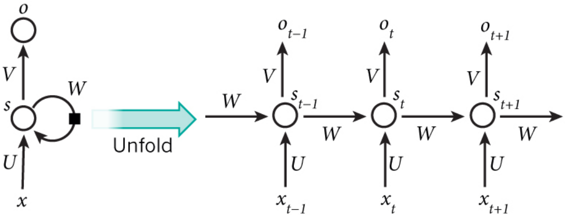


Figure: the Architecture of Recurrent Neural Network (RNN)

Recurrent Neural Network

- x_t is the input at time step t.
- s_t is the hidden state at time step t. It is the “memory” of the network.
- o_t is the output at step t.

$$\begin{aligned} s_t &= f(Ux_t + Ws_{t-1}) \\ o_t &= q(Vs_t) \end{aligned}$$

Taylor Expansion Approximation

$$\hat{Y}^{\mathcal{D}}_{t_{i+1}} = g(\pi^M(\log S(X_{u_i,u_{i+1}})), \hat{Y}^{\mathcal{D}}_{t_i})$$

(Suppose $g(x, y) \approx q(V(Ux + Wy))$).

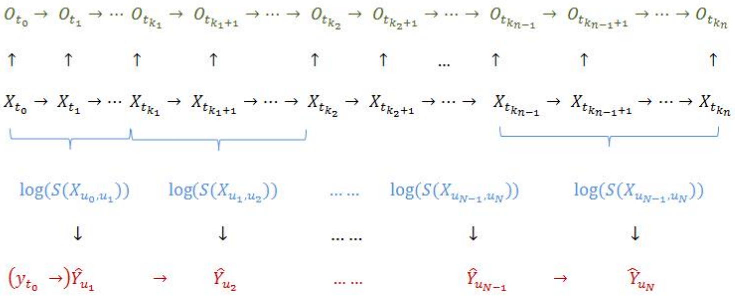


Figure: the Architecture of RNN + Log Signature Framework. Here $\mathcal{D}_0 = \{0 = t_0 < t_1 < \cdots < t_n = T\}$ and $\mathcal{D} \subset \mathcal{D}_0$, i.e. $\mathcal{D} = \{0 = u_0 < u_1 < \cdots < u_N = T | \forall i \in \{1, \cdots, N\}, \exists k_i, u_i = s_{k_i}\}$.

Remark

- When $N = n$ and the degree of the truncated log signature d_{ls} is set to 1, our method is the same as the standard RNN;
- For any given arbitrage error tolerance ε, when Δu is small enough and d_{ls} is large enough, there exists the RNN with the input being $\{\log(S(X_{u_i,u_{i+1}}))\}_{i=0}^{N-1}$ to approximate Y_T up to the error tolerance ε.
- Advantage: more efficient in terms of run time.

Proposed Algorithm

1. For each input path $\{X_{t_i}\}_{i=1}^{n}$, calculate the log signature feature set of the input $\{\log(S(X_{u_i,u_{i+1}}))\}_{i=0}^{N-1}$
2. For the given error tolerance ε and the fixed maximum iteration N_I, calculate the optimal parameters in the RNN model with log signature feature as inputs.
3. Calculate R^2 statistics in the backtesting set as the indicator of the goodness of the fitting.

Revisit the SDE Example

Suppose Y_t satisfies the following SDE:

$$dY_t = (1 - Y_t)dX_t^{(1)} + 2Y_t^2 dX_t^{(2)}, Y_0 = 0.$$

where $X_t = (X_t^{(1)}, X_t^{(2)}) = (t, W_t)$, and the integral is in the Stratonovich sense. [Papavasiliou et al. (2011)]

Based on the Milstein's Method we generate samples of pairs $(X_{[0,T]}, Y_T)$ for $\mathbf{T} = \mathbf{1.0}$. We split the samples into the training dataset and the backtesting dataset.

	Log Sig	Increment
$\varepsilon = 0.01$	99.7469%	99.7509%
$\varepsilon = 0.001$	99.9757%	99.9712%
$\varepsilon = 0.0001$	99.9976%	99.9976%

Table: R^2 comparison of the testing dataset

	Log Sig	Increment
$\varepsilon = 0.01$	1199.78 s	4785.99 s
$\varepsilon = 0.001$	3487.65 s	10107.11 s
$\varepsilon = 0.0001$	39788.25 s	241790.99 s

Table: Run time comparison

Applications

- Online Character/Text Recognition;
  - First to use the signature feature and convolutional neural network for OLCHR [Graham(2013)]. DNN + Signature (or its variants)-> OLCHR [Reizenstein(2014)], [Yang et al.(2016)Yang, Jin, Ni, and Lyons]
  - Convolutional Recurrent Neural Network (CRNN) + Signature + Implicit Language Model-> Online Text Recognition [Xie et al.(2016)Xie, Sun, Jin, Ni, and Lyons]).
- Action Recognition:
  - Signature of Signature of the landmark paths + Drop connected neural network [Weixin Yang(2017)]

Future Work

Apply the RNN + Log Signature approach to the online text recognition and action classification.

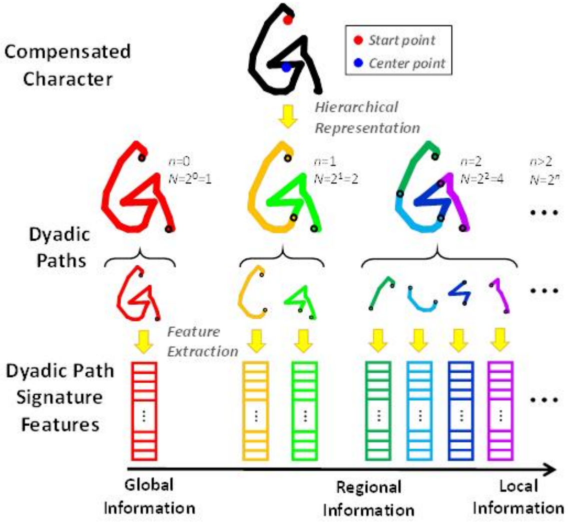


Figure: Illustration of the proposed dyadic path signature features.

References I

H. Boedihardjo, X. Geng, T. Lyons, and D. Yang.
The signature of a rough path: Uniqueness.
arXiv preprint arXiv:1406.7871, 2014.

H. Boedihardjo, T. Lyons, D. Yang, et al.
Uniform factorial decay estimates for controlled differential equations.
Electronic Communications in Probability, 20, 2015.

B. Graham.
Sparse arrays of signatures for online character recognition.
arXiv preprint arXiv:1308.0371, 2013.

References II

B. Hambly and T. Lyons.
Uniqueness for the signature of a path of bounded variation and the reduced path group.
Annals of Mathematics, 171(1):109–167, 2010.

J. Reizenstein.
Signatures in online handwriting recognition.
Preprint, 2014.

H. N. C. S. L. J. J. C. Weixin Yang, Terry Lyons.
Leveraging the path signature for skeleton-based human action recognition.
arXiv preprint arXiv:1308.0371, 2017.

References III

Z. Xie, Z. Sun, L. Jin, H. Ni, and T. Lyons.
Learning spatial-semantic context with fully convolutional recurrent network for online handwritten chinese text recognition.
Submitted to IEEE on Transactions on Pattern Analysis and Machine Intelligence, 2016.

W. Yang, L. Jin, H. Ni, and T. Lyons.
Rotation-free online handwritten character recognition using dyadic path signature features, hanging normalization, and deep neural network.
Accepted by International Conference on Pattern Recognition, 2016.