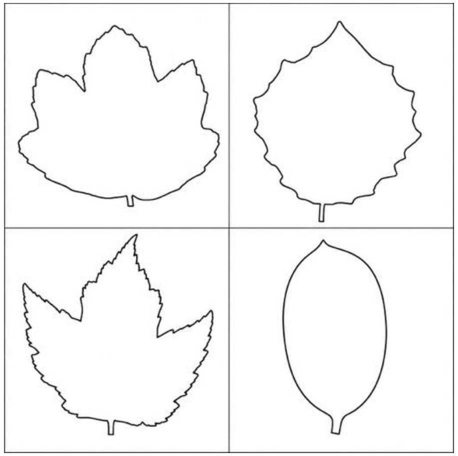
Geometric Methods for Curve Classification: *Mathematical Handwriting Recognition*

Stephen M. Watt

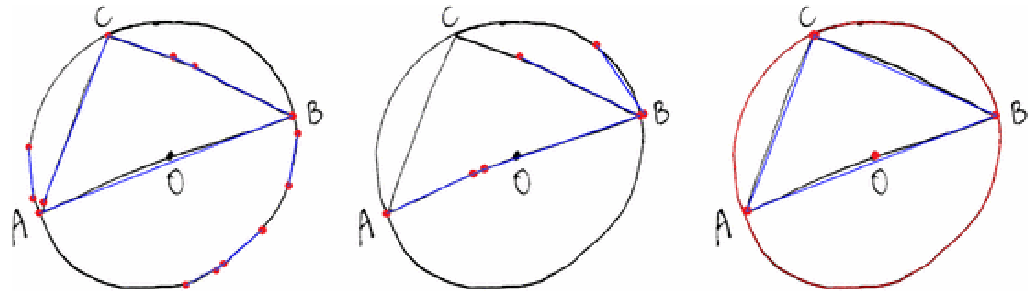
David R. Cheriton School of Computer Science
University of Waterloo

Go20 Conference on Scientific Computing and Software
Marsalforn, Gozo, May 17-22, 2026

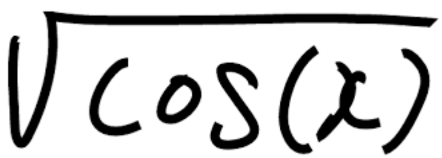
Curve Classification is Everywhere



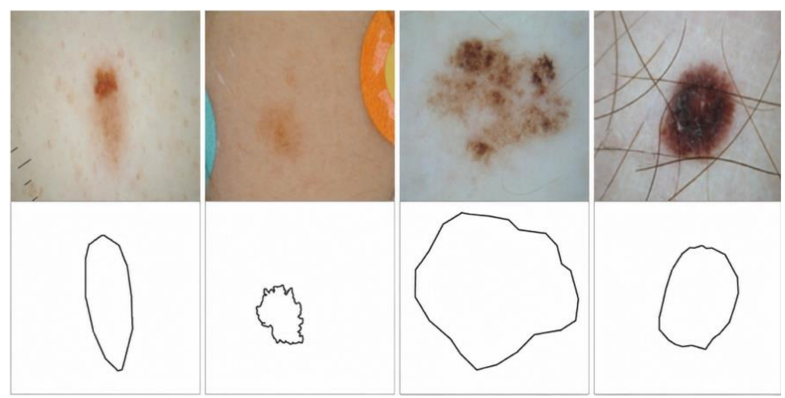
https://github.com/sunsided/kaggle-leaf-classification



https://link.springer.com/article/10.1007/s10618-017-0494-1



https://link.springer.com/chapter/10.1007/978-3-031-41679-8_33



https://challenge.isic-archive.com/landing/2018/45/

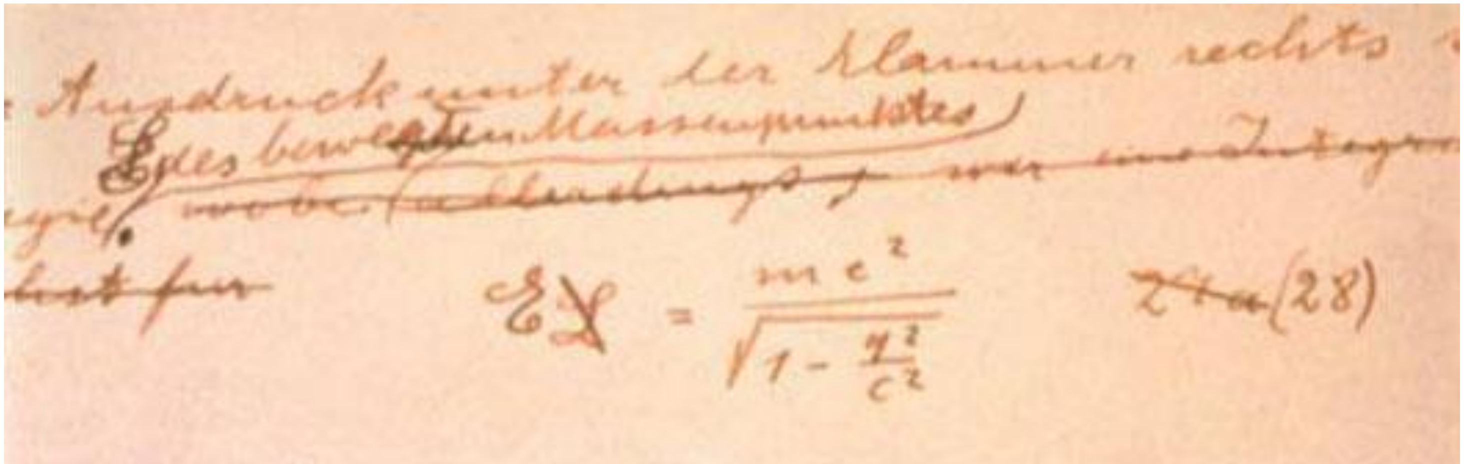
The Common Problem

- Given a sampled curve γ and a set of labelled model curves Σ, classify γ by shape.

- Examples:
  * leaf margin → species
  * lesion border → diagnostic class
  * ink trace → mathematical symbol

- We want a representation that is *compact*, geometric, *fast*, and *trainable from few examples*.

Pen-Based Math



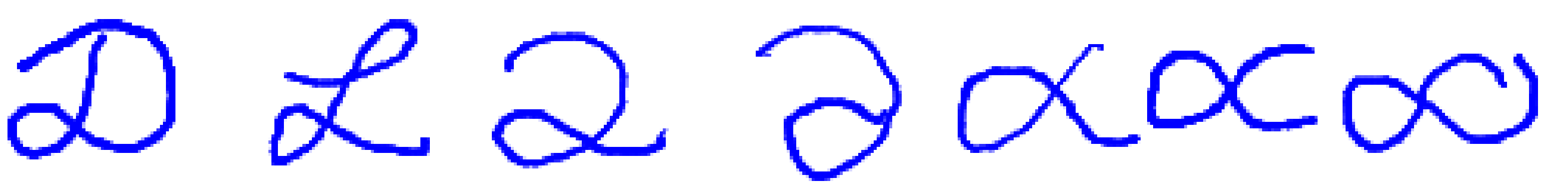
- Input for CAS and document processing.
- 2D editing.
- Computer-assisted collaboration.

Why not just use Deep Neural Networks?

- Deep neural networks are excellent when training data is abundant.
- Here the regime is different:
  * few examples per class
  * new users and idiosyncratic symbols
  * need fast interactive response
  * want meaningful distances between shapes
  * want algebraic operations on curves
- What geometry does the data already have?

Pen-Based Math

- Different than natural language recognition:
  - 2-D layout is a combination of writing and drawing.
  - Many similar few-stroke characters.
  - Many alphabets, used idiosyncratically.
  - Many symbols, each person uses a subset.
  - No fixed dictionary for disambiguation.



Character Recognition

- A story about a UI proposal to Maplesoft
- A story about three statisticians
- Will concentrate on character recognition
- Several projects ignoring this problem

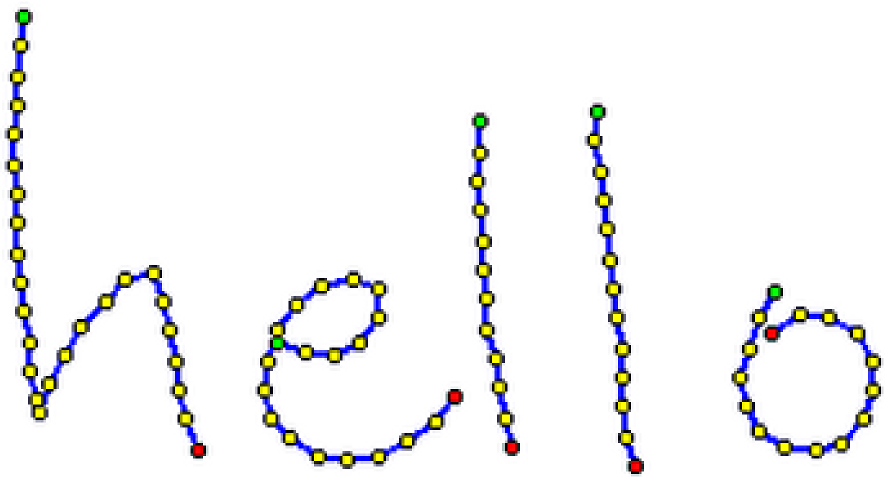
Digital Ink

- Location, time information, sometimes also pressure and angles.
- Capture online pen strokes, *not* images.

- Suitable for
  - **Recognition** algorithms
  - **Semantic** grouping
  - **Annotation**
  - **Manipulation**: search, transformation, archival.

- Problem: Multiple vendor-specific formats.

Digital Ink Formats



- Collected by surface digitizer or camera
- Sequence of (x, y) points sampled at some known frequency
- Possibly other info (angles, pressure, etc)
- Grouping into traces, letters, words + labelling

W3C Recommendation

Ink Markup Language (InkML)

W3C Recommendation 20 September 2011

This version:
: http://www.w3.org/TR/2011/REC-InkML-20110920/

Latest version:
: http://www.w3.org/TR/InkML

Previous version:
: http://www.w3.org/TR/2011/PR-InkML-20110510/

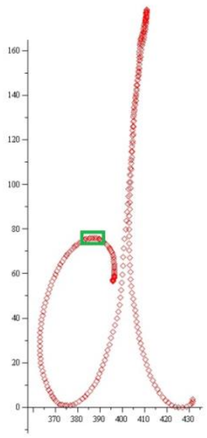
Editors:

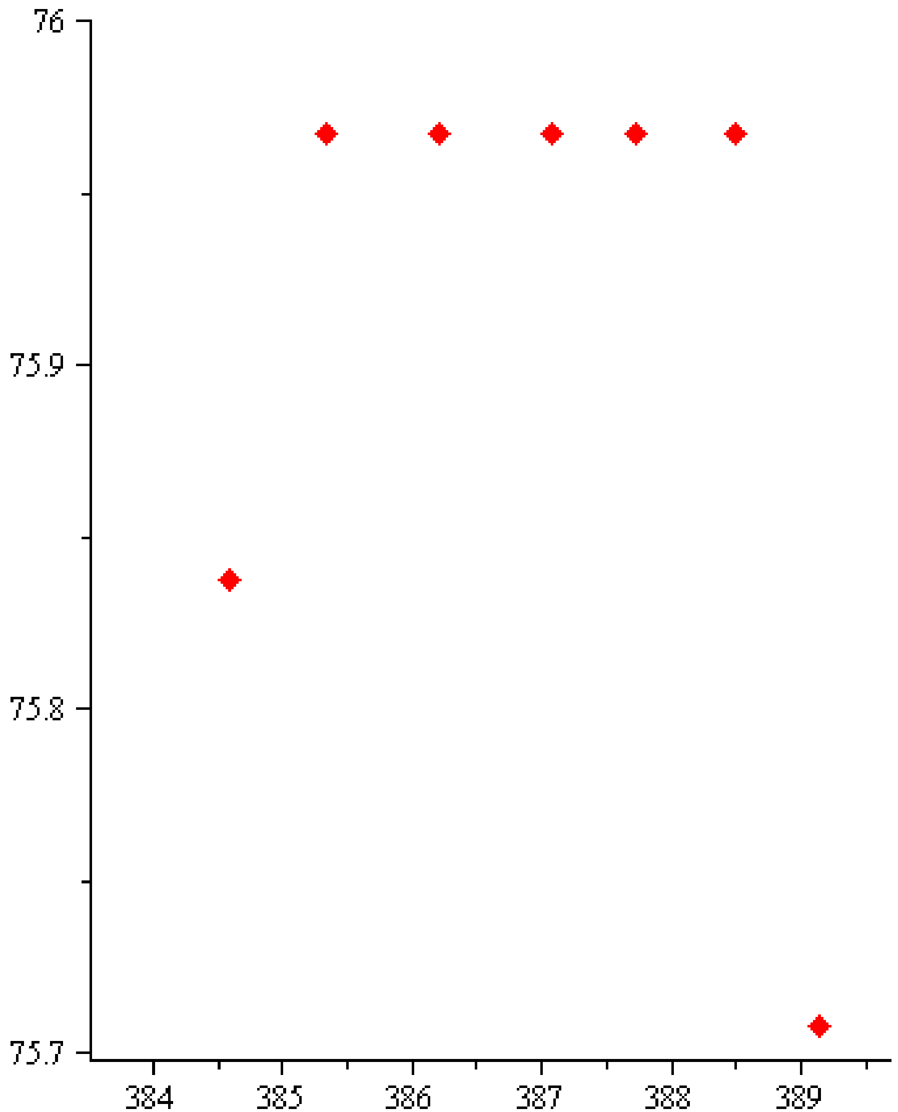
Stephen M. Watt, University of Western Ontario and Maplesoft
Tom Underhill, Microsoft

Authors:

Yi-Min Chee (until 2006 while at IBM)
Katrin Franke (until 2004 while at Fraunhofer Gesellschaft)
Max Froumentin (until 2006 while at W3C)
Sriganesh Madhvanath (until 2009 while at HP)
Jose-Antonio Magaña (until 2006 while at HP)
Grégory Pakosz (until 2007 while at Vision Objects)
Gregory Russell (until 2005 while at IBM)
Muthuselvam Selvaraj (until 2009 while at HP)
Giovanni Seni (until 2003 while at Motorola)
Christopher Tremblay (until 2003 while at Corel)
Larry Yaeger (until 2004 while at Apple)

What the Computer Sees





Usual Character Reco. Methods

- Smooth and re-sample data *THEN*
- Match against *N* models by sequence alignment
  OR
- Identify "features", such as
  - coordinate values of sample points, number of loops, cusps, writing direction at selected points, *etc*

  Use a classification method, such as
  - Nearest neighbour, Subspace projection, Cluster analysis, Support Vector Machine

  THEN
- Rank choices by consulting dictionary

Difficulties

- Having many similar characters (*e.g.* for math) means comparison against all possible symbol models is slow.
- Determining features from points
  - Requires many *ad hoc* parameters.
  - Replaces measured points with interpolations
  - It is not clear how many points to keep, and most methods depend on number of points
  - Device dependent
- What to do since there is no dictionary?
- New ideas are needed!

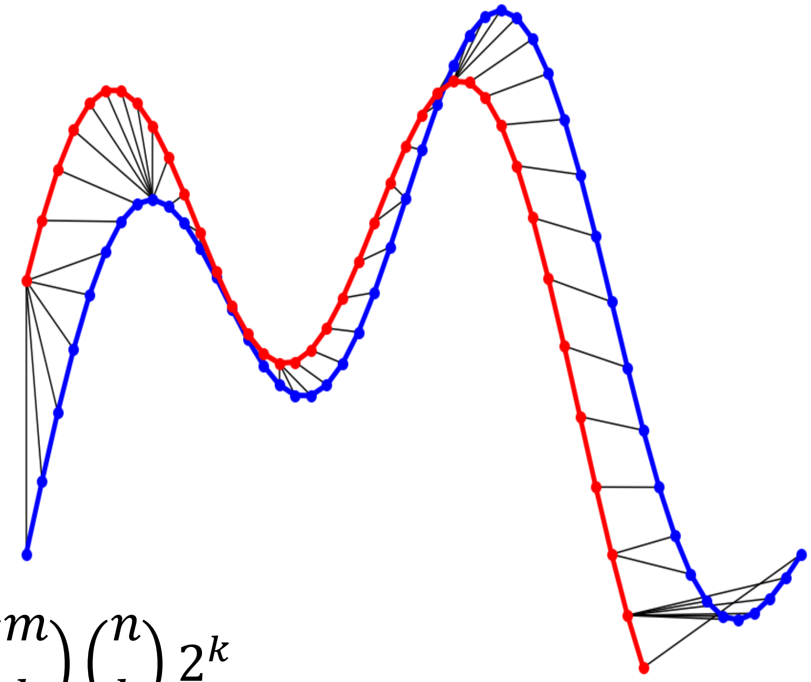
Recognition

- Classification methods compute a similarity between the input curve and models by finding a point correspondence minimizing the sum of squared distances.
- “Elastic matching” with “dynamic time warping”.
- High complexity in the number of sample points.
- Many tricks and heuristics to improve on this.

  E.g. Limit amount of dynamic time warping, pre-classify based on features, ...

Dynamic Time Warping

- Find point correspondence with minimal distance sum.



- Number of matches

$$D(m,n) = \sum_{k=0}^{\min(m,n)} \binom{m}{k} \binom{n}{k} 2^k$$

Delannoy number.

Similarity of Symbols

- Elastic matching:
  - Sequence alignment
  - Interpolation ("resampling")

- Don't we really just want the area?

Polynomial Representation

- **Main ideas**:
  - Represent strokes as parametric curves.
  - Polynomial approximation.
  - Orthogonal polynomial basis.

Polynomial Representation

- **Advantages**:
  - *Compact* – few coefficients needed
  - *Geometric*
    - – the truncation order is a property of the character set
    - – gives a natural metric on the space of characters
  - *Algebraic*
    - – properties of curves can be computed algebraically (instead of numerically using heuristic parameters)
  - *Device independent*
    - – resolution of the device is not important

Space Between Curves

$\bar{x}(t) = x(t) + \xi(t) \qquad \xi(t) \approx \sum_{i=0}^{d} \xi_i \phi_i(t), \qquad \phi_i$ orthogonal with $\langle \phi_i, \phi_j \rangle = \delta_{ij} h_j$.

$\bar{y}(t) = y(t) + \eta(t) \qquad \eta(t) \approx \sum_{i=0}^{d} \eta_i \phi_i(t)$

$$\rho^2(\bar{C}, C) \approx \langle \xi, \xi \rangle + \langle \eta, \eta \rangle$$

$$= \sum_{i=0}^{d} \sum_{j=0}^{d} \xi_i \xi_j \langle \phi_i, \phi_j \rangle + \sum_{i=0}^{d} \sum_{j=0}^{d} \eta_i \eta_j \langle \phi_i, \phi_j \rangle$$

$$= \sum_{i=0}^{d} (\xi_i^2 + \eta_i^2) \langle \phi_i, \phi_i \rangle + cross\ terms$$

$$= \sum_{i=0}^{d} (\xi_i^2 + \eta_i^2) h_i$$

Comparison of Candidate to Models

- Use Euclidean distance in the coefficient space.

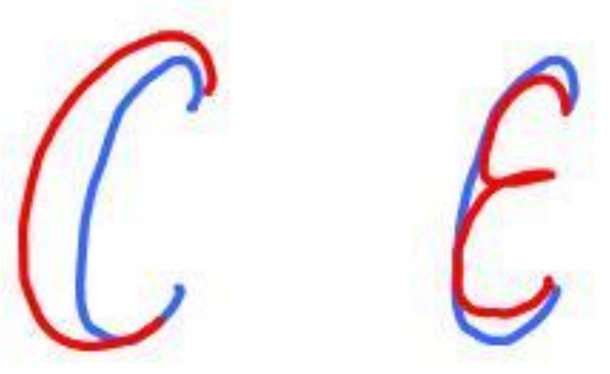
- *Just as accurate* as elastic matching.
- *Much less expensive.*
- Linear in d, the degree of the approximation.
  < 3 d machine instructions (30ns) *vs* several thousand!
- Can trace through SVM-induced cells incrementally.

- Normed space for characters gives other advantages.

Problems

- Want fast response –
  how to work while trace is being captured.

- Low separation area does not mean similar shape.



Problem 1. On-Line Ink

- A problem:

  In handwriting recognition, the human and the computer take turns thinking and sitting idle.

- We ask:

  Can we do useful work while the user is writing and thereby get the answer faster after the user stops writing?

- *The answer is "Yes"!*

On-Line Series Coefficients

- Use Legendre polynomials P_i as basis on the interval $[-1,1]$, with weight function 1.
- Collect numerical values for $f(\lambda)$ on $[0, L]$.
  $\lambda \quad =$ arc length.
  L is not known until the pen is lifted.
- *As the sample points are collected*, numerically integrate the moments $\int \lambda^i f(\lambda) d\lambda$.
- After last point, compute series coefficients for f
  with domain and range scaled to $[-1,1]$.
  This uses a simple linear transformation of the moments.

On-Line Series Coefficients

- Transform moments $\mu_i(f, L)$ of $f(\lambda)$ on $[0, L]$ to coefficients of $\hat{f}(\lambda) = \sum_k \hat{\alpha}_k P_k(\lambda)$ on $[-1, 1]$:

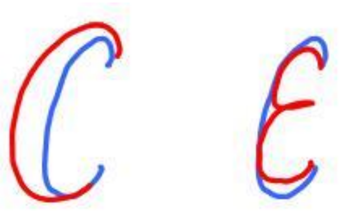
$$\hat{\alpha}_k = (-1)^k \frac{2k+1}{L} \sum_{i=0}^{k} \left(\frac{-1}{L}\right)^i \binom{k}{i} \binom{k+i}{i} \mu_i(f, L)$$

- Normalize range of f:

$$\hat{\alpha}_k \frac{b-a}{f_M - f_m} + \delta_{i0} \frac{a\, f_M - b\, f_m}{f_M - f_m}$$

- Constant time at pen up.

Problem 2. Shape vs Variation



- The corners are not in the right places.
- Work in a jet space to force coords & derivatives close.
- Use a Sobolev inner product

$$\langle f, g\rangle = \int_a^b f(t)g(t)w_0(t)dt$$

$$+\mu_1 \int_a^b f'(t)g'(t)w_1(t)dt + \mu_2 \int_a^b f''(t)g''(t)w_2(t)dt + \cdots$$

- 1^{st} jet space ⇒ set $\mu_i = 0$ for $i > 1$.
  - Choose μ_1 experimentally to maximize reco rate.
  - Can be also done on-line.

    [Golubitsky + SMW 2008, 2009]

Linear Separability

- Can separate N classes with $N(N-1)$ SVM planes.
- Each class is then (mostly) within its own convex polyhedral cell.
- Can classify either by
  - SVM majority voting + run-off elections (96%)
  - Distance to convex hull of *k* nearest neighbours (97.5%).
  - On-line computation.

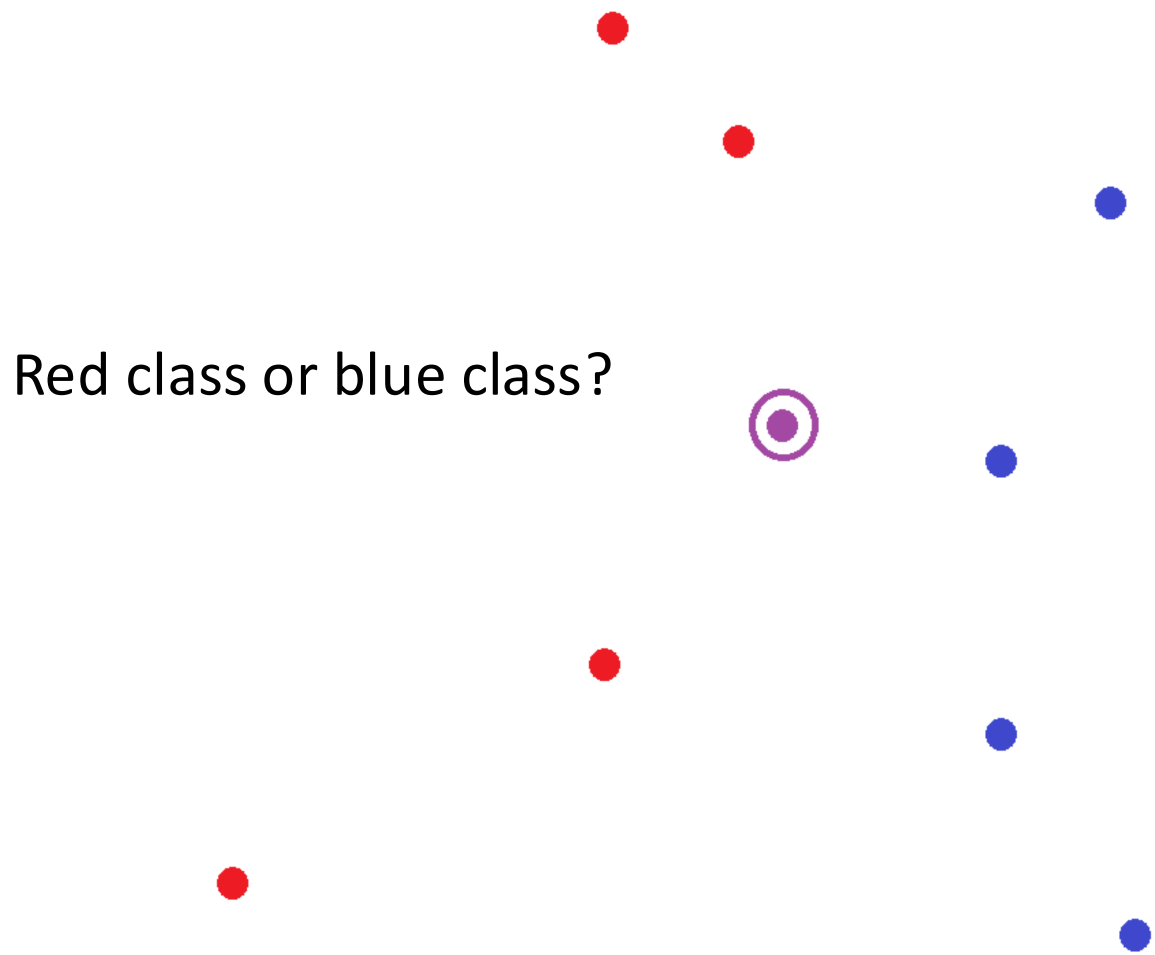
The Joy of Convexity

- Convexity ⇒ Linear homotopies stay within a class

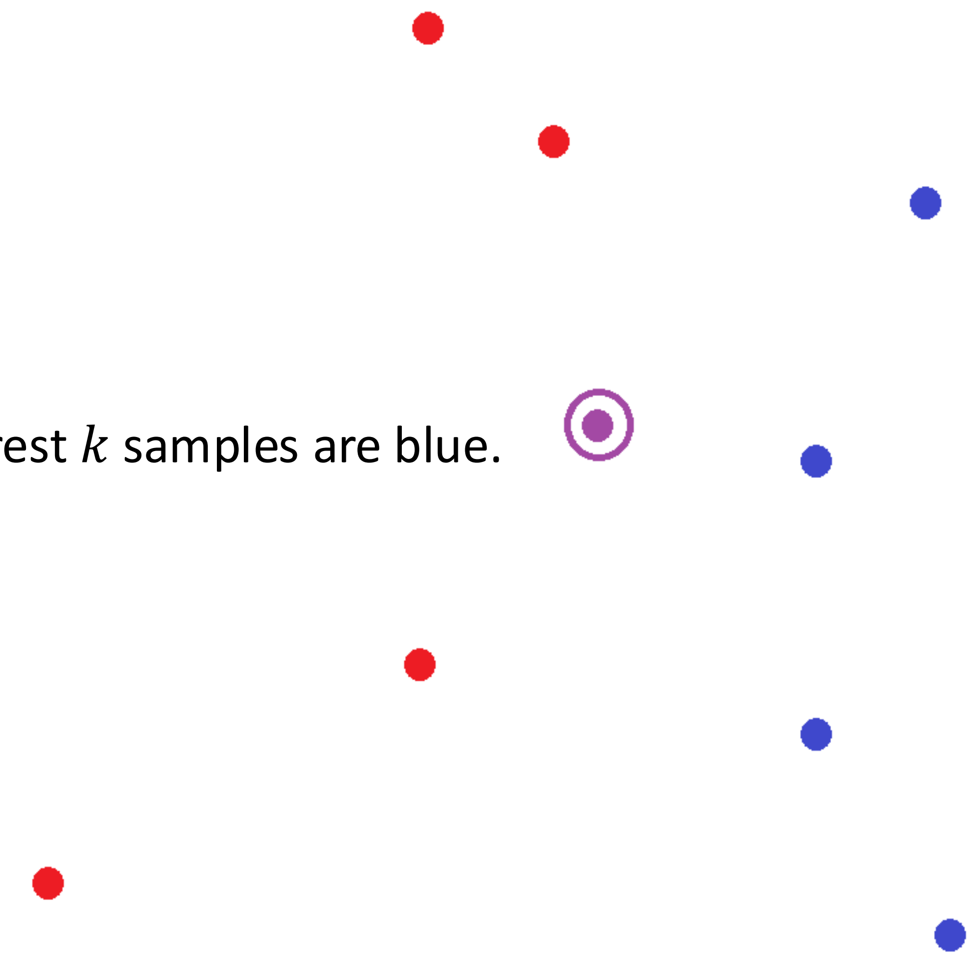
$$C = (1- t)\, A + t\, B$$

- Can compute distance of a sample to this line
- Distance to convex hull of nearest neighbors in class gives best recognition [Golubitsky+SMW 2009,2010]

Choosing Among Alternatives



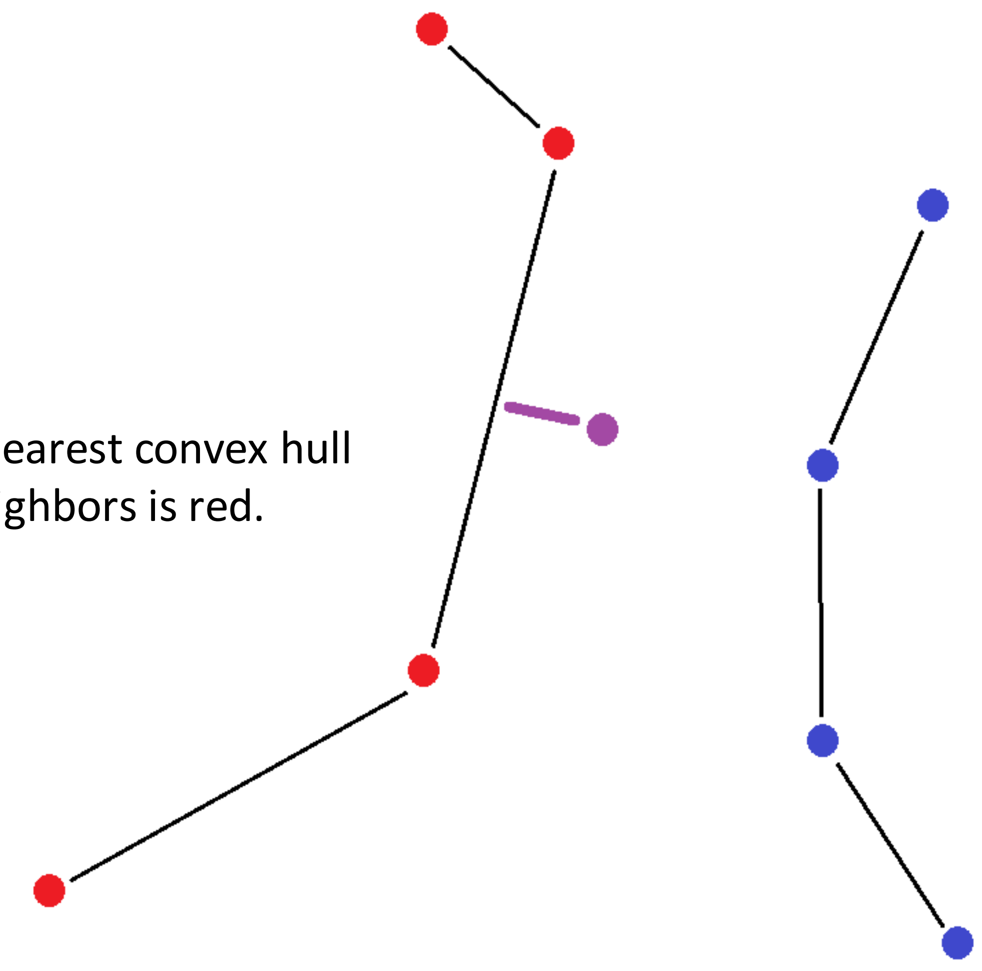
Choosing between Alternatives



The nearest k samples are blue.

Choosing between Alternatives

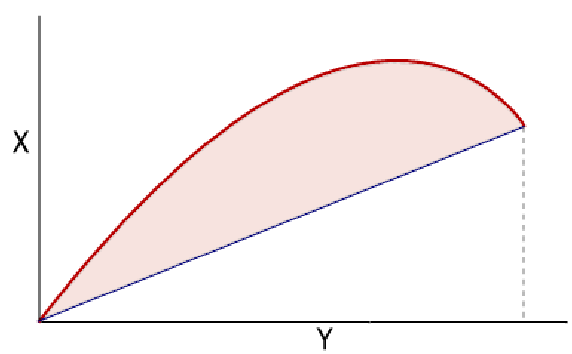
The nearest convex hull of neighbors is red.



Orientation and Shear

- Reco when writing at an angle, or with slanted chars.

- Instead of taking ortho series of coord fns $x(\lambda)$ and $y(\lambda)$, use ortho series of integral invariants of these.
  [Golubitsky, Mazalov, SMW 2009 rotn, 2010 shear]



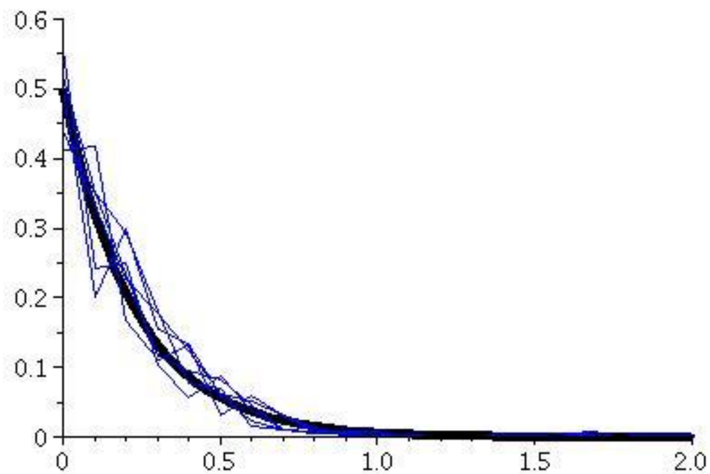
$I_0(\lambda) = radius$ $\qquad$ $I_1(\lambda) = area$

$I_{k>1}(\lambda) = more\ complicated\ integrals$

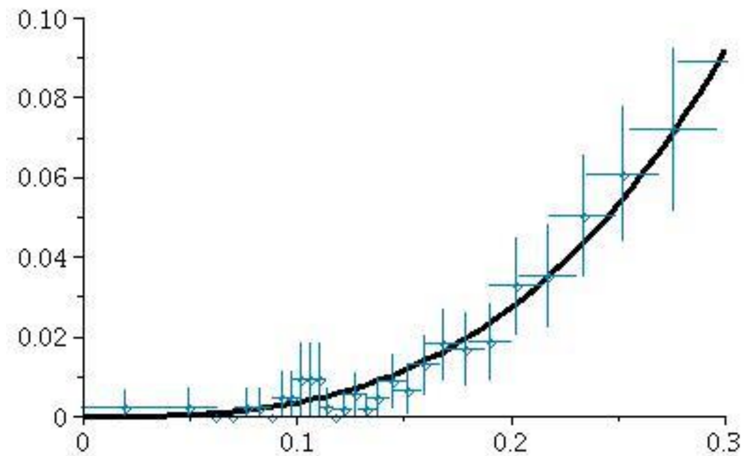
Training

- Using CHKNN allows training with relatively few samples. (Dozens vs Thousands per class)

Error Rates as Fn of Distance



SVM



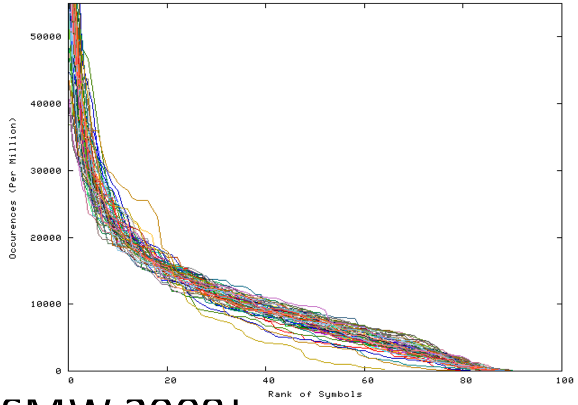
Convex Hull

- Error rate as fn of distance gives confidence measure for classifiers [MKM – Golubitsky + SMW 2009]

Combining with Statistical Info

- Empirical confidence on classifiers allows geometric recognition of isolated symbols
  to be combined with statistical methods.

- Domain-specific *n*-gram information:
  - Research mathematics –
    20,000 articles from arXiv
    [MKM -- So+SMW 2005]
  - 2nd year engineering math –
    most popular textbooks
    [DAS -- SMW 2008]
  - Inverse problem –
    identifying area via *n*-gram freq! [DML -- SMW 2008]



Deciding with Confidence Measure

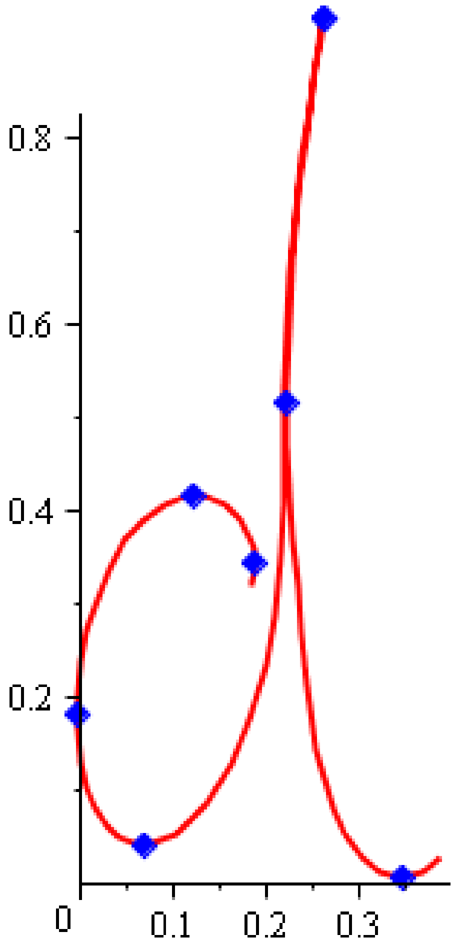
Symbol X in an Expression E

Symbol Recognizer:
$X \in \text{Class}_1$ **with Conf** ξ_1

Context-Based Predictor:
$X \in \text{Class}_2$ **with Conf** ξ_2

$X \in \text{Class}_i \Leftrightarrow \xi_i = \max(\xi_1, \xi_2)$

Sensible Critical Points



- Functional approx uses non-local information.
- Puts critical points where they should be.
- Polynomial common root finding.

$$\frac{dx}{d\lambda} = 0 \quad \frac{dy}{d\lambda} = 0$$

SNC Problems

- Symbolic Numeric Algorithms for Polynomials
- Step away from fuzzy equality in algos for exact computation

- "The Singular Value Decomposition for Polynomial Systems"
  Corless, Gianni, Trager, Watt 1995

  Introduced backward error analysis to symbolic computation:

  E.g. Given two polynomials $p, q \in Z[x]$ of degree $\leq d_{pq}$,
  do there exist $\Delta p, \Delta q$ of degree $\leq d$ and norm $\leq \varepsilon$ such that
  $\gcd(p + \Delta p, q + \Delta q)$ has degree $\geq d_g$ and, if so, find them.

SNC Problems for HWR

- Transformation in and out of monomial basis is ill-conditioned.
- Want to compute resultants, *etc*, without transforming to monomial basis
  - work with Alvandi [Legendre Sobolev] and
  - with Singh Kalhan [Chebyshev-Sobolev]

Real Time Capture

Proposition 1. *Let* $m_k(f,L) = \int_0^L \lambda^k f(\lambda) \mathrm{d}\lambda$ *and* $\hat{f}(\lambda) = \sum_{k=0}^{d} \alpha_k S_k^{\mu}(\lambda)$ *be the scaled function on* $[-1,1]$*. Then* α_k *may be computed as*

$$\alpha_k = \sum_{j=0}^{d} \frac{m_j(f,L)}{L^{j+1}} \; C_{j+1 \;\; k+1}$$

$$C_{ij} = \begin{cases} (-1)^{i+j} c_{ij}(\mu) & \textit{for } i > j \\ (-1)^{i+j} c_{ij}(\mu) + (-1)^{i+j}(2j-1)\binom{j-1}{i-1}\binom{i+j-2}{j-1}\frac{1}{a_{j-1}(\mu)} & \textit{for } i \leq j \end{cases}$$

$$c_{ij}(\mu) = \left(\frac{1}{a_{j-1}(\mu)} - \frac{1}{a_{j+1}(\mu)}\right) \sum_{k=\max(1,\lceil \frac{i-j}{2} \rceil)}^{\lfloor \frac{d+1-j}{2} \rfloor} (2j+4k-1)\binom{j+2k-1}{i-1}\binom{i+j+2k-2}{j+2k-1}$$

$$a_h(\mu) = \sum_{k=0}^{\max(0,\lfloor \frac{h-1}{2} \rfloor)} \left(\frac{\mu}{4}\right)^k \frac{(h+2k-1)!}{(2k)!(h-2k-1)!}.$$

Note that the entries of matrix C are independent of the problem and may be computed in advance.

with Parisa Alvandi

Derivatives

Theorem 2 *We have*

$$\left[\, S_1'(\lambda) \cdots S_n'(\lambda) \,\right]^{\boldsymbol{T}} = D \left[\, S_0(\lambda) \cdots S_{n-1}(\lambda) \,\right]^{\boldsymbol{T}},$$

where $D = H \times N^{-1}$, *and for* $i, j = 1, \ldots, n$, *we have*

$$[D]_{i,j} := \begin{cases} \frac{(2\,i-1)a_i(\mu)}{a_{i-1}(\mu)} & \text{if } i = j \\ \frac{(2\,j-1)a_j(\mu)}{a_{j-1}(\mu)} + (\frac{1}{a_{j-1}(\mu)} - \frac{1}{a_{j+1}(\mu)})\times & \text{if } j = i - 2\,\ell, \\ \sum_{k=0}^{\lfloor \frac{i-j}{2} \rfloor - 1} (2\,i - 4\,k - 1)a_{i-2k}(\mu) & \text{for } \ell = 1, \ldots, \lfloor \frac{i-1}{2} \rfloor \\ 0 & \text{otherwise.} \end{cases}$$

GCD

Theorem 7 *Let* $f(\lambda) = \sum_{i=0}^{n} \alpha_i S_i^{\mu}(\lambda)$ *and* $h(\lambda) = \sum_{i=0}^{m} \beta_i S_i^{\mu}(\lambda)$, *where* $n > m$, *and matrix* B_n *is the matrix defined in Theorem* 6 *for polynomial* $f(\lambda)$. *Let also* $g(\lambda)$ *be the monic gcd of polynomials* $f(\lambda)$ *and* $h(\lambda)$ *and* $g(\lambda) := \lambda^k + \sum_{i=0}^{k-1} \hat{\zeta}_i \lambda^i = \sum_{i=0}^{k} \zeta_i S_i^{\mu}(\lambda)$. *Define* $h(B_n) = \beta_0 I_n + \beta_1 S_1^{\mu}(B_n) + \cdots + \beta_m S_m^{\mu}(B_n)$, *then* $k = n - rank(h(B_n))$. *Furthermore, let* c_i *be the* i*th column of matrix* $h(B_n)$, *for* $i = 1, \dots, n$. *Then the columns* $c_{k+1}, \dots, c_n$ *are linearly independent and if the numbers* $x_{i,j}$ *are defined by*

$$c_i = x_{i,k+1} c_{k+1} + \sum_{j=k+2}^{n} x_{i,j} c_j, \quad \text{for } i = 1, \dots, k, \quad \text{and some } x_{i,j} \in \mathbb{R} \tag{15}$$

then $\zeta_{i-1} = \zeta_k x_{i,k+1}$, *for* $i = 1, \dots, k$, *where* $\zeta_k := (\prod_{\ell=1}^{k} [B_n]_{\ell,\ell+1})^{-1}$.

Can be computed with SVD method [Corless, Gianni, Trager + W 1996]

Bounding Sobolev Norms

Theorem

Let f and g be two polynomials of degree $\leq d$ with coefficient vectors $\mathbf{f}$ and $\mathbf{g}$ in the polynomial basis $\{B_i\}_{i=0}^{d}$ orthogonal wrt the inner product $\langle p, q \rangle_S = \int_0^1 p(t)q(t)w_0(t)dt + \int_0^1 p'(t)q'(t)w_1(t)dt$, and let $\mathbf{D}$ be the differentiation matrix for the basis.

Then the Sobolev norm of the difference between f and g satisfies

$$\|f - g\|_S \leq \sqrt{d+1}\, \|\mathbf{f} - \mathbf{g}\|_\infty (1 + \mu \|\mathbf{D}\|).$$

with Deepak Singh Kalhan

Differentiation Matrices

- Polynomial differentiation is a linear operation so may be represented matrix multiplication on coefficient vectors.
- For basis $\{B_i\}_{i=0}^{d}$, define $\mathbf{D}$ by $B_i'(x) = \sum_{j=0}^{d} D_{ji} B_j(x)$.
- Then $\mathbf{f}' = \mathbf{D}\,\mathbf{f}$, for $f(x) = \sum_{i=0}^{d} f_i B_i(x)$, $f'(x) = \sum_{i=0}^{d} f_i' B_i(x)$.
- Degree 5 examples:

$$\mathbf{D}_{x^i} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ & 0 & 2 & 0 & 0 & 0 \\ & & 0 & 3 & 0 & 0 \\ & & & 0 & 4 & 0 \\ & & & & 0 & 5 \\ & & & & & 0 \end{bmatrix} \qquad \mathbf{D}_{T_i} = \begin{bmatrix} 0 & 1 & 0 & 3 & 0 & 5 \\ & 0 & 4 & 0 & 8 & 0 \\ & & 0 & 6 & 0 & 10 \\ & & & 0 & 8 & 0 \\ & & & & 0 & 10 \\ & & & & & 0 \end{bmatrix}$$

[Amiraslani,C,Gunasingham 2000]

Baseline Estimation

- Figure out baseline from the characters,
rather than the other way around, which is more ususal.

- We can locate some important features by identifying special points.

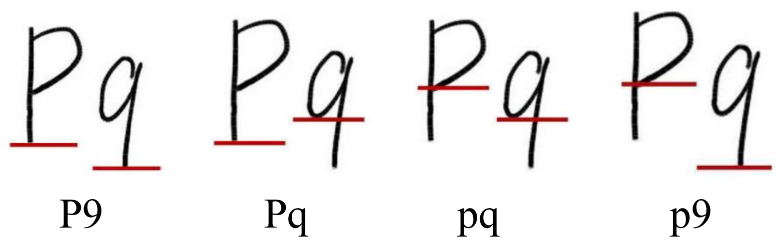
We refer to a point such as this, that determines the height of a metric line, as a *determining point*.

Determining Points

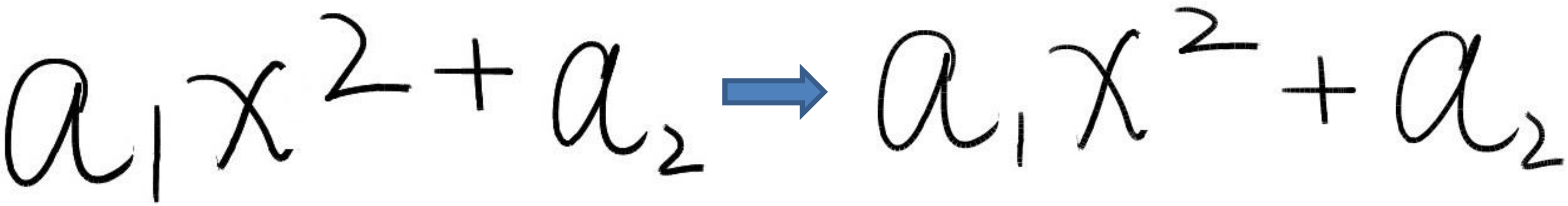
- We focus on European alphabets.
- We consider 6 types of determining points.

Baseline Estimation

- Juxtaposition ambiguity



- Handwriting neatening

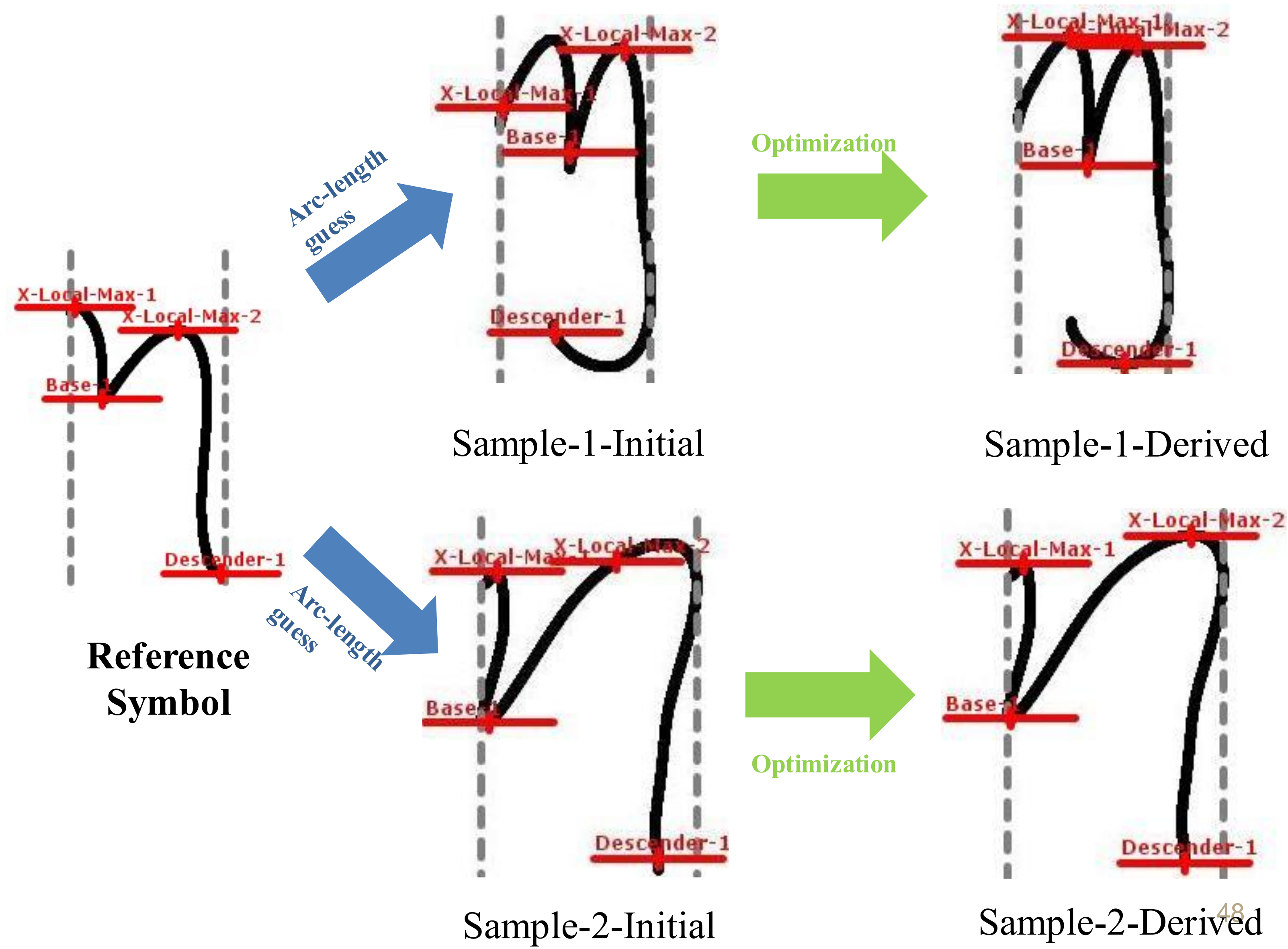


Average Symbol

- The average symbol of a set of known samples for a class can be computed as the average point in the functional space,

$$\bar{C} = \sum_{i=1}^{n} C_i / n$$

Deriving from a Reference Symbol

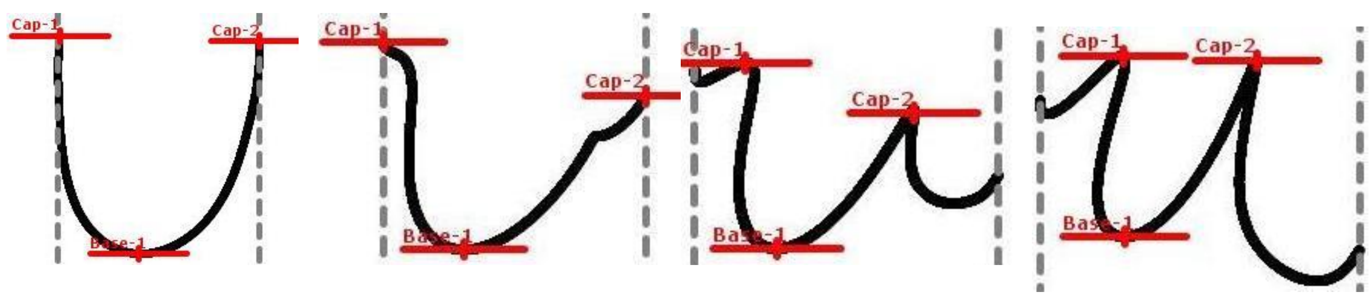


A Homotopy Method

- Some samples are far away from the reference symbol.

Average **Target**

- We use a homotopy between the reference symbol and the target sample in a multi-step method.

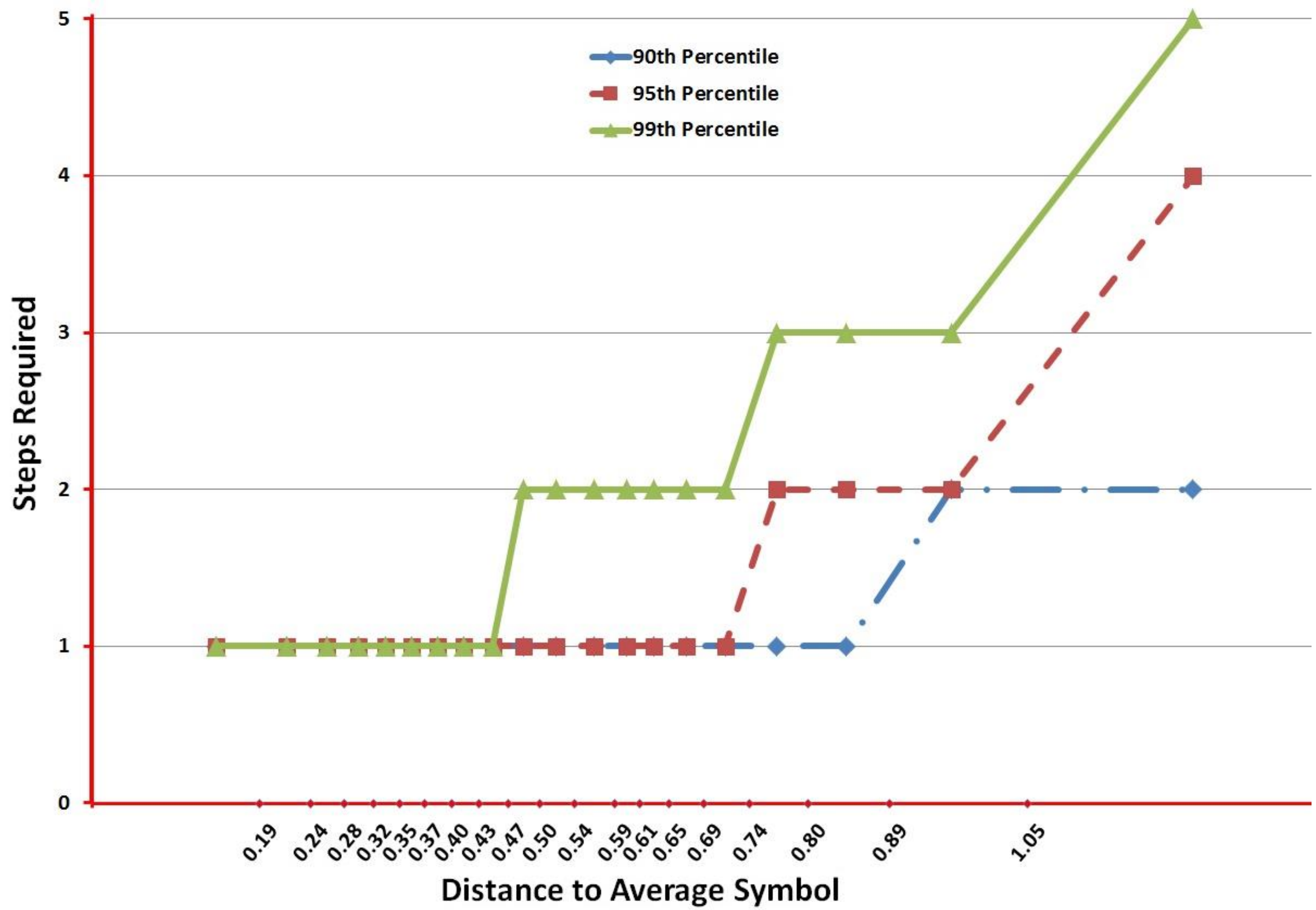


Average **Step-1** **Step-2** **Target**

Measuring Number of Steps Needed

- We went back and tested the data by forgetting the marked points and seeing how many steps it took to recover them.
- For each sample, we tried 1 step, 2 steps, etc up to 30, until a correct result was achieved.
- Samples that did not converge with a 30 step homotopy were considered to fail.
- We examined the ***relationship*** between the ***distance*** to the reference symbols and the ***number of steps*** the homotopy required.

Number of Steps as Fn of Distance to Average Symbol



The overall success rate is 99.63%. Each distance interval in the dense area (x≤0.59) contains 2800 samples while each interval in the sparse area (x>0.59) contains 1500 samples except the last one contains 1370 samples.

Recognition Summary

- Database of samples ⇒ set of LS or ChS points
- Character to recognize ⇒
  - reparameterize by arc-length
  - numerically integrate to get coef. vector
- Classify by distance to convex hull of k-NN.
- Very fast
- Requires very few training points

Conclusions

- Ask what are we really trying to do.
- Work with ink traces as ***curves***,
  rather than as collections of sample points.

- Admits powerful analytic tools.
- Have useful geometry on space of curves.

- Gives device/resolution independence.
- Gives useful insights and faster algorithms.

- Needs only a few examples of each class.
- Compact representation of reference samples.

Conclusions

- The handwriting application is one instance of

  curve classification can improve when we use its geometry rather than converting it too early into pixels, point clouds, or ad hoc features.