

A First Look at Chebyshev-Sobolev Series for Digital Ink

Deepak Singh Kalhan¹, Stephen M. Watt¹

¹Cheriton School of Computer Science, University of Waterloo, Waterloo Canada, N2L 3G1

Abstract

Considering digital ink as plane curves provides a valuable framework for various applications, including signature verification, note-taking, and mathematical handwriting recognition. These plane curves can be obtained as parameterized pairs of approximating truncated series $(x(s), y(s))$ determined by sampled points. Earlier work has found that representing these truncated series (polynomials) in a Legendre or Legendre-Sobolev basis has a number of desirable properties. These include compact data representation, meaningful clustering of like symbols in the vector space of polynomial coefficients, linear separability of classes in this space, and highly efficient calculation of variation between curves. In this work, we take a first step at examining the use of Chebyshev-Sobolev series for symbol recognition. The early indication is that this representation may be superior to Legendre-Sobolev representation for some purposes.

Keywords

Digital ink, Functional approximation, Orthogonal polynomials, Chebyshev polynomials, Sobolev polynomials

1. Introduction

Digital ink is an important tool in modern computing with applications ranging from note-taking, signature verification to handwriting and mathematical expression recognition. In order to accurately use digital ink data, methods like orthogonal series representations can be employed. The use of orthogonal polynomials such as Legendre, Laguerre, Chebyshev or other polynomials provides a powerful framework for approximating functions. The Legendre-Sobolev series has gained attention for its improved recognition rates [1]. In this paper, we extend the exploration of orthogonal series representation by focusing on Chebyshev-Sobolev series.

Viewing digital ink as parametric plane curves is a sharp departure from previous work which used either a pixel-based approach or point-sequence approach. Both of these earlier approaches are resolution dependent, one in space and the other in time. This leads to a host of challenges. For example, older data is “resampled” (*i.e.* interpolated, then discretized) to use these methods with data from different sources. Another issue is in matching corresponding points in comparing curves, leading to challenging optimization of alignment-based “dynamic time warping”. Working with parametric curves as algebraic objects avoids all these problems.

Math UI 2024, Montreal Canada, August 9 2024

✉ dsinghkalhan@uwaterloo.ca (D. S. Kalhan); smwatt@uwaterloo.ca (S. M. Watt)

🌐 www.linkedin.com/in/dsinghkalhan/ (D. S. Kalhan); https://cs.uwaterloo.ca/~smwatt/ (S. M. Watt)

🆔 0009-0007-7585-2901 (D. S. Kalhan); 0000-0001-8303-4983 (S. M. Watt)



© 2024 Copyright for this paper by its authors.

Building on the results using Legendre-Sobolev series, this paper explores the potential of the Chebyshev-Sobolev series for digital ink representation. Chebyshev polynomials are known for stability [2] and improved accuracy for functional approximation [3]. The Sobolev space improves shape matching by including the derivatives in the expression. Therefore, we suspect Chebyshev-Sobolev series may be suitable for handwritten text and symbol recognition.

This paper is organized as follows: Section 2 discusses the background methods to use an orthogonal polynomial basis for digital ink representation. In Section 3, we define a “Chebyshev-Sobolev” inner product and give some of its properties. Section 4 gives an algorithm to classify a sample curve against a database of many reference curves. This uses the same idea as earlier work based on Legendre-Sobolev polynomials. Section 5 describes some initial experiments examining the use of digital ink in a Chebyshev-Sobolev basis representation. Finally, we give some brief conclusions in Section 6.

2. Background

Digital ink is typically generated by sampling points from pen traces which results in a series of points, each giving x and y coordinates. The time interval and resolution depend on the hardware so devices with different hardware will result in different series of points from the same input trace. As technology evolves, various *ad hoc* treatments have been developed, such as size normalization and resampling to deal with these differences. However, these methods introduce their own problems. Representing handwritten symbols in the space of coefficients of an orthogonal basis approximation is a useful alternative method [4, 5, 6, 7] and has proven to be robust against changes in hardware [8]. Additionally, this method has high recognition rates with small training sets [4].

2.1. Parameterized Curves in an Orthogonal Basis

In our work, we have considered an ink trace as a segment of plane curve $x(s)$ and $y(s)$, parameterized by euclidean arc length s , where

$$ds^2 = dx^2 + dy^2. \quad (1)$$

If we let $\{B_i(s)\}_{i=0,\dots,d}$ be a graded polynomial basis for approximating $(x(s), y(s))$, a trace can be represented as

$$x(s) \approx \sum_{i=0}^d x_i B_i(s) \quad (2)$$

$$y(s) \approx \sum_{i=0}^d y_i B_i(s) \quad (3)$$

where, x_i and y_i are the coefficients and d is the degree of the truncated series. To allow efficient calculation of coefficients, the functions $B_i(s)$ can be chosen to be polynomials that are orthogonal with respect to a functional inner product. The approximated curve can be made close to the original trace by choosing higher value of d .

Given a functional inner product, a graded basis of orthogonal polynomials may be obtained using Gram-Schmidt orthogonalization of the set of monomials $\{s^i\}_{i \in [0..d]}$. Then a digital ink trace can be represented in this basis as the vector of coefficients $(x_0, \dots, x_d, y_0, \dots, y_d)$.

We can now illustrate the benefit of representing curves in an orthogonal basis. The usual way of comparing two traces given as series of sampled points is to find a correspondence between points of the traces to minimize the sum of distances between corresponding points. Suppose two traces are given as point sequences $(X_0, Y_0), \dots, (X_m, Y_m)$ and $(U_0, V_0), \dots, (U_n, V_n)$. (Note that here we use subscripts to indicate a position in the sequence, not a coefficient.) Without loss of generality, assume $m \geq n$. Then we measure the squared distance between the two traces as

$$\min_{\phi: [0..m] \rightarrow [0..n]} \sum_{i=0}^m (X_i - U_{\phi(i)})^2 + (Y_i - V_{\phi(i)})^2 \quad (4)$$

over all choices of non-decreasing ϕ with $\phi(0) = 0$ and $\phi(m) = n$. Considerable attention has been given to strategies for this minimization problem.

In contrast, suppose we have represented the traces as parameterized plane curves in the basis $B_i(s)$, then we measure the squared distance as

$$\begin{aligned} & \langle x - u, x - u \rangle_B + \langle y - v, y - v \rangle_B \\ &= \sum_{i=0}^d \sum_{j=0}^d (x_i - u_i)(x_j - u_j) \langle B_i, B_j \rangle_B + \sum_{i=0}^d \sum_{j=0}^d (y_i - v_i)(y_j - v_j) \langle B_i, B_j \rangle_B \\ &= \sum_{i=0}^d ((x_i - u_i)^2 + (y_i - v_i)^2) \langle B_i, B_i \rangle_B. \end{aligned} \quad (5)$$

This is straightforward and fast to compute since the $\langle B_i, B_i \rangle_B$ are a set of $d + 1$ fixed values.

2.2. Legendre Inner Product and Legendre Basis

The classical Legendre polynomials $P_n(x)$ are a degree-graded family orthogonal with respect to the following functional inner product

$$\langle f, g \rangle_P = \int_{-1}^1 f(x)g(x)dx \quad (6)$$

and normalized as

$$\langle P_m, P_n \rangle_P = \begin{cases} \frac{2}{2n+1} & \text{if } m = n \\ 0 & \text{otherwise.} \end{cases} \quad (7)$$

We call (6) the *Legendre inner product* and the corresponding norm is

$$\|f\|_P^2 = \langle f, f \rangle_P. \quad (8)$$

If f and g are polynomials given in Legendre basis such that

$$f(x) = \sum_{i=0}^d f_i P_i(x) \quad g(x) = \sum_{i=0}^d g_i P_i(x), \quad (9)$$

then, using (7), we have

$$\langle f, g \rangle_P = \sum_{i=0}^d \frac{2}{2i+1} f_i g_i. \quad (10)$$

Note that the coefficients f_i may be obtained as

$$f_i = \frac{\langle f, P_i \rangle_P}{\langle P_i, P_i \rangle_P}. \quad (11)$$

2.3. Legendre-Sobolev Inner Product

A Sobolev inner product of order N on a domain D with Borel measures μ_i and coefficients $\lambda_i \in \mathbb{R}^+$ is defined as

$$\langle f, g \rangle_S = \sum_{i=0}^N \lambda_i \int_D f^{(i)} g^{(i)} d\mu_i.$$

Earlier work [7] examined classification using the *Legendre-Sobolev* inner product studied by Althammer [9],

$$\langle f, g \rangle_{LS} = \int_{-1}^1 f(x)g(x)dx + \lambda \int_{-1}^1 f'(x)g'(x)dx. \quad (12)$$

3. Chebyshev-Sobolev Inner Product

We now come to the main topic of the article. We propose to use a Sobolev inner product based on Chebyshev polynomials. Chebyshev polynomials are used extensively in numerical approximation due to their “minimax” property, that is minimizing the maximum error in the approximation of a function over an interval. As changing polynomial basis may be numerically ill conditioned, we explore using a Chebyshev basis *ab initio* and working with this representation throughout.

3.1. Chebyshev Inner Product

We begin by noting the familiar properties of the Chebyshev polynomials of the first kind $T_n(x)$, which are a degree-graded family orthogonal with respect to the inner product

$$\langle f, g \rangle_T = \int_{-1}^1 \frac{f(x)g(x)}{\sqrt{1-x^2}} dx \quad (13)$$

and normalized as

$$\langle T_n, T_m \rangle_T = \begin{cases} 0 & \text{if } n \neq m \\ \pi & \text{if } n = m = 0 \\ \frac{\pi}{2} & \text{otherwise.} \end{cases} \quad (14)$$

We call (13) the *Chebyshev inner product* and the corresponding norm is

$$\|f\|_T^2 = \langle f, f \rangle_T. \quad (15)$$

If $f(s)$ and $g(s)$ are polynomials of degree d written in a Chebyshev basis as

$$f(x) = \sum_{i=0}^d f_i T_i(x) \quad g(x) = \sum_{i=0}^d g_i T_i(x), \quad (16)$$

then, using (14), we have

$$\begin{aligned} \langle f, g \rangle_T &= \int_{-1}^1 \left(\sum_{i=0}^d \sum_{j=0}^d f_i T_i(x) g_j T_j(x) \right) \frac{dx}{\sqrt{1-x^2}} \\ &= \int_{-1}^1 \left(\sum_{i=0}^d f_i g_i T_i(x) T_i(x) + \text{cross terms} \right) \frac{dx}{\sqrt{1-x^2}} \\ &= \frac{\pi}{2} \left(2f_0 g_0 + \sum_{i=1}^d f_i g_i \right). \end{aligned} \quad (17)$$

The coefficients f_i may be obtained as

$$f_i = \frac{\langle f, T_i \rangle_T}{\langle T_i, T_i \rangle_T}. \quad (18)$$

3.2. Chebyshev-Sobolev Inner Product

We now consider the first order Sobolev inner product for Chebyshev polynomials,

$$\langle f, g \rangle_{CS} = \int_{-1}^1 \frac{f(x)g(x)}{\sqrt{1-x^2}} ds + \lambda \int_{-1}^1 \frac{f'(x)g'(x)}{\sqrt{1-x^2}} ds \quad (19)$$

where $\lambda > 0$. We call this a *Chebyshev-Sobolev inner product* with corresponding norm

$$\|f\|_{CS}^2 = \langle f, f \rangle_{CS}. \quad (20)$$

Applying Gram-Schmidt orthogonalization with respect to $\langle \cdot, \cdot \rangle_{CS}$ to monomials $\{1, x, x^2, \dots\}$ gives a degree-graded family, we call *Chebyshev-Sobolev polynomials*. Lets denote by $S_{\lambda i}$. We take the same normalization as for Chebyshev polynomials so $S_{0 i} = T_i$.

An analytic function $f(x)$ may be written as a series using the basis $S_{\lambda i}$

$$f(x) = \sum_{i=0}^{\infty} f_i S_{\lambda i}(s),$$

where the coefficients f_i satisfy

$$f_i = \frac{\langle f, S_{\lambda i} \rangle_{CS}}{\langle S_{\lambda i}, S_{\lambda i} \rangle_{CS}}.$$

Approximations to $f(x)$ may be obtained by summing only up to some degree d .

4. A Matching Algorithm

Comparing a sample input handwritten character to a set of model characters to find the best match can now be done as follows:

1. Receive as inputs points $(x_i, y_i), i \in [0..N]$.
2. Compute piece-wise polynomial splines (e.g. piece-wise-linear or piece-wise-cubic) interpolating the sequences x_i and y_i .
3. Compute arc lengths $s_i, i \in [0..N]$ from the point (x_0, y_0) to (x_i, y_i) along the interpolating splines.
4. Shift and scale so that $s_0 = -1$ and $s_N = 1$. This scales the character to a standard size.
5. Let $C_X(s)$ and $C_Y(s)$ be the piece-wise splines such that $C_X(s_i) = x_i$ and $C_Y(s_i) = y_i$.
6. Compute the coefficients X_i and $Y_i, i \in [1..d]$ such that $C_X(s) = \sum_{i=0}^d X_i T_i(s)$ and $C_Y(s) = \sum_{i=0}^d Y_i T_i(s)$. This requires $2N$ low-degree polynomial integrations done by formula. The coefficients of T_0 are to be dropped to center the character.
7. At this point, we have a resolution-independent representation of the character that is centered and scaled to a standard size. This representation requires $2d$ numbers.
8. Compare this sample character against the model characters one by one. For each model character G compute the euclidean norm as follows:

$$D_G^2 = \|C_X - G_X\|^2 + \|C_Y - G_Y\|^2$$

and return the character model that gives the minimum.

We remark on a few things:

- All the approximation comes from choosing the truncation degree d of the series.
- The choice of order of the interpolating spline should ideally be the same for the sample character as for all the models, but in practice it might not matter very much. The step that scales s is important.
- The integrations to obtain the coefficients in C_X and C_Y can be done by polynomial formulas and do not require numerical integration techniques.
- If the summations to compute the Chebyshev basis coefficients of the derivative polynomials becomes computationally significant, then we can consider using Clenshaw summation [10].
- This whole procedure could use shifted Chebyshev polynomials $T_n^*(x) = T_n(2x - 1)$ on $[0, 1]$, in which case s would be scaled to $[0, 1]$.

The main point with this, and with the Legendre-Sobolev method, is that it is possible to compare efficiently against a great many models, since the computation for each depends only on the degree d of the approximation and in particular does not depend on the number of sample points N .

5. Experiments on Digital Ink

To give a first assessment of the suitability of the Chebyshev-Sobolev polynomials, we have run tests on Ink ML dataset [11] and UCI pendigits dataset [12]. The ink dataset consist of traces of different letters and UCI pendigits dataset contain samples of handwritten digits (0-9) of multiple users. We started the experiment by approximating the digital ink using Chebyshev-Sobolev polynomials. For the sake of comparison, we have also considered Legendre, Chebyshev, and Legendre-Sobolev polynomial. We perform the following experiments:

5.1. Handwriting representation using Chebyshev-Sobolev series

In this experiment, we plot the approximated curve considering different degrees for the orthogonal polynomials. We have considered the hyper parameter λ as $\frac{1}{8}$ and compare the approximated curve with the original trace by increasing degree. It can be observed that the quality of approximated curve improves with increase in degree. Figure 1a shows the letters in original trace of Ink ML dataset. In this figure, we can observe large number of points resulting in shorter distance between the adjacent curves. From figure 1b,1c, and 1d it can be seen that the approximated curve are able to accurately and smoothly represent the letters. Additionally, with Chebyshev-Sobolev polynomials, we are able to accurately represent the self overlapping letters like "e", and "o".

Figure 2a shows the 10 digits selected at random from the UCI pen digits dataset. In this figure we can observe that each digit has exactly 8 points and we connected these points considering them as linearly separable. Due to limited number of points and equal distances in the points, the handwritten digits are coarsely represented. From figure 2b, 2c and 2d, it can be seen that the approximated curve are able to accurately represent the handwritten digits. The gaps between the point is smooth and there are no vibrations near the end points.

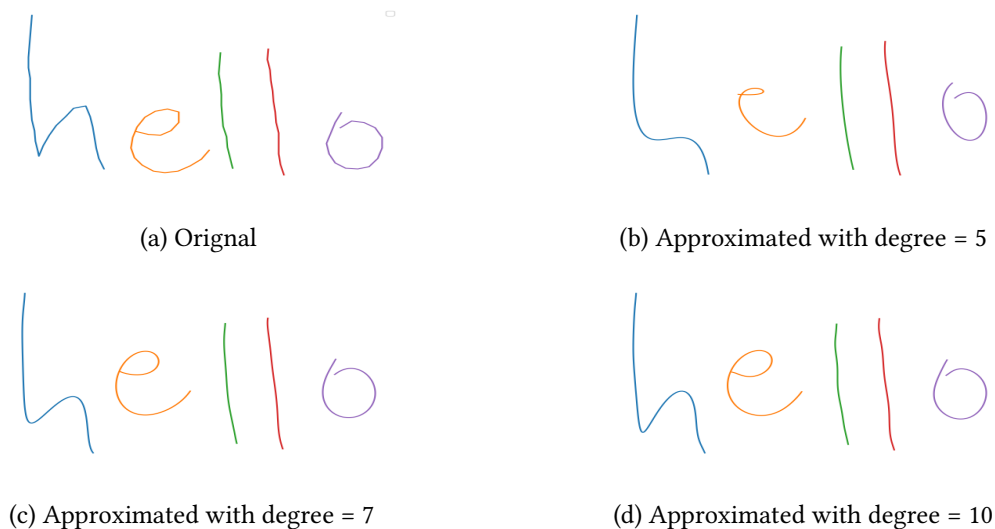


Figure 1: Handwriting representation test on Ink ML dataset

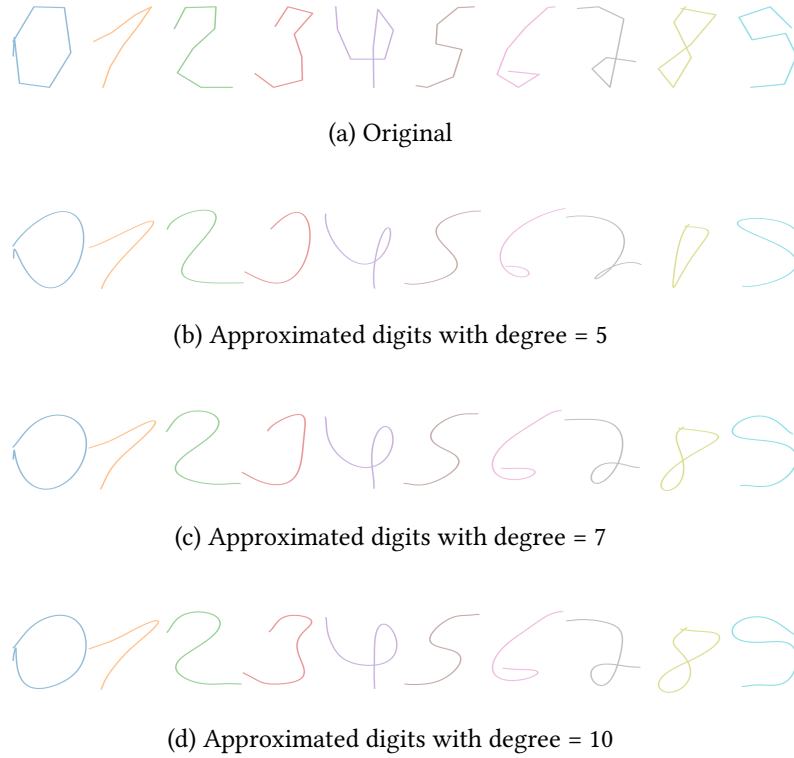


Figure 2: Handwriting representation test on UCI pen digits dataset

5.2. Representation error of approximated curve using Chebyshev-Sobolev series

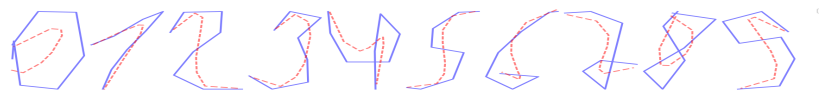
In this test, we measure the representation error by calculating euclidean distance from the original points and approximated points. The error can be represented as:

$$Error := \sum_{i=0}^N \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2} \quad (21)$$

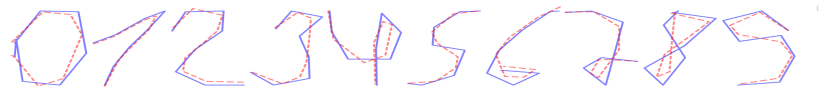
where, N is the total number of points in the original trace, (x_i, y_i) are the original points, and (\hat{x}_i, \hat{y}_i) are the approximated points in the trace. Figure 3a, 3b, 3c, and 3d represents the points generated by considering Chebyshev-Sobolev polynomials upto degree 3, 7, 10, and 15 respectively. For the sake of comparison we have joined the points with the straight line to see likelihood with the original trace. The approximated points for some traces like of "1" are very close with degree 7, while some self looping traces like "9" need higher degree polynomials for approximation. These findings are also justified in Figure 5b which is plot of representation error in equation (21) vs degree of polyomial. Similarly, the Figure 4a, 4b, 4c, and 4d shows approximated points of Ink ML dataset considering polynomials upto degree 3, 7, 10, and 20. Additionally, figure 5a and 5b shows the plot of representation error vs degree.

5.3. Handwriting recognition using k nearest neighbours

In order to evaluate the effectiveness of handwriting recognition using Chebyshev-Sobolev series we perform a test using UCI pendigits dataset from [12]. The dataset consist of 10992



(a) Approximated digits with degree = 3



(b) Approximated digits with degree = 7



(c) Approximated digits with degree = 10



(d) Approximated digits with degree = 15

Figure 3: Representation error on UCI pen digits dataset



(a) Approximated with degree = 3



(b) Approximated with degree = 7

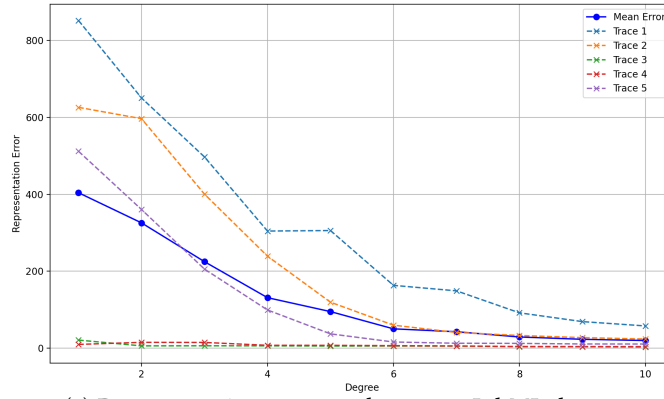


(c) Approximated with degree = 10

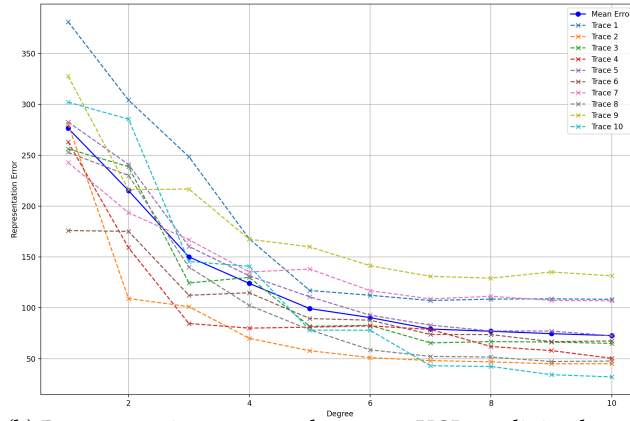


(d) Approximated with degree = 15

Figure 4: Representation error on Ink ML dataset



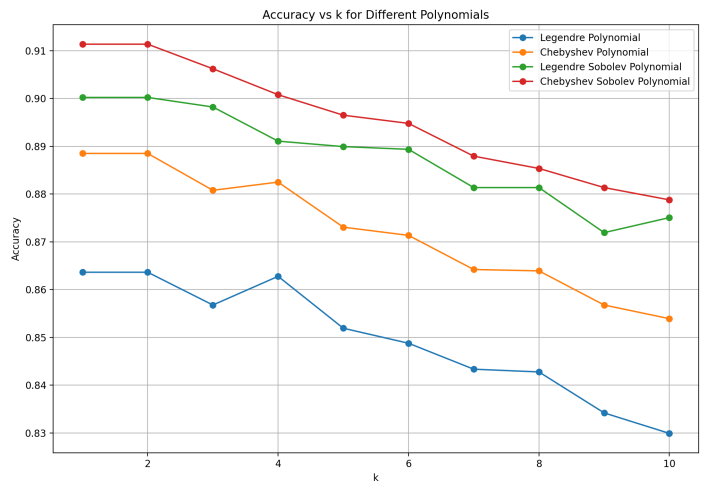
(a) Representation error vs degree on InkML dataset



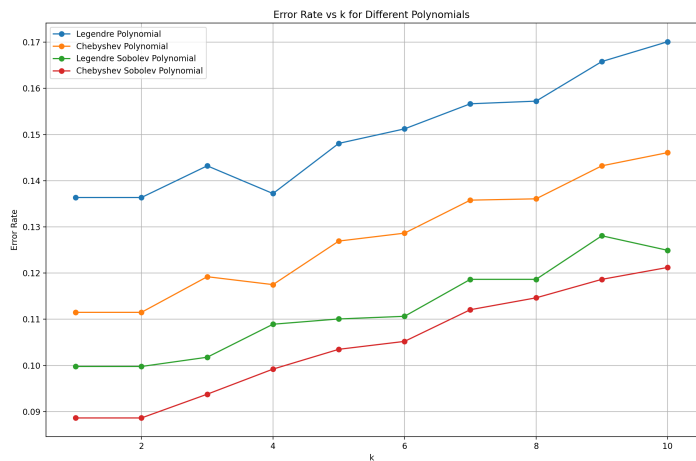
(b) Representation error vs degree on UCI pendigits dataset

Figure 5: Representation error using Chebyshev-Sobolev series

samples of 10 different classes of numerical digits (0-9). Each digit consist of equal 8 points and each class has approximately 1000 samples. In order to examine the classification performance of Chebyshev-Sobolev series, we used two third of the handwritten digits in the data set for training, the remaining ones for testing, and then computed the coefficients of these handwritten digits respectively. We have truncated the series upto $d=10$ and used $\lambda = \frac{1}{8}$. Afterwards, Euclidean distance was used as the metric to perform k -nearest neighbours classification of the test set from the training set, where k ranges from 1 to 10. Fig. 6a and Fig. 6b shows the accuracy and error rate for k ranging from 1 to 10 for Legendre , Chebyshev, Legendre- Sobolev, and Chebyshev-Sobolev Polynomial. The accuracy remains high, but decreases with increase in k for all the methods. The error rates increases with increase in k . This is an expected behaviour of k -NN method where the smaller k captures the best trends of the data. It can be Observed that Chebyshev-Sobolev polynomials performs the best for all the values of k . Also, Legendre-Sobolev has the accuracy higher than Chebhyshev Polynomial and Legendre Polynomials. The decrease in accuracy with k is comparatively gradual for Chebyshev-Sobolev polynomial.



(a) Accuracy vs k for Different Polynomials



(b) Error Rate vs k for Different Polynomials

Figure 6: Handwriting recognition using k nearest neighbours

6. Conclusions

We have defined the family of Chebyshev-Sobolev polynomials in analogy to the Legendre-Sobolev polynomials studied by Althammer and used in earlier work in mathematical handwriting recognition. Some initial results show that they can be superior to Legendre-Sobolev polynomials in some circumstances. This motivates us to study their properties in more detail and their effectiveness on more varied symbol corpora.

References

- [1] V. Mazalov, S. M. Watt, Improving isolated and in-context classification of handwritten characters, in: Document Recognition and Retrieval XIX, volume 8297, SPIE, 2012, pp. 77–84.
- [2] G. Wu, Y. Zhang, A new chebyshev polynomials descriptor applicable to open curves, Pattern Recognition Letters 62 (2015) 41–48.
- [3] C. Lanczos, Solution of systems of linear equations by minimized iterations, J. Res. Nat. Bur. Standards 49 (1952) 33–53.
- [4] S. M. Watt, Polynomial approximation in handwriting recognition, in: Proceedings of the 2011 International Workshop on Symbolic-Numeric Computation, 2012, pp. 3–7.
- [5] O. Golubitsky, S. M. Watt, Online stroke modeling for handwriting recognition, in: Proceedings of the 2008 conference of the center for advanced studies on collaborative research: meeting of minds, 2008, pp. 72–80.
- [6] B. W. Char, S. M. Watt, Representing and characterizing handwritten mathematical symbols through succinct functional approximation, in: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), volume 2, IEEE, 2007, pp. 1198–1202.
- [7] O. Golubitsky, S. M. Watt, Distance-based classification of handwritten symbols, International Journal on Document Analysis and Recognition (IJ DAR) 13 (2010) 133–146.
- [8] R. Hu, S. M. Watt, Identifying features via homotopy on handwritten mathematical symbols, in: 2013 15th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, IEEE, 2013, pp. 61–67.
- [9] P. Althammer, Eine Erweiterung des Orthogonalitätsbegriffes bei Polynomen und deren Anwendung auf die beste approximation., J. Reine Ang. Math. 211 (1962) 192–204.
- [10] C. Clenshaw, A note on the summation of Chebyshev series, Math Comp 9 (1955) 118–120.
- [11] Y.-M. Chee, K. Franke, M. Froumentin, S. Madhvanath, J.-A. Magana, G. Russell, G. Seni, C. Tremblay, S. Watt, L. Yaeger, Ink Markup Language (InkML), Technical Report, W3C, 2006.
- [12] E. Alpaydin, F. Alimoglu, Pen-Based Recognition of Handwritten Digits, UCI Machine Learning Repository, 1998. DOI: <https://doi.org/10.24432/C5MG6K>.