Algorithms and Data Structures in Spaces of Curves

Anne Driemel

TU Eindhoven, the Netherlands
A curve is a continuous map

\[ f : [0, 1] \rightarrow \mathbb{R}^d \]
Spaces of Curves

A curve is a continuous map

\[ f : [0, 1] \rightarrow \mathbb{R}^d \]

Example:

time series data
Spaces of Curves

A curve is a continuous map

\[ f : [0, 1] \rightarrow \mathbb{R}^d \]

Example:
time series data
Spaces of Curves

A curve is a continuous map

\[ f : [0, 1] \rightarrow \mathbb{R}^d \]

Example:
time series data
Spaces of Curves

A curve is a continuous map

\[ f : [0, 1] \rightarrow \mathbb{R}^d \]

Example:
time series data

Anne Driemel: Algorithms and data structures for spaces of curves
Machine Learning

Step (1): feature extraction

Step (2): learning the distribution in feature space

Hyndman, Wang, Laptev: *Large-Scale Unusual Time Series Detection*, ICDM Workshops 2015

Anne Driemel: Algorithms and data structures for spaces of curves
Disadvantages of working in feature space:

– features have to be manually designed
– features capture certain aspects only
– features are not always independent of one another
– results depend on the choice of features
Disadvantages of working in feature space:

- features have to be manually designed
- features capture certain aspects only
- features are not always independent of one another
- results depend on the choice of features

**Question:** Is it possible to bypass the feature space and learn the distribution of curves directly?
Overview of this talk

Machine learning in spaces of curves
Overview of this talk

Distribution of curves
- Density estimate
- Median and Depth
- Clustering

Machine learning in spaces of curves
Overview of this talk

Distribution of curves
- Density estimate
- Median and Depth
- Clustering

Classification of curves
- NN-suche
- Range searching
- Decision trees

Machine learning in spaces of curves
Overview of this talk

Machine learning in spaces of curves

Representation
- Dimension reduction
- Overfitting
- Geometric approximation

Distribution of curves
- Density estimate
- Median and Depth
- Clustering

Classification of curves
- NN-suche
- Range searching
- Decision trees
Overview of this talk

- Complexity: Time, Space, Error rate, Sample size
- Distribution of curves: Density estimate, Median and Depth, Clustering
- Representation: Dimension reduction, Overfitting, Geometric approximation
- Classification of curves: NN-suche, Range searching, Decision trees

Machine learning in spaces of curves
Density estimate using sophisticated histograms (KDE)
Only works in low-dimensional space
[Scott *Multivariate Density Estimation* 1950].
Density estimate for curves?

What is the typical shape of a hurricane trajectory?
Let \((X, d)\) be a **metric space** on a set \(X\) with distance function \(d : X^2 \rightarrow \mathbb{R}_{\geq 0}\).

The **median** of a set \(P \subseteq X\) is defined as

\[
\text{argmin}_{c \in X} \sum_{p \in P} d(c, p)
\]
Overfitting of the median

A

B
Overfitting of the median

Anne Driemel: Algorithms and data structures for spaces of curves
Overfitting of the median

\[ A_1, \ldots, A_{n/2} \]

\[ B_1, \ldots, B_{n/2} \]
**Definition:** Let $X_{\ell}^d$ be the set of polygonal curves in $\mathbb{R}^d$ that have exactly $\ell$ edges.

The $\ell$-Median of a set $P \subseteq X_{m}^d$ is defined as

$$\arg\min_{c \in X_{\ell}^d} \sum_{p \in P} d(c, p)$$
** Definition: ** Let $X^d_\ell$ be the set of polygonal curves in $\mathbb{R}^d$ that have exactly $\ell$ edges.

The **$\ell$-Median** of a set $P \subseteq X^d_m$ is defined as

$$\text{argmin}_{c \in X^d_\ell} \sum_{p \in P} d(c, p)$$
Let $C \subseteq X^d_\ell$ be a $k$-set of centers
Let $nn(p, C)$ be the closest center to a point $p$

The $(k, \ell)$-median-clustering is
\[ \arg\min_{C \subseteq X^d_\ell \atop |C| = k} \sum_{p \in P} d(nn(p, C), p) \]

The $(k, \ell)$-center-clustering is
\[ \arg\min_{C \subseteq X^d_\ell \atop |C| = k} \max_{p \in P} d(nn(p, C), p) \]
Most variants of clustering are NP-hard or APX-hard

**k-center:**
- 2-approximation in $O(kn)$ time [Gonzales ’85]

**k-median:**
- $(1 + \sqrt{3} + \varepsilon)$-approximation [Li and Svensson ’13]
- Lower bound for the approximation factor $\sim 1.73$

**k-median, randomised:**
- $(1 + \varepsilon)$-approximation in $O(n2^{(k/\varepsilon)^{O(1)}})$ time
- Algorithm of [Kumar, Sabharwal, Sen ’10]
- Doubling spaces: [Ackermann, Blömer, Sohler ’10]
Distance measure for curves

The Fréchet distance measures the similarity of curves.

What is the minimal distance with which two people can traverse both entire curves?
Clustering of curves

**Single distance computation:**

Given two curves $m_1$ and $m_2$ edges

- $O(m_1 m_2 \log(m_1 + m_2))$
  [Alt and Godau '95] [Buchin et al. '13]

- Lower bound based on SETH $\Omega((m_1 + m_2)^2 - \delta)$ ($\forall \delta > 0$)
  [Bringmann '14] [Bringmann and Mulzer '15]

**1-center (no smoothing):**

Given $n$ curves in $\mathbb{R}^d$ with $m$ edges each

- **weak Fréchet distance:** $O(m^n)$ time and space
  [Har-Peled, Raichel '11]

- **discrete Fréchet distance:** $O(m^n)$ time and space
  [Ahn et al. '15]
**Theorem:**
We can compute an $O(1)$-approximation for the $(k, \ell)$-center-clustering and $(k, \ell)$-median-clustering in $\tilde{O}(nm)$ time. For $d = 1$ we can compute a $(1 + \varepsilon)$-approximation in the same time.

Driemel, Krivosija und Sohler:
*Clustering time series under the Fréchet distance*
SODA 2016
Our results for clustering

**Theorem:**
We can compute an $O(1)$-approximation for the $(k, \ell)$-center-clustering and $(k, \ell)$-median-clustering in $\tilde{O}(nm)$ time. For $d = 1$ we can compute a $(1 + \varepsilon)$-approximation in the same time.

**Sampling-Property:**
For $d = 1$ one can derive a $(1 + \varepsilon)$-approximation of the $(1, \ell)$-median based on a sample of size $m_{\varepsilon\ell}$.

Driemel, Krivosija und Sohler:
*Clustering time series under the Fréchet distance*
SODA 2016
Overview of this talk

- Complexity
  - Time
  - Space
  - Error rate
  - Sample size

- Distribution of curves
  - Density estimate
  - Median and Depth
  - Clustering

- Machine learning in spaces of curves

- Classification of curves
  - NN-suche
  - Range searching
  - Decision trees

- Representation
  - Dimension reduction
  - Overfitting
  - Geometric approximation
A central task in machine learning is the approximation of a function

\[ f : X \rightarrow L \]

\[ X = \text{instances} \]
\[ L = \text{labels} \]

The input is usually a set of examples

\[ \{(p, \ell) \mid p \in X, \ell \in L\} \]
Classification of curves


Anne Driemel: Algorithms and data structures for spaces of curves
Nearest-neighbor searching
Nearest-neighbor searching

![Diagram of nearest-neighbor searching with query point q and data points labeled 1 and 2.](Image)
Nearest-neighbor searching

Answer:
The label of the example instance closest to $q$
Nearest-neighbor searching

Answer:
The label of the example instance closest to $q$

(1) Partition the space into homogeneous areas

$$f : X \to \mathcal{L}$$

(2) Data structure for point location

- label 1
- label 2
Nearest-neighbor searching

Answer:
The label of the example instance closest to $q$

(1) Partition the space into homogeneous areas

$$f : X \rightarrow \mathcal{L}$$

(2) Data structure for point location

Voronoi diagram

$$\Theta(n^{\lceil d/2 \rceil})$$
A set of **hash functions** $H$ is called $(d_1, d_2, p_1, p_2)$-locality-sensitive if it holds for all $p, q \in \mathbb{R}^d$:

(a) \quad \text{if } d(p, q) \leq d_1 \quad \text{then} \quad \Pr [h(p) = h(q)] \geq p_1

(b) \quad \text{if } d(p, q) \geq d_2 \quad \text{then} \quad \Pr [h(p) = h(q)] \leq p_2
Our results for NN-searching

We show LSH-families for the discrete Fréchet distance

**input:** $n$ curves from $X^d_m$.

<table>
<thead>
<tr>
<th>Space</th>
<th>Query time</th>
<th>Approximation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$O(mn)$</td>
<td>$O(m^2n)$</td>
<td>$1$</td>
</tr>
<tr>
<td>$O\left(</td>
<td>U</td>
<td>\sqrt{m}(m\sqrt{m}n)^2\right)$</td>
</tr>
<tr>
<td>$O(n \log n + mn)$</td>
<td>$O(m \log n)$</td>
<td>$O(m)$</td>
</tr>
<tr>
<td>$O(2^{4md}n \log n + mn)$</td>
<td>$O(2^{4md}m \log n)$</td>
<td>$O(1)$</td>
</tr>
<tr>
<td>$O(2^{2k}m^{k-1}n \log n + mn)$</td>
<td>$O(2^{2k}m^{k} \log n)$</td>
<td>$O(m/k)$</td>
</tr>
</tbody>
</table>

Driemel und Silvestri: *Locality-sensitive-hashing for curves*
SoCG 2017
Overview of this talk

- Complexity
  - Time
  - Space
  - Error rate
  - Sample size

- Representation
  - Dimension reduction
  - Overfitting
  - Geometric approximation

- Distribution of curves
  - Density estimate
  - Median and Depth
  - Clustering

- Classification of curves
  - NN-suche
  - Range searching
  - Decision trees

Machine learning in spaces of curves
Range searching for curves

Which hurricane trajectories are similar to the query curve $q$?
Range searching for curves

ACM SIGSPATIAL Cup 2017

Problem Definition

Input: A file dataset.txt containing paths to trajectory files making up the dataset. Each referenced files is in a SSV format with the two holding coordinates for X and Y. Note that time stamps are not given as the Fréchet distance does not incorporate absolute time.

Parameters: A file queries.txt containing pairs of a query trajectory file names (as given in dataset.txt) and Fréchet distance bounds.

Output: A set of file names for each query written to result-XXXX.txt, where XXXX stands for the line number of the query (starting from 0).

Assumptions:

• Irrespective of the actual nature of the coordinates (e.g., WGS84), we will always use Euclidean distance between the coordinate points.
A range query $(q, r)$ among a set of curves $S \subseteq X^d_m$ should return the set

$$\{ p \in S \mid d(p, q) \leq r \}$$

The range is defined as the metric ball of radius $r$ centered at $q \in X^d_{\ell}$. 
Range searching for curves

Only known data structure for this type of queries:
[de Berg, Cook and Gudmundsson ’13]

For any parameter value $1 \leq s \leq \sqrt{n}$
- approximation $\approx 6.24$
- query time in $O(s \text{ polylog } n)$
- space in $O((\frac{n}{s})^2 \text{ polylog } n)$

Queries: Only supports query curves from $X_1^2$
Output: Count of the curves in the range
Our results for range searching

**Theorem:**
Assume there exists a data structure for Fréchet range queries among an $n$-set from $X^d_m$ with
- Space in $\leq S(n)$
- Query time in $\leq Q(n) + O(k)$ ($k$ is output size)

Then it holds

$$S(n) = \Omega \left( \left( \frac{n}{Q(n)} \right)^2 \right)$$

Our proof uses a volume argument of [Afshani '13]

\[ Q = [0, 1]^2 \]

**Input:**
Slabs in \( \mathbb{R}^2 \)

**Output:**
Slabs that contain \( q \)
Range searching for curves

Our proof uses a volume argument of [Afshani ’13]

\[ Q = [0, 1]^2 \]

**Input:**
Slabs in \( \mathbb{R}^2 \)

**Output:**
Slabs that contain \( q \)

Input construction with output size \( k = t \)
Range searching for curves

Our proof uses a volume argument of [Afshani ’13]

\[ Q = [0, 1]^2 \]

**Input:**
Slabs in \( \mathbb{R}^2 \)

**Output:**
Slabs that contain \( q \)

Input construction with output size \( k = t \)
Our proof uses a volume argument of [Afshani ’13]

\[ Q = [0, 1]^2 \]

**Input:**
Slabs in \( \mathbb{R}^2 \)

**Output:**
Slabs that contain \( q \)

Input construction with output size \( k = t \)
Range searching for curves

Our proof uses a volume argument of [Afshani '13]

\[ Q = [0, 1]^2 \]

**Input:**
Slabs in \( \mathbb{R}^2 \)

**Output:**
Slabs that contain \( q \)

Input construction with output size \( k = t \)

\[ S(n) \geq \frac{t}{v_{\text{max}}} \]
Our proof uses a volume argument of [Afshani ’13]

\[ Q = [0, 1]^2 \]

**Input:**
Slabs in \( \mathbb{R}^2 \)

**Output:**
Slabs that contain \( q \)

Input construction with output size \( k = t \)

\[ S(n) \geq \frac{t}{v_{\text{max}}} \geq \left( \frac{n}{t} \right)^2 \]
Our proof uses a volume argument of [Afshani ’13]

\[ Q = [0, 1]^2 \]

**Input:**
Slabs in \( \mathbb{R}^2 \)

**Output:**
Slabs that contain \( q \)

Input construction with output size \( k = t \)

\[
S(n) \geq \frac{t}{v_{\max}} \geq \left( \frac{n}{t} \right)^2 = \left( \frac{n}{Q(n)} \right)^2
\]
Dualization of range queries

input space

query space
Range searching for curves

Restrict queries to $X_1^2$ (Single edges)

What is the locus of queries that output this curve?
Range searching for curves

Restrict queries to $X^2_1$ (Single edges)

What is the locus of queries that output this curve?
Restrict queries to $X_1^2$ (Single edges)
What is the locus of queries that output this curve?
Range searching for curves

Restrict queries to $X_1^2$ (Single edges)

What is the locus of queries that output this curve?
Represent query lines in the dual space \((\beta, \alpha)\)
Represent query lines in the dual space \((\beta, \alpha)\)

Lines stabbing the disk at \((x_c, y_c)\) with radius \(r\):

\[
\alpha \leq x_c \sin(\beta) + y_c \cos(\beta) + r
\]

\[
\alpha \geq x_c \sin(\beta) + y_c \cos(\beta) - r
\]
Range searching for curves

Dualization of range queries for curves

input space

query space

Anne Driemel: Algorithms and data structures for spaces of curves
Range searching for curves

Dualization of range queries for curves

input space

query space
Range searching for curves

Dualization of range queries for curves

input space  query space

Anne Driemel: Algorithms and data structures for spaces of curves
Sketching the input construction for the lower bound:
– Fix the input curves at their endpoints
Range searching for curves

Sketching the input construction for the lower bound:
- Fix the input curves at their endpoints
- A query line outputs an input curve \( p_{ij} \) iff the query line intersects the vertical grid edge at \((i, j)\)
Sketching the input construction for the lower bound:

- Fix the input curves at their endpoints
- A query line outputs an input curve $p_{ij}$ iff the query line intersects the vertical grid edge at $(i, j)$
Sketching the input construction for the lower bound:

- Fix the input curves at their endpoints
- A query line outputs an input curve $p_{ij}$ iff the query line intersects the vertical grid edge at $(i, j)$
Sketching the input construction for the lower bound:

- Fix the input curves at their endpoints
- A query line outputs an input curve $p_{ij}$ iff the query line intersects the vertical grid edge at $(i, j)$
Range searching for curves

Two input curves in the query space

\[
\alpha \\
\beta
\]

\[Q\]
Range searching for curves

Two input curves in the query space
Lower bounds for range searching

**Theorem:**
Assume there exists a data structure for Fréchet range queries in $X^d_\ell$ among an $n$-set from $X^d_m$ with
- Space in $\leq S(n)$
- Query time in $\leq Q(n) + O(k)$  ($k$ is output size)

Then it holds

$$S(n) = \Omega \left( \left( \frac{n}{Q(n)} \right)^2 \right)$$

Driemel und Afshani, *On the complexity of range searching among curves* (Manuscript)
Assume there exists a data structure for Fréchet range queries in $X^d_{\ell}$ among an $n$-set from $X^d_m$ with
- Space in $\leq S(n)$
- Query time in $\leq Q(n) + O(k)$  ($k$ is output size)

Then it holds

$$S(n) = \Omega \left( \left( \frac{n}{Q(n)} \right)^2 \cdot \left( \frac{\log(n/Q(n))}{\ell^2} \right)^{\ell-2} \right)$$

Driemel und Afshani, *On the complexity of range searching among curves* (Manuscript)
Data structures

We can match the lower bound with a multi-level partition tree that has the correct number of levels:

**Theorem (discrete Fréchet):**
We can build a data structure for discrete Fréchet range queries in $X^d_\ell$ among an $n$-set from $X^d_m$ with

- Space in $O(n(\log \log n)^{m-1})$
- Query time in $O(n^{1-1/d} \log^{O(m)} n \cdot \ell^{O(d)} + k)$

(where $k$ is the output size)

assuming $\ell \in O(\text{polylog } n)$.

Driemel und Afshani, *On the complexity of range searching among curves* (Manuscript)
We can match the lower bound with a multi-level partition tree that has the correct number of levels:

**Theorem (continuous Fréchet):**
We can build a data structure for continuous Fréchet range queries in $X_\ell^2$ among an $n$-set from $X_m^2$ with

- Space in $O(n (\log \log n)^{O(m^2)})$
- Query time in $O(\sqrt{n} \log^{O(m)} n + k)$

(where $k$ is the output size)

assuming $\ell \in O(\text{polylog } n)$.

Driemel und Afshani, *On the complexity of range searching among curves* (Manuscript)
Data structure for discrete Fréchet

Sketch of the data structure

(input curves)
Data structure for discrete Fréchet

Sketch of the data structure

Multi-level partition tree

(input curves)
Data structure for discrete Fréchet

Sketch of the data structure

Multi-level partition tree

(input curves)
Data structure for discrete Fréchet

Sketch of the data structure

Multi-level partition tree

(input curves)
Data structure for discrete Fréchet

Sketch of the data structure

Multi-level partition tree

(input curves)
Data structure for discrete Fréchet

Sketch of the query algorithm

(query curve)
Data structure for discrete Fréchet

Sketch of the query algorithm

(query curve)
Sketch of the query algorithm

(query curve)
Data structure for discrete Fréchet

Sketch of the query algorithm

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>q1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>q2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>q3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>q4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>q5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Data structure for discrete Fréchet

Sketch of the query algorithm

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$q_2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$q_3$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$q_4$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$q_5$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$P_1$
Data structure for discrete Fréchet

Sketch of the query algorithm

<table>
<thead>
<tr>
<th></th>
<th>$q_1$</th>
<th>$q_2$</th>
<th>$q_3$</th>
<th>$q_4$</th>
<th>$q_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

$P_2$
Data structure for discrete Fréchet

Sketch of the query algorithm

<table>
<thead>
<tr>
<th></th>
<th>q1</th>
<th>q2</th>
<th>q3</th>
<th>q4</th>
<th>q5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

$P_3$
Data structure for discrete Fréchet

Sketch of the query algorithm

Anne Driemel: Algorithms and data structures for spaces of curves
Data structure for discrete Fréchet

Sketch of the query algorithm

<table>
<thead>
<tr>
<th></th>
<th>q1</th>
<th>q2</th>
<th>q3</th>
<th>q4</th>
<th>q5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

feasible alignment
Data structures

We can match the lower bound with a multi-level partition tree that has the correct number of levels:

**Theorem (discrete Fréchet):**
We can build a data structure for discrete Fréchet range queries in $X^d_\ell$ among an $n$-set from $X^d_m$ with
- Space in $O(n \cdot (\log \log n)^{m-1})$
- Query time in $O(n^{1-1/d} \log^{O(m)} n \cdot \ell^{O(d)} + k)$
  (where $k$ is the output size)
assuming $\ell$ in $O(\text{polylog } n)$.

Driemel und Afshani, *On the complexity of range searching among curves* (Manuscript)
Data structures

We can match the lower bound with a multi-level partition tree that has the correct number of levels:

**Theorem (continuous Fréchet):**
We can build a data structure for continuous Fréchet range queries in $X^2_\ell$ among an $n$-set from $X^2_m$ with
- Space in $O(n \cdot (\log \log n)^{O(m^2)})$
- Query time in $O(\sqrt{n} \log^{O(m)} n + k)$

(where $k$ is the output size)
assuming $\ell$ in $O(\text{polylog } n)$.

Driemel und Afshani, *On the complexity of range searching among curves* (Manuscript)
Overview of this talk

**Complexity**
- Time
- Space
- Error rate
- Sample size

**Distribution of curves**
- Density estimate
- Median and Depth
- Clustering

**Representation**
- Dimension reduction
- Overfitting
- Geometric approximation

**Classification of curves**
- NN-suche
- Range searching
- Decision trees

**Machine learning in spaces of curves**
We talked about:

(1) Clustering for curves
(2) Data structuring for curves
(3) Machine learning for curves

Challenges:

– Curse of dimensionality
– Complexity of the distance measure
– Risk of overfitting
Some open problems

- **Clustering**
  
  Practical algorithms for clustering of curves
  More robust definitions of median and depth

- **NN-Searching**

  LSH for continuous Fréchet distance
  Randomized curve simplification

- **Range-Searching**

  DS for high-space/ low-query-time regime
  Continuous Fréchet distance in 3D (seems much harder)