Studying complicated fluid dynamics with persistent homology

Rachel Levanger (Rutgers University)

ATHDDA, University of Victoria, Victoria, BC

August 28, 2015

Two problems:

- Dealing with lots of data (e.g. point cloud with $\approx 70K$ points)
- Organizing lots of hard-to-visualize non-linear data (e.g. 3D simulations of fluid flows)

Problem one: Large point clouds.

Rayleigh-Bénard Convection: An example of complicated, almost-periodic 2D dynamics.

Goal: Want to show that we see a loop (periodic structure) in the space of persistence diagrams.

First attempt: Sample trajectory of solutions at regular time-intervals over $\approx$2 periods, obtaining 500 sample points.

General method:

- Convert each scalar field (sample point) to a vector of persistence diagrams.
- Choose a metric from the space of persistence diagrams and compute the pairwise distances between each to get a distance matrix D_{ij}.
- Use the distance matrix to compute a filtered Vietoris-Rips Complex.
- Produce a vector of persistence diagrams from this filtration.

Video.

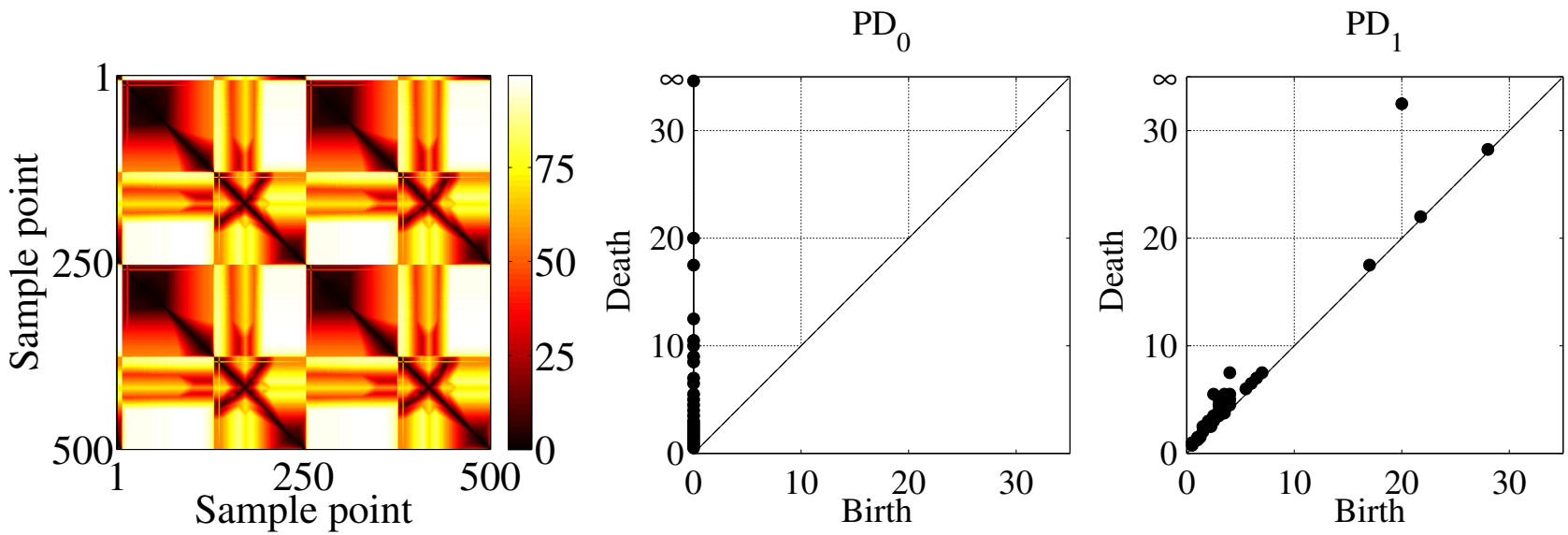


Figure: (a) Distance matrix for 500 sample points and persistence diagrams for Vietoris-Rips complex filtration at (b) H_0 and (c) H_1.

Conclusion: Trajectory not sampled densely enough to resolve the periodic dynamics.

Second attempt: Re-sample trajectory of solutions at faster time intervals over $\approx$4.5 periods.

- Get max consecutive d_B distance of 4 (vs. 83.5 in first sampling).
- Obtain $70,000K$ sample points.

Challenges:

- Cannot compute the distance matrix, since $n^2/2$ computations of Bottleneck (or Wasserstein) distance will take too long.
- Explosion in the number of simplices, so cannot compute the Vietoris-Rips Complex filtration.
- Explosion in the number of simplices, so cannot compute persistent homology on filtered complex.

Technique: Subsampling the point cloud.

Remark

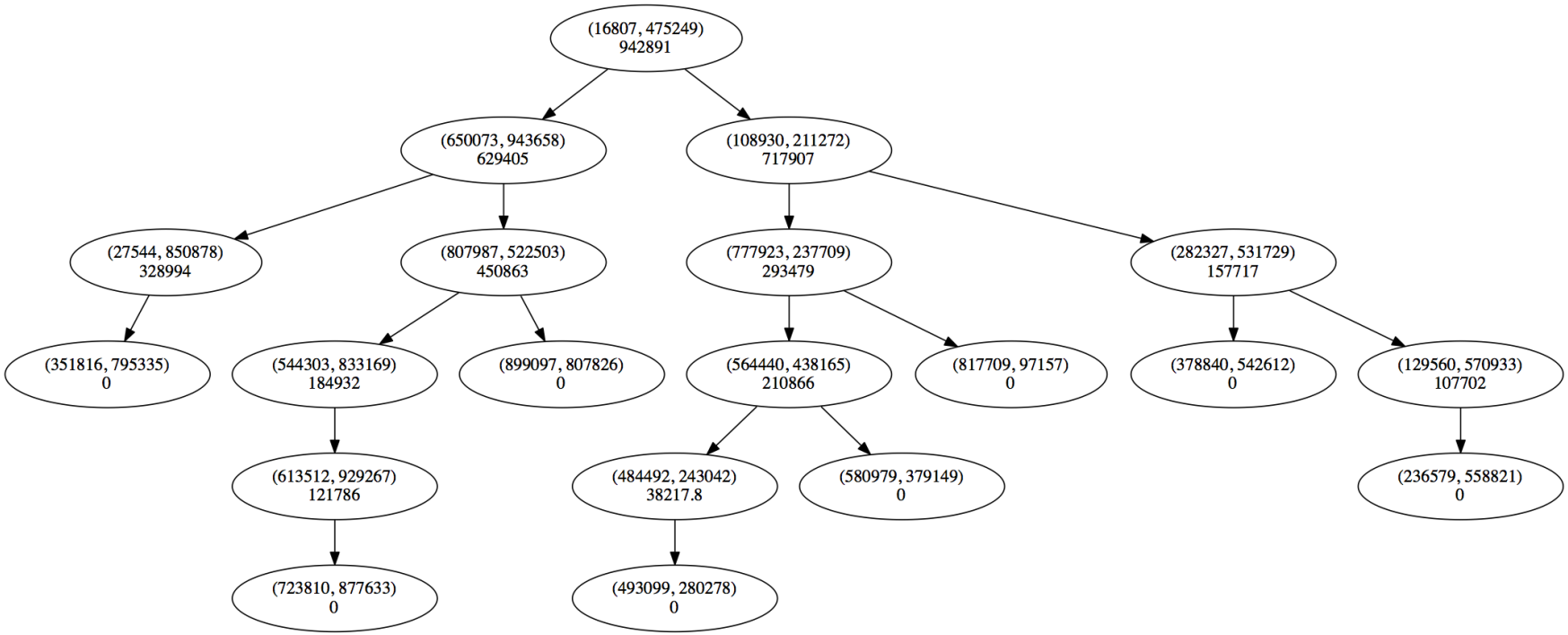
Let X be a finite metric space with metric function d and Y a δ-dense subsample of X. Then $d_B(\mathrm{PD}(X), \mathrm{PD}(Y)) \leq \delta$, where d_B is the bottleneck distance between the persistence diagrams.

To "minimize" the size of $Y \subseteq X$, also want Y to be δ-sparse, so that for any $y_1, y_2 \in Y$, $d(y_1, y_2) \geq \delta$.

[Dey, et. al, "Graph Induced Complex on Point Data," 2013.]

Speed-up: Parallel processing for distance calculations and leveraging a metric tree data structure.

- Metric tree structure yields best-case $O(\log n)$ time for searches by using "branch-and-bound" strategy. We need to find the nearest neighbor.
  - On metric tree structure with $\approx 2M$ nodes, finding a closest point resulted in 94 computations.
  - Compare to $\approx 2M$ computations for brute force method.
- Computation of subsample on 2D example in $\mathbb{R}^2$ with 1,000,000 points yields:
  - Approx 1.5 hrs on a laptop (4 cores)
  - 48,495,151 distance computations to produce subsample of 134,961 points
  - full distance matrix for subsample would have required 18,214,471,521/2 computations
  - 185 times as many computations!


(16807, 475249)
942891
(650073, 943658)
629405
(108930, 211272)
717907
(27544, 850878)
328994
(807987, 522503)
450863
(777923, 237709)
293479
(282327, 531729)
157717
(351816, 795335)
0
(544303, 833169)
184932
(899097, 807826)
0
(564440, 438165)
210866
(817709, 97157)
0
(378840, 542612)
0
(129560, 570933)
107702
(613512, 929267)
121786
(484492, 243042)
38217.8
(580979, 379149)
0
(236579, 558821)
0
(723810, 877633)
0
(493099, 280278)
0

For Rayleigh-Bénard Convection $70,000K$ point cloud:

- Computed subsample for $\delta = 4.5$ with Bottleneck distance d_B.
- Obtain 523 points in subsample.
- Subsampling took a few hours using just 75 cores.

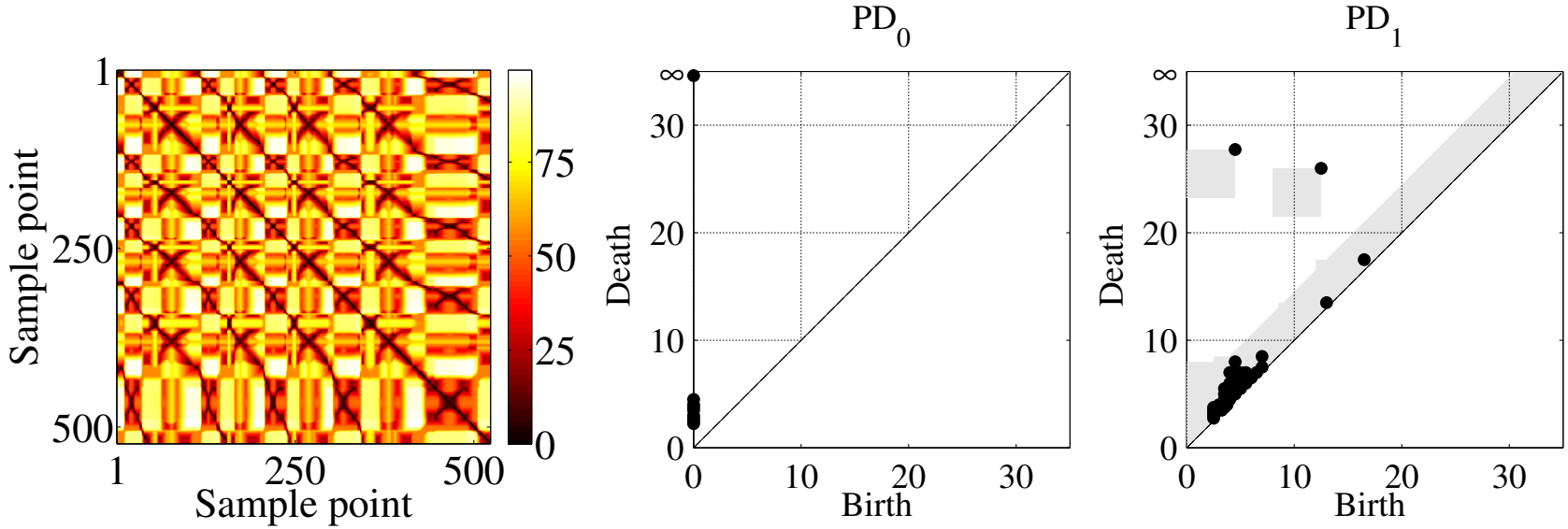
Code available at:
`https://github.com/shaunharker/subsample`

Rigorous estimation of persistence diagrams:
Using the Induced Matching Theorem (Bauer, Lesnick, 2015)

- Can use the inclusion maps $f : \mathcal{R}(Y) \hookrightarrow \mathcal{R}(X)$ at the level of Vietoris-Rips Complexes to leverage the induced matching theorem.
- Can show that coker f and $\ker f$ are δ-trivial at the level of homology.
- Hence, the induced matching yields $\langle b, d \rangle \mapsto \langle b', d' \rangle$ with

$$b' \leq b < d' \leq d,$$

$d - d' \leq \delta$, and $b - b' \leq \delta$.



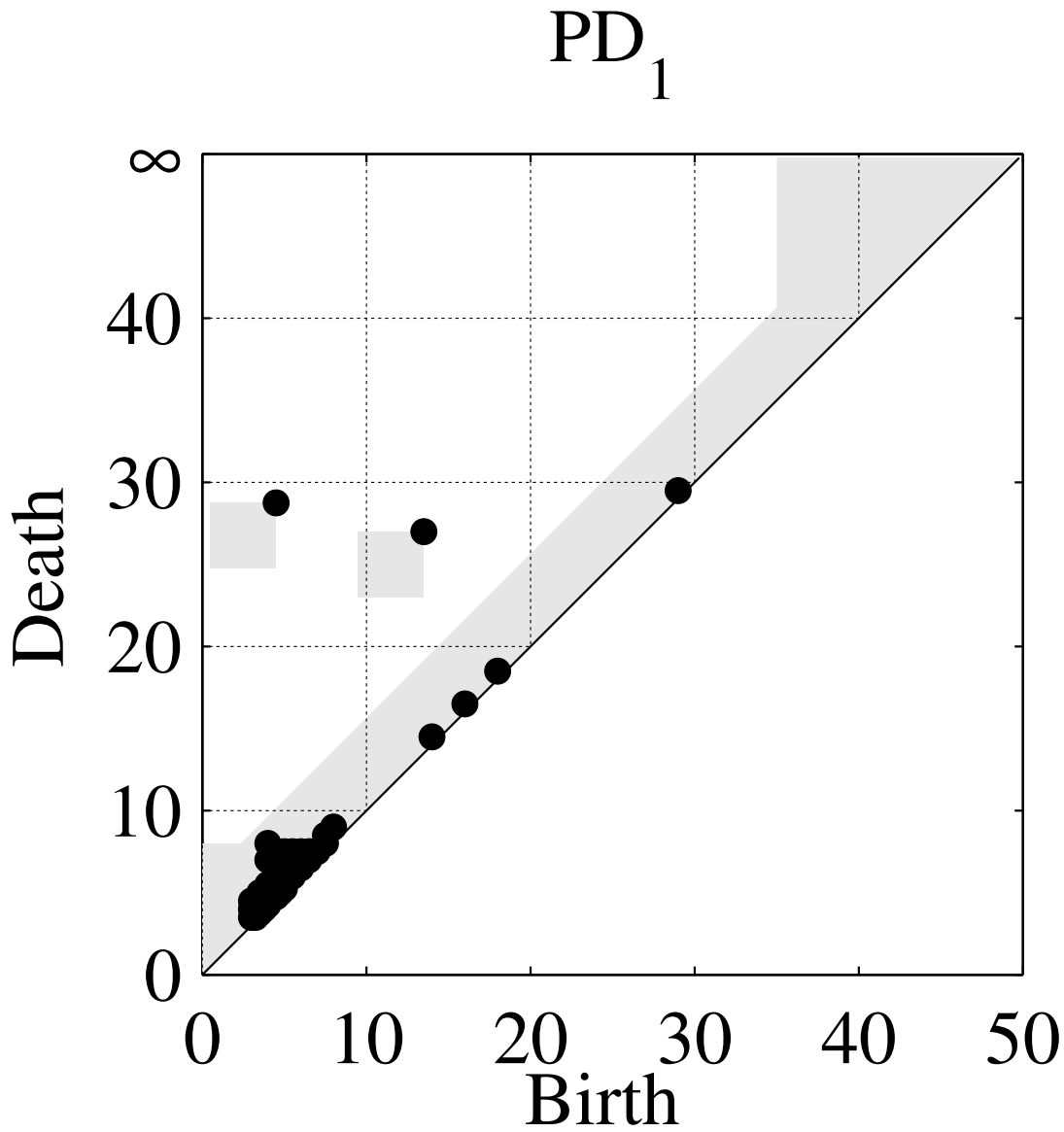
(a) (b) (c)

Figure: (a) Distance matrix for 523 subsampled points and persistence diagrams for Vietoris-Rips complex filtration at (b) H_0 and (c) H_1.

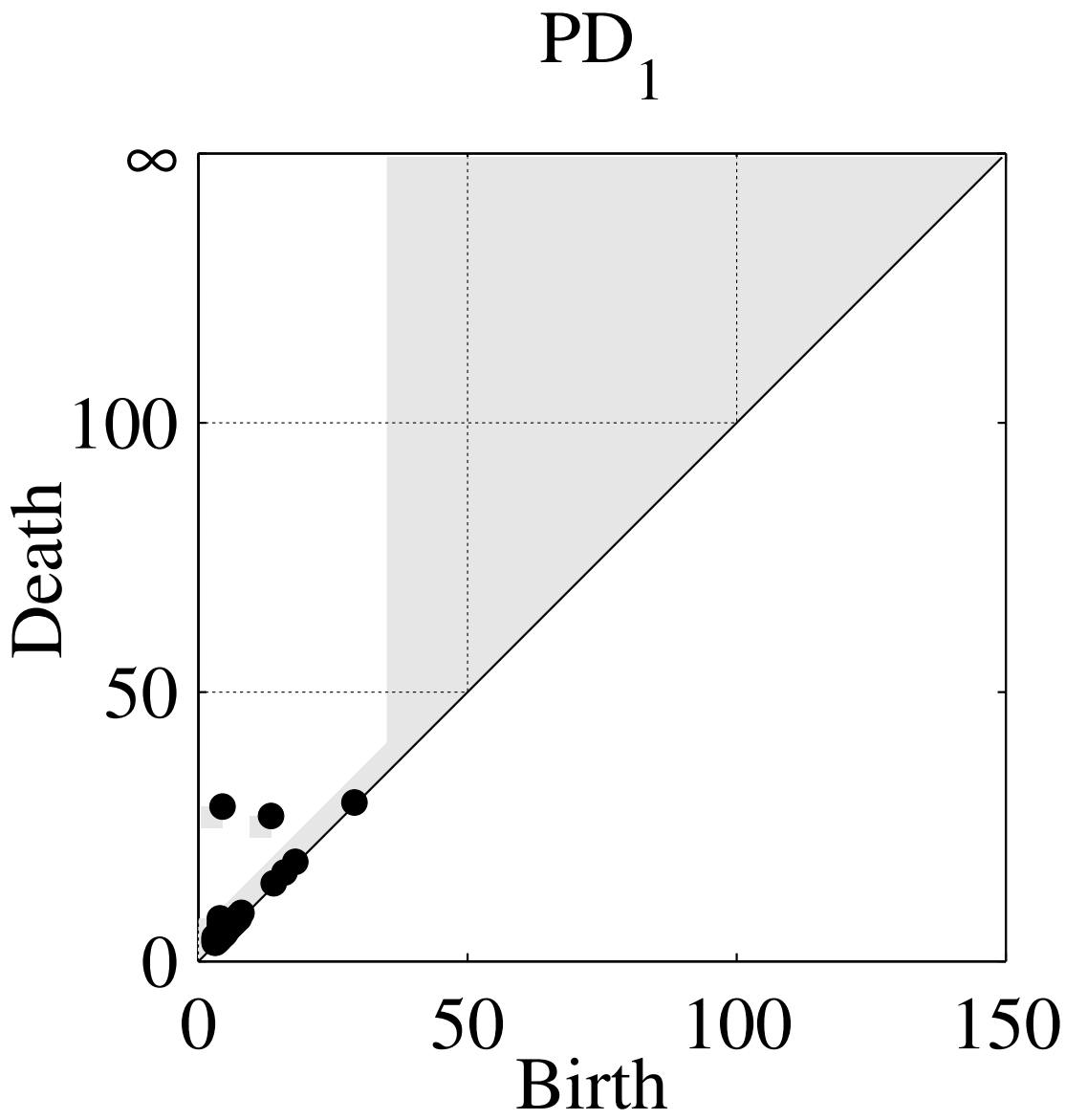
NOTE: $\delta = 4.5$ while max consecutive skip distance is 4.

Could not compute persistence (using Perseus) on the complete Vietoris-Rips Complex filtration with 260Gb of RAM!

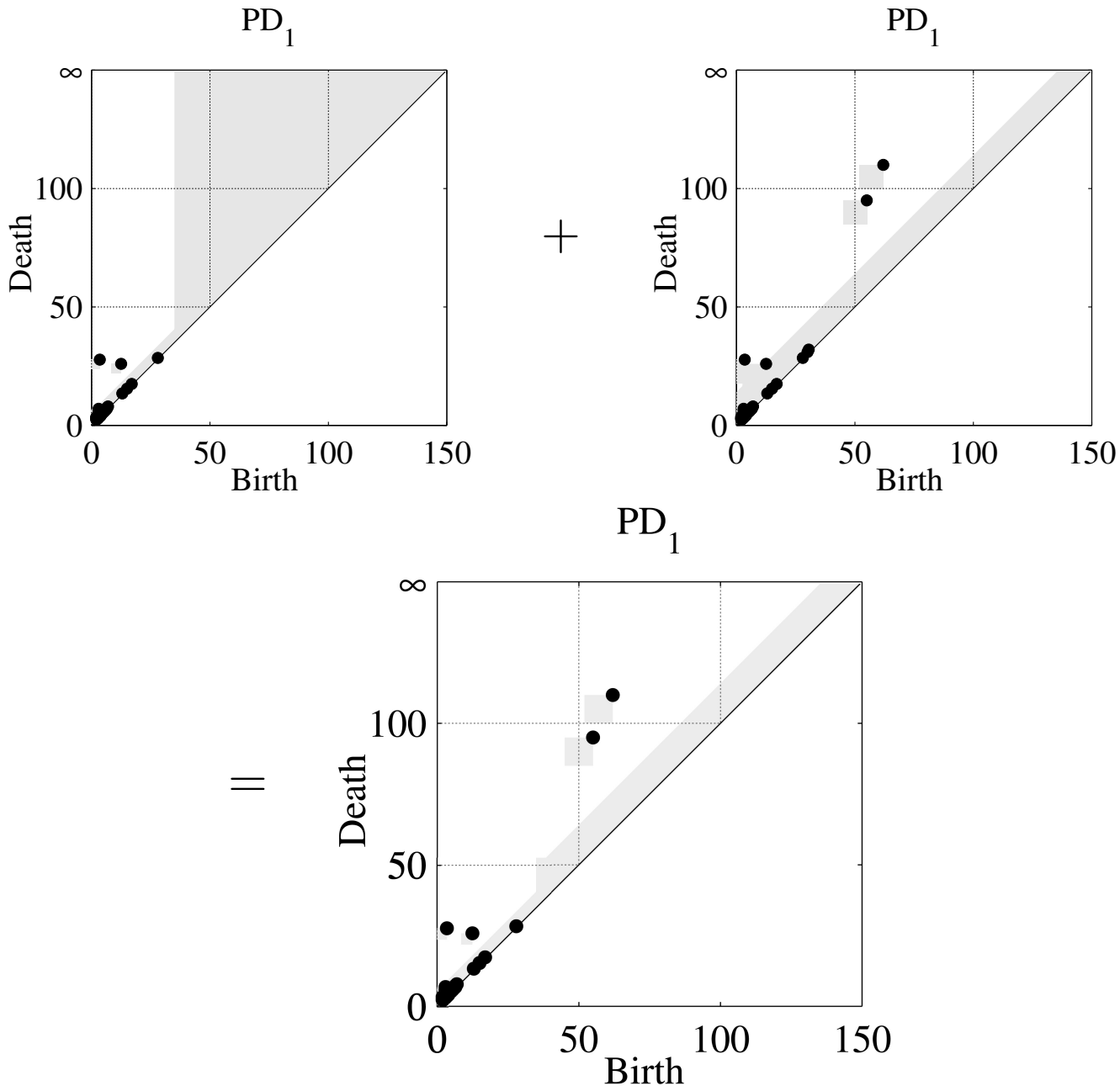
This might be fine if the diameter of the point cloud is 100...



...but what if it is 300?



Is it possible to combine these two persistence diagrams in a natural way?

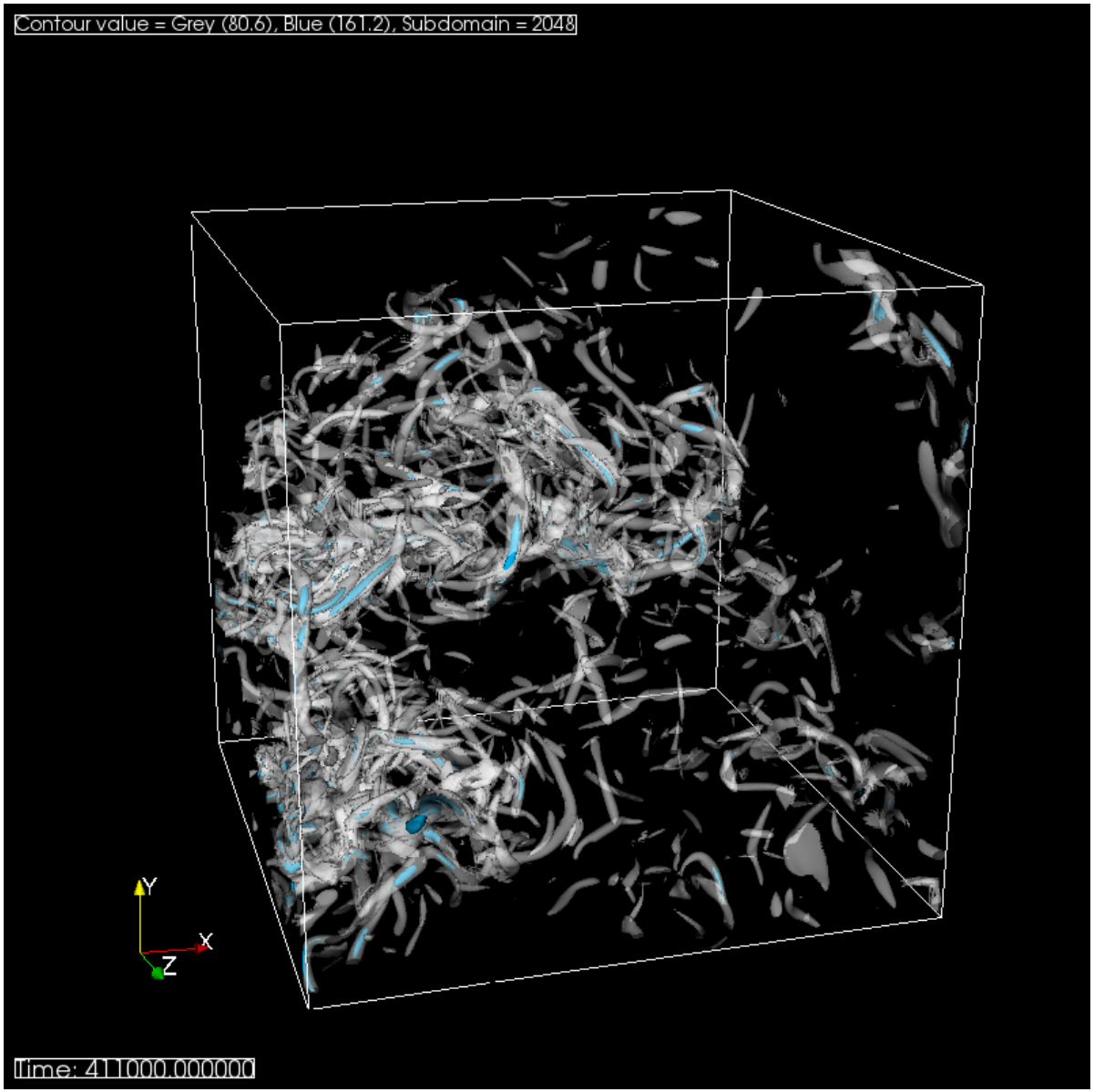

PD1
Death
Birth
0
50
100
150
∞
+
PD1
Death
Birth
0
50
100
150
∞
=
PD1
Death
Birth
0
50
100
150
∞

To be continued...

Problem two: Organizing hard-to-visualize data.

Large-scale simulations of 3D fluid flows: 2048^3 voxels simulating 3D fluid dynamics.

- Scalar fields: magnitude of the vorticity.
- Break into 512 subsets of 256^3 voxels each ($8 \times 8 \times 8$ cubes of simulated fluid flow).
- Want to study the homogeneity of the vorticity fields in each subdomain (after quotienting out symmetries).
- Especially want to compare our method to the use of enstrophy as a data separator (L^2 norm of vorticity).


Contour value = Grey (80.6), Blue (161.2), Subdomain = 2048
Y
X
Z
Time: 411000.000000

General approach:

- Project each 3D vorticity plot to a 3-vector of persistence diagrams (recall: one homology dimension for each spatial dimension).
- Compute the distance matrix D_{ij} using the Landscape L^2 metric (cheaper to compute than d_B or d_{W^p}).
- Use a diffusion map (or other NLDR techniques) to project persistence diagram vectors to $\mathbb{R}^3$.
- Look for organization in the data.

Landscape L^p metric:

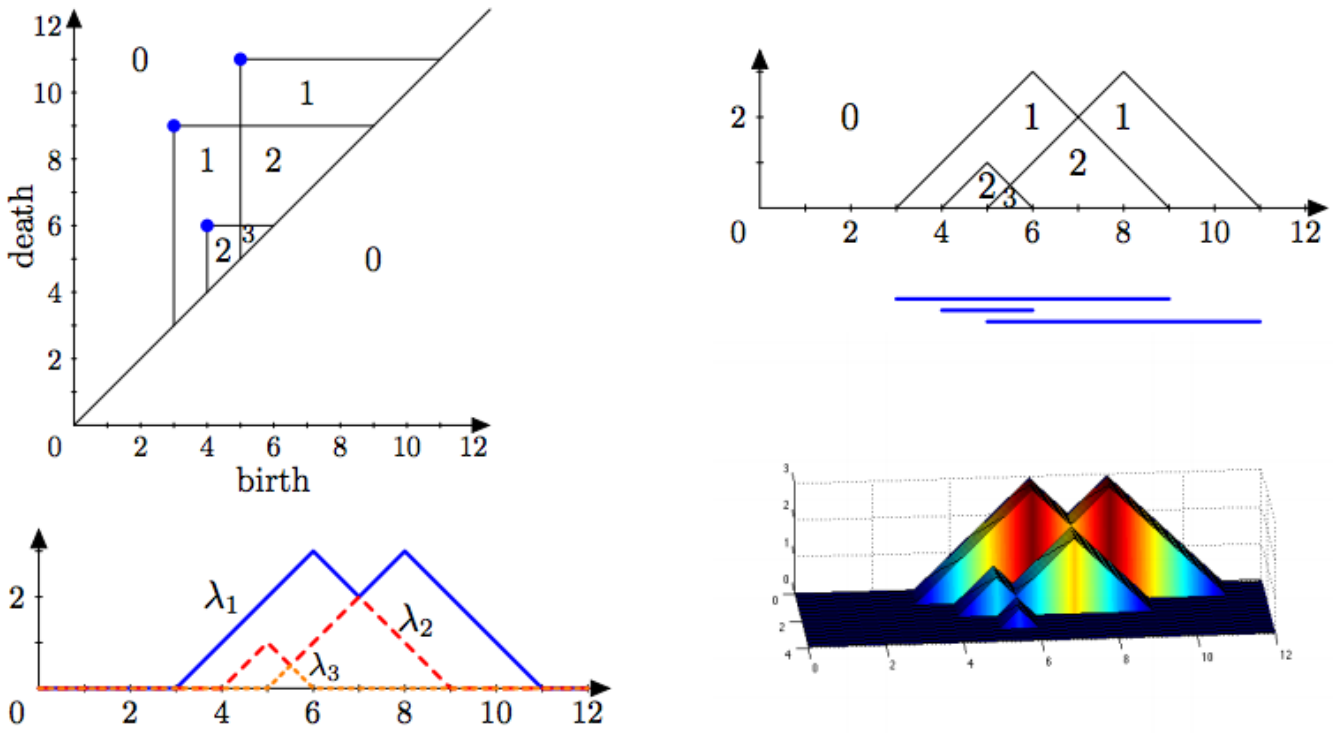


Figure: Figures from P. Bubenick, "Statistical Topological Data Analysis Using Persistence Landscapes," 2015.

The landscape L^p metric is just the regular L^p metric on continuous, real-valued functions.

Diffusion map projection: For more detailed information, visit...

Diffusion maps

Ronald R. Coifman *, Stéphane Lafon [1]

Mathematics Department, Yale University, New Haven, CT 06520, USA

Received 29 October 2004; revised 19 March 2006; accepted 2 April 2006

Available online 19 June 2006

Communicated by the Editors

Abstract

In this paper, we provide a framework based upon diffusion processes for finding meaningful geometric descriptions of data sets. We show that eigenfunctions of Markov matrices can be used to construct coordinates called *diffusion maps* that generate efficient representations of complex geometric structures. The associated family of *diffusion distances*, obtained by iterating the Markov matrix, defines multiscale geometries that prove to be useful in the context of data parametrization and dimensionality reduction. The proposed framework relates the spectral properties of Markov processes to their geometric counterparts and it unifies ideas arising in a variety of contexts such as machine learning, spectral graph theory and eigenmap methods.

Keywords: Diffusion processes; Diffusion metric; Manifold learning; Dimensionality reduction; Eigenmaps; Graph Laplacian

A (very) brief idea:

Construction of the family of diffusions

(1) Fix $\alpha \in \mathbb{R}$ and a rotation-invariant kernel $k_\varepsilon(x, y) = h\left(\frac{\|x-y\|^2}{\varepsilon}\right)$.
(2) Let

$$q_\varepsilon(x) = \int_X k_\varepsilon(x, y) q(y)\, dy$$

and form the new kernel

$$k_\varepsilon^{(\alpha)}(x, y) = \frac{k_\varepsilon(x, y)}{q_\varepsilon^\alpha(x) q_\varepsilon^\alpha(y)}.$$

(3) Apply the weighted graph Laplacian normalization to this kernel by setting

$$d_\varepsilon^{(\alpha)}(x) = \int_X k_\varepsilon^{(\alpha)}(x, y) q(y)\, dy$$

and by defining the anisotropic transition kernel

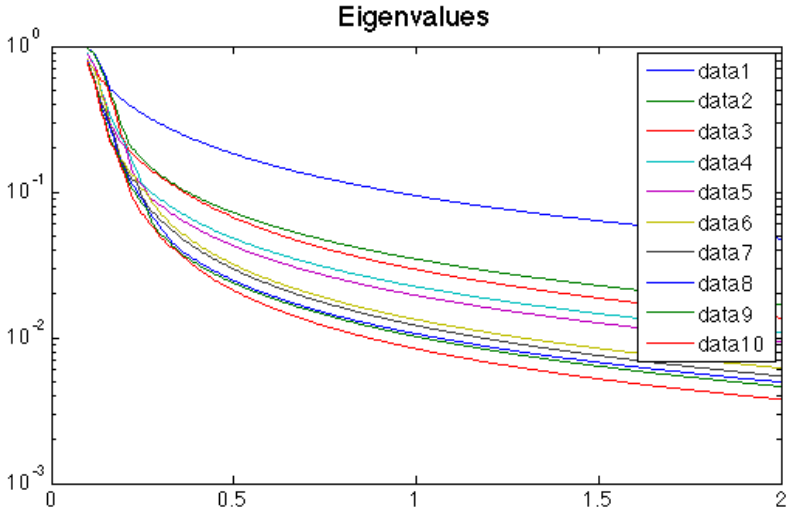
$$p_{\varepsilon,\alpha}(x, y) = \frac{k_\varepsilon^{(\alpha)}(x, y)}{d_\varepsilon^{(\alpha)}(x)}.$$

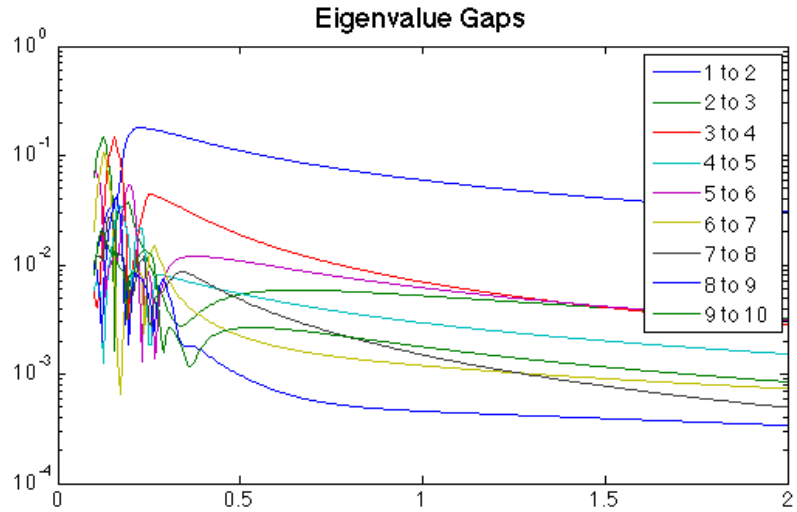
- Compute the eigenvalues/eigenvectors of the matrix P_{ij} using $\alpha = 1/2$ and a choice for ε as a tuning parameter.
- $1 = \lambda_0 \geq \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \cdots$
- Multiply the eigenvectors by their corresponding eigenvalues and plot the coordinates of the first $i = 1, 2, 3$ vectors.

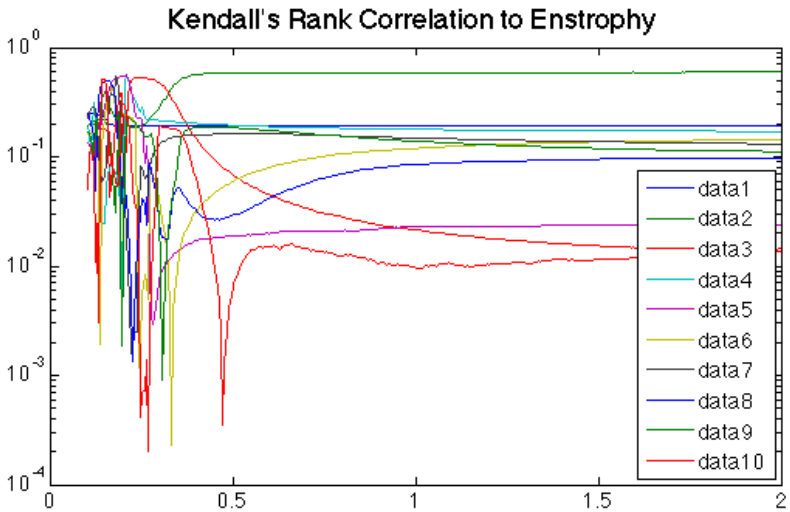
How to choose a value for ε in the diffusion projection?

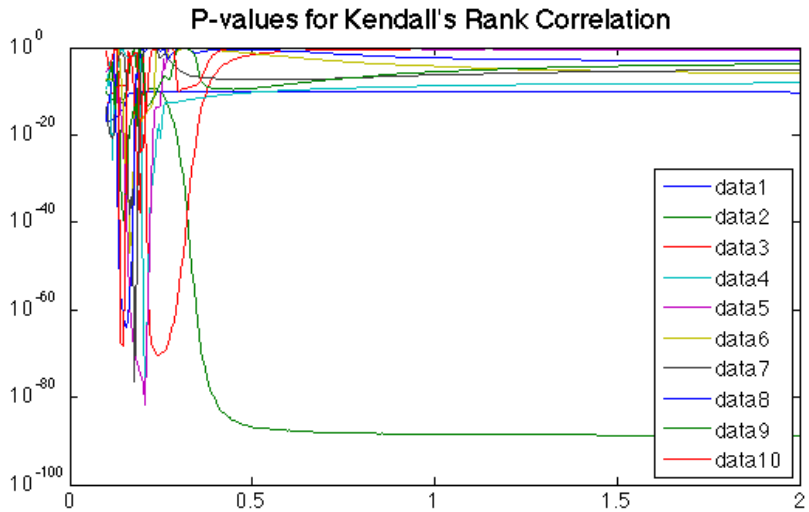
- Plot the eigenvalues as a function of ε
- Plot correlations of eigenvector coefficients against known measurements for the dataset as a function of ε
- Choose a value for your analysis

Our correlation: Enstrophy, the L^2 norm of the vorticity field of the datapoint.



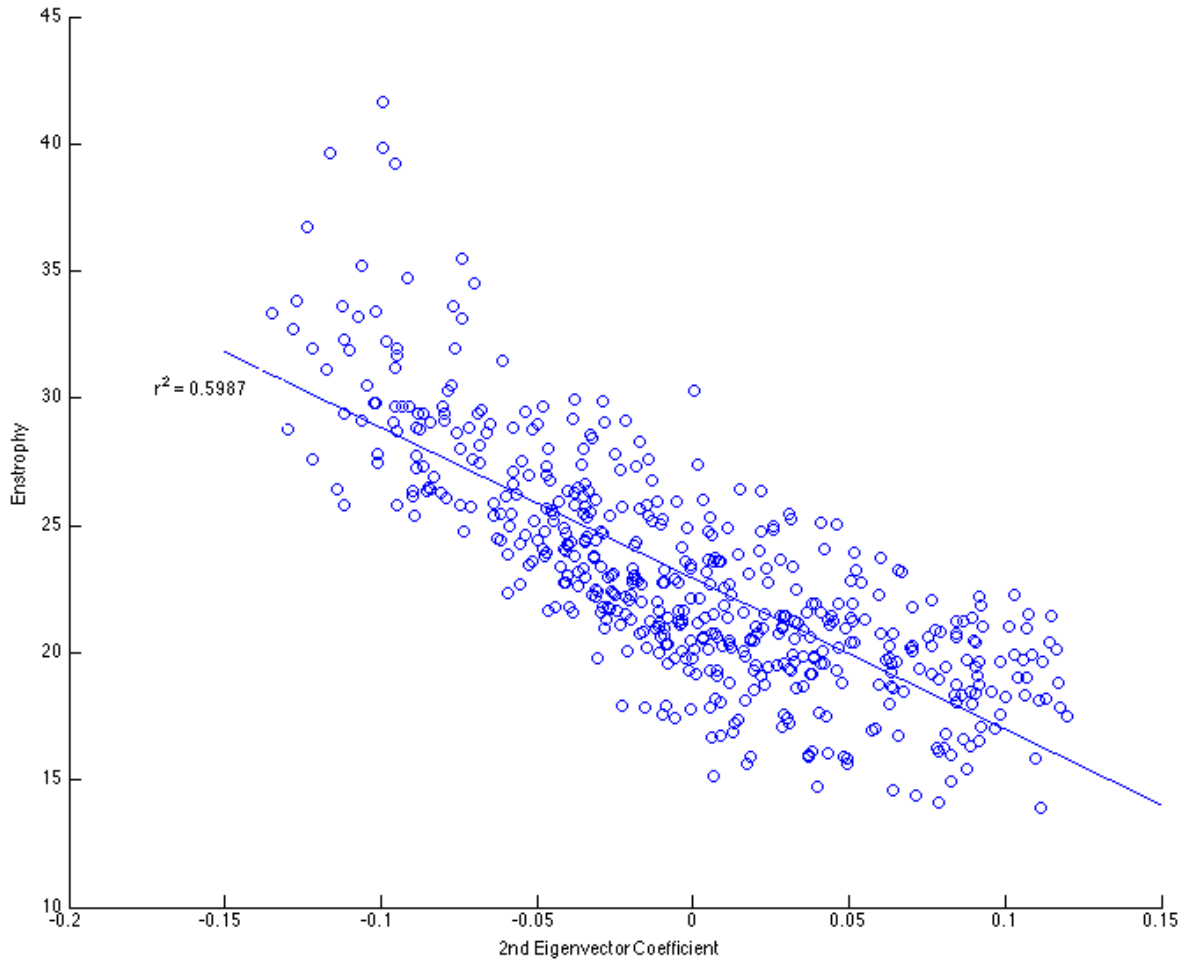




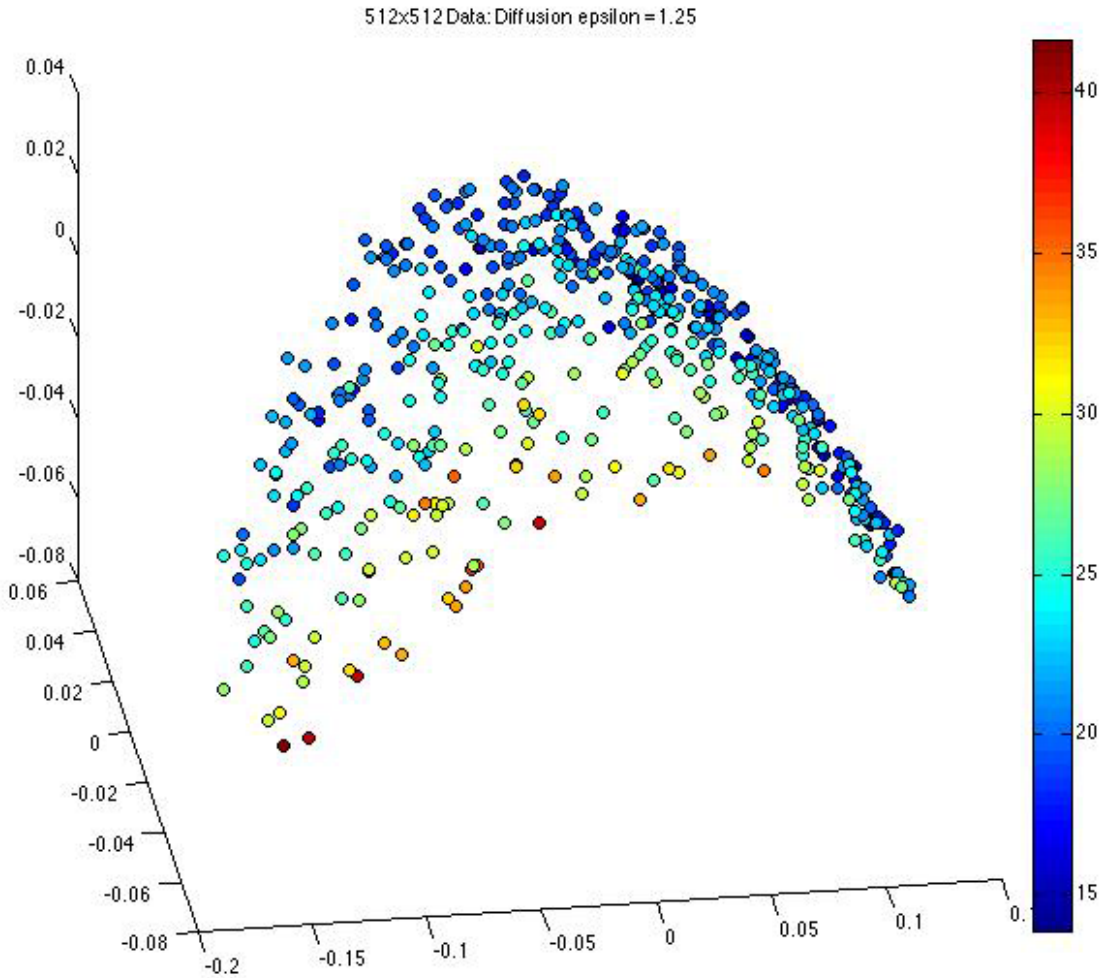


Our choice: $\varepsilon = 0.62 = \text{mean}(D_{ij})$.

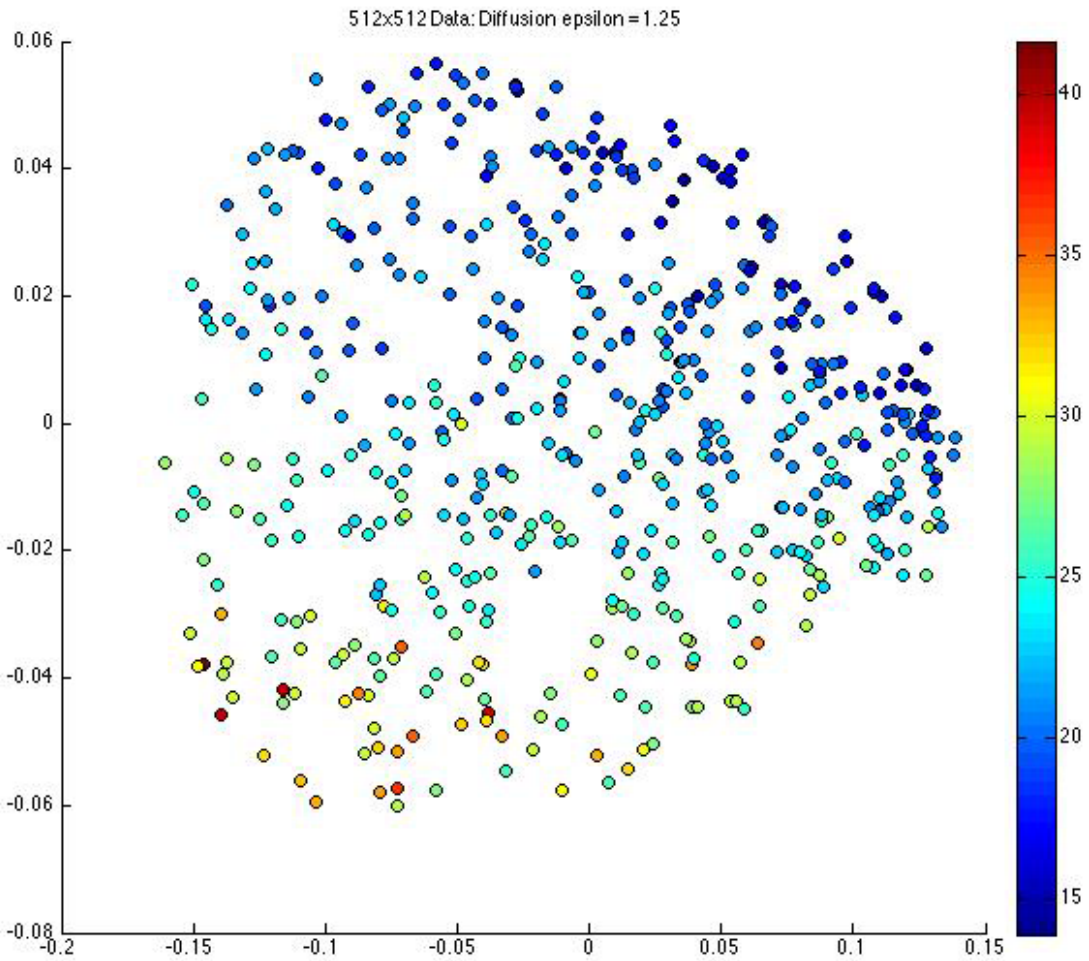
Correlation of coordinates of 2^{nd} eigenvector with enstrophy:



Plot the coordinates of the first three eigenvectors:

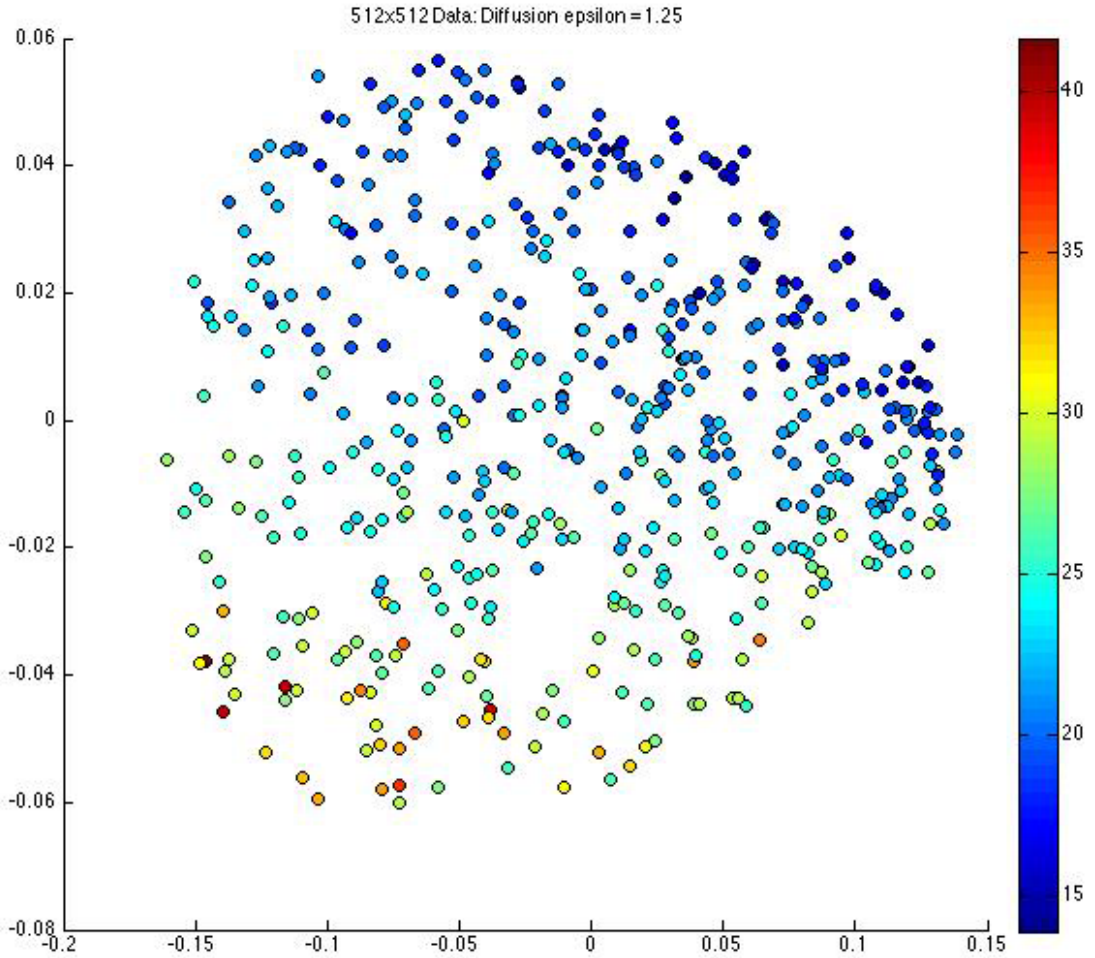


Plot the coordinates of the first two eigenvectors:



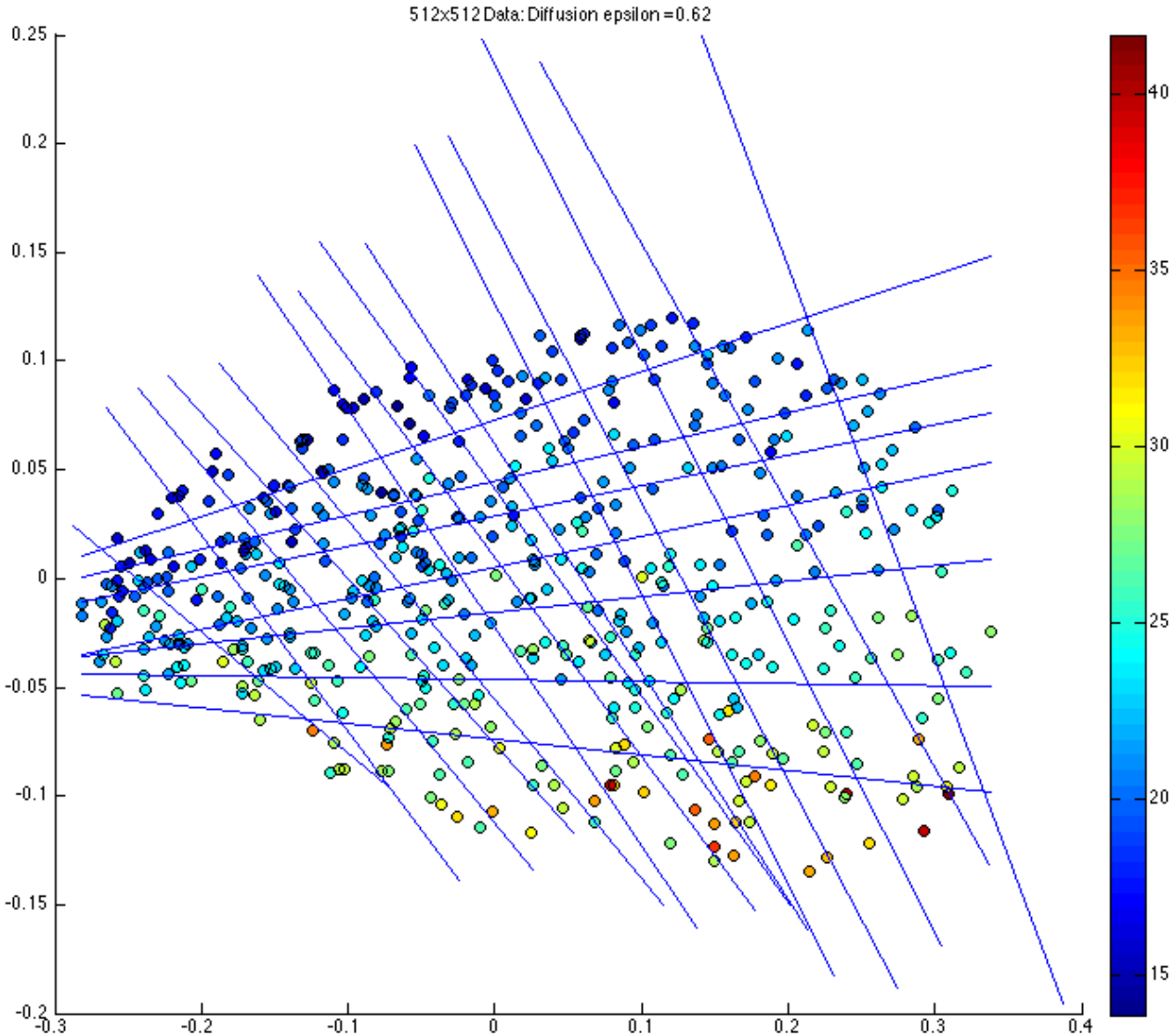
What is behind the organization of the data
along the first eigenvector, orthogonal to enstrophy?

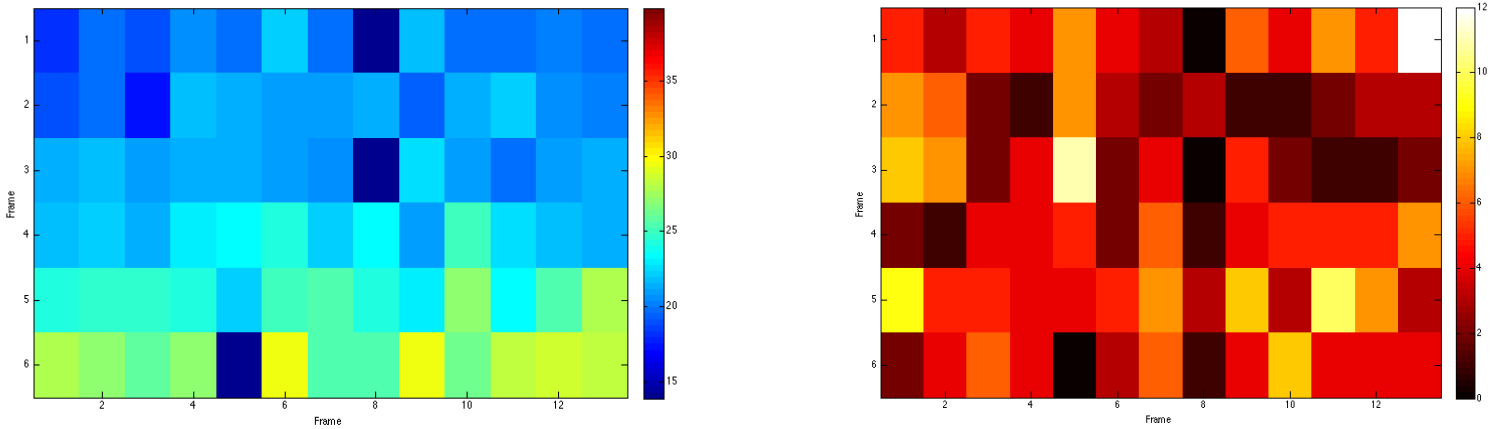
Challenge: How do we coordinatize the data in the $2D$ projection in a way that the x-axis respects the enstrophy stratification and the y-axis is orthogonal to it?



Quantiles and SVD: An approach to superimposing a grid structure on a (roughly) uniform distribution of points in a region.

- Sort data points into 7 quantiles with respect to enstrophy.
- Use SVD to generate lines of best fit within each enstrophy quantile (horizontal grid lines).
- Project each data point within the quantile to the line of best fit and sort points into 15 quantiles.
- Use SVD to generate lines of best fit within each orthogonal quantile (vertical grid lines).
- Use the interior grid lines to find "maximal" grid for the data.


512x512 Data: Diffusion epsilon = 0.62
0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4
40
35
30
25
20
15



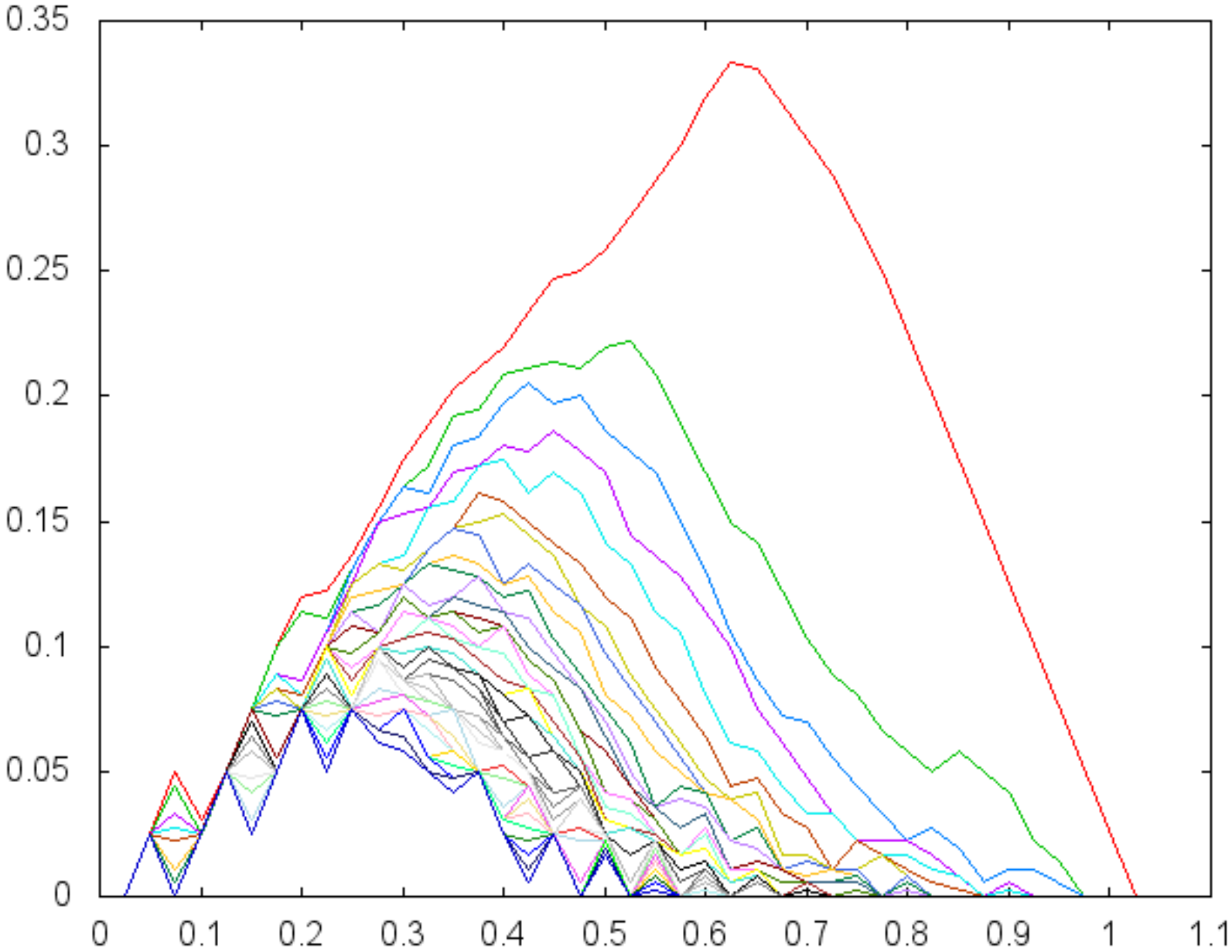
(a) (b)

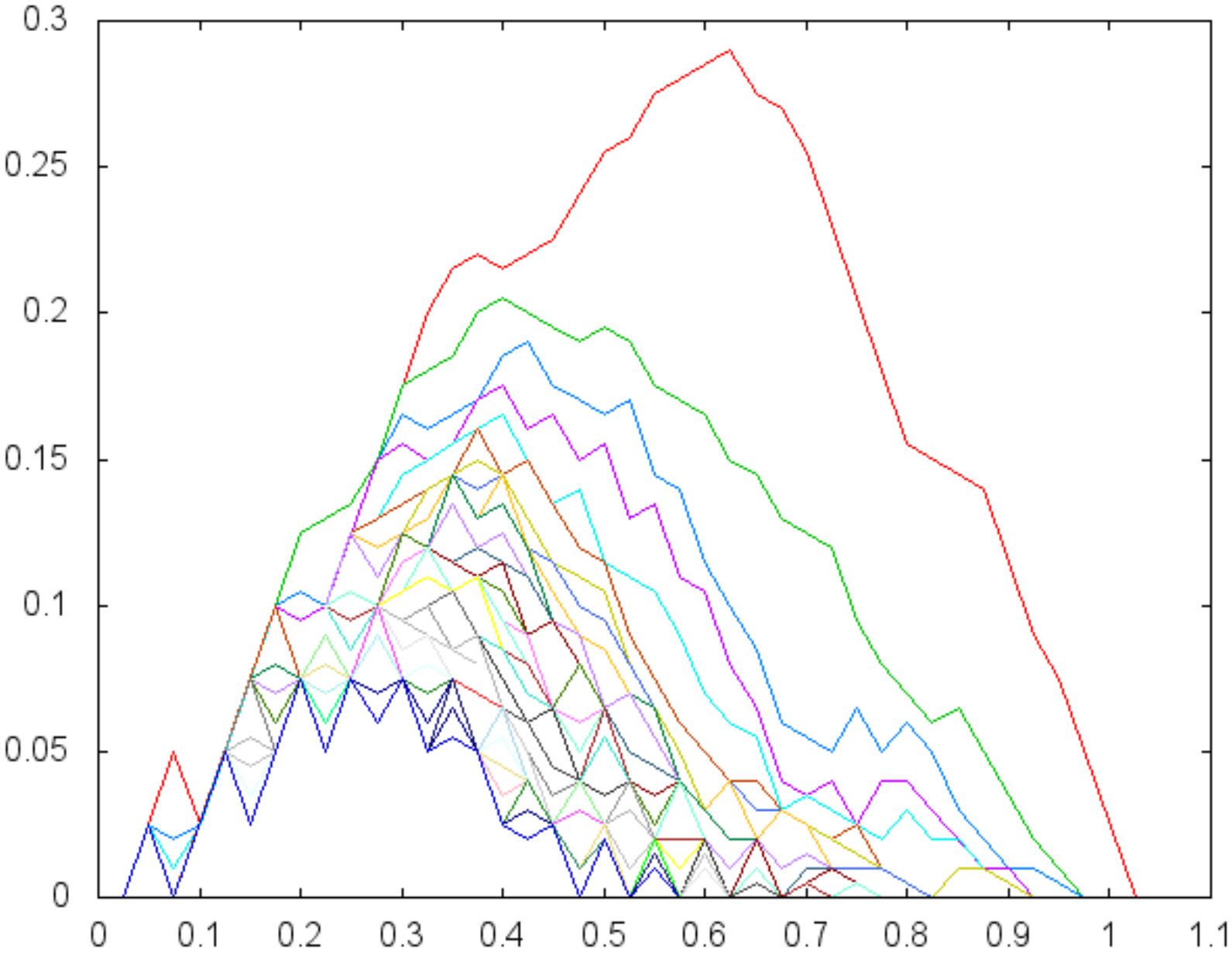
Figure: (a) Average enstrophy (0-40) and (b) number of points (0-12) for each grid box.

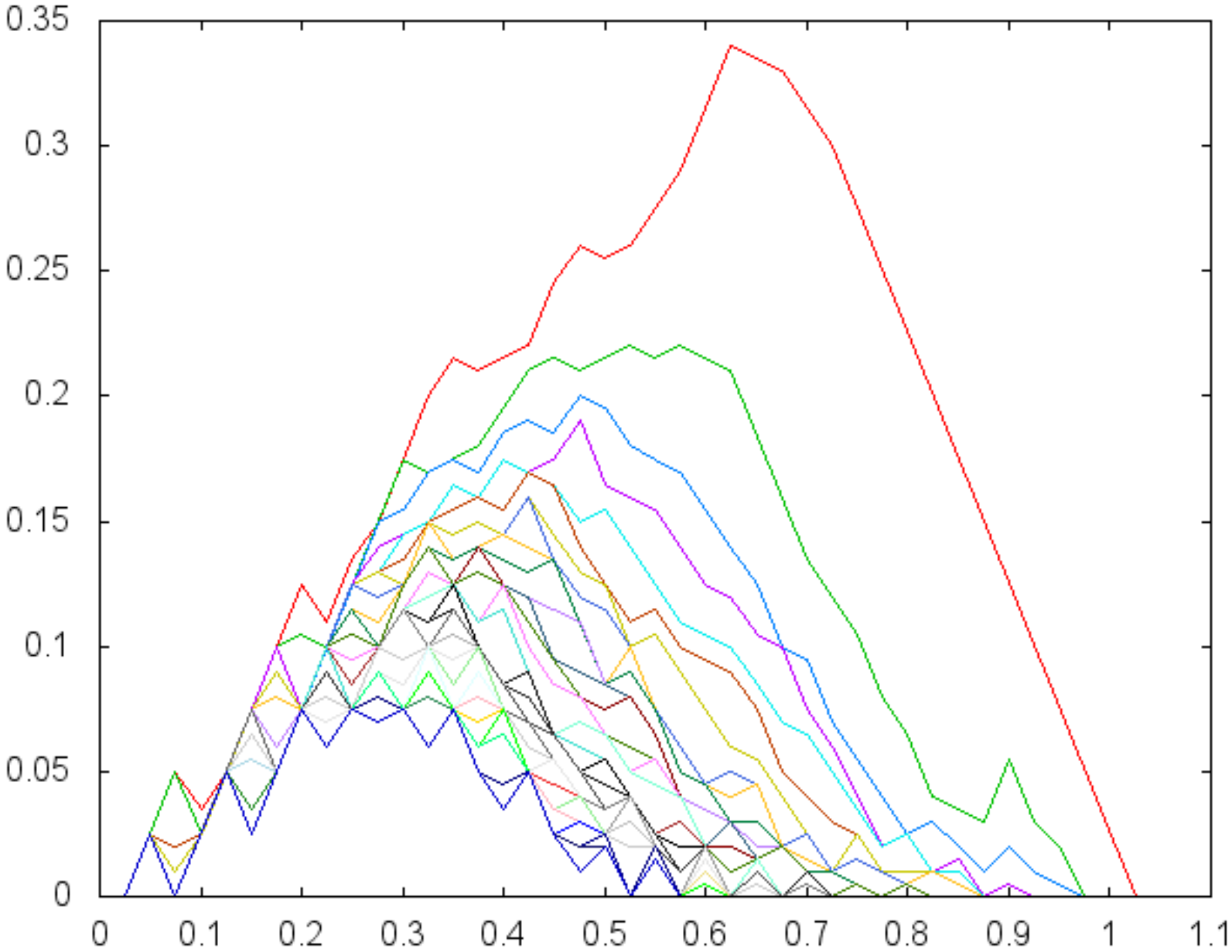
Using landscape diagrams to compute average behavior:

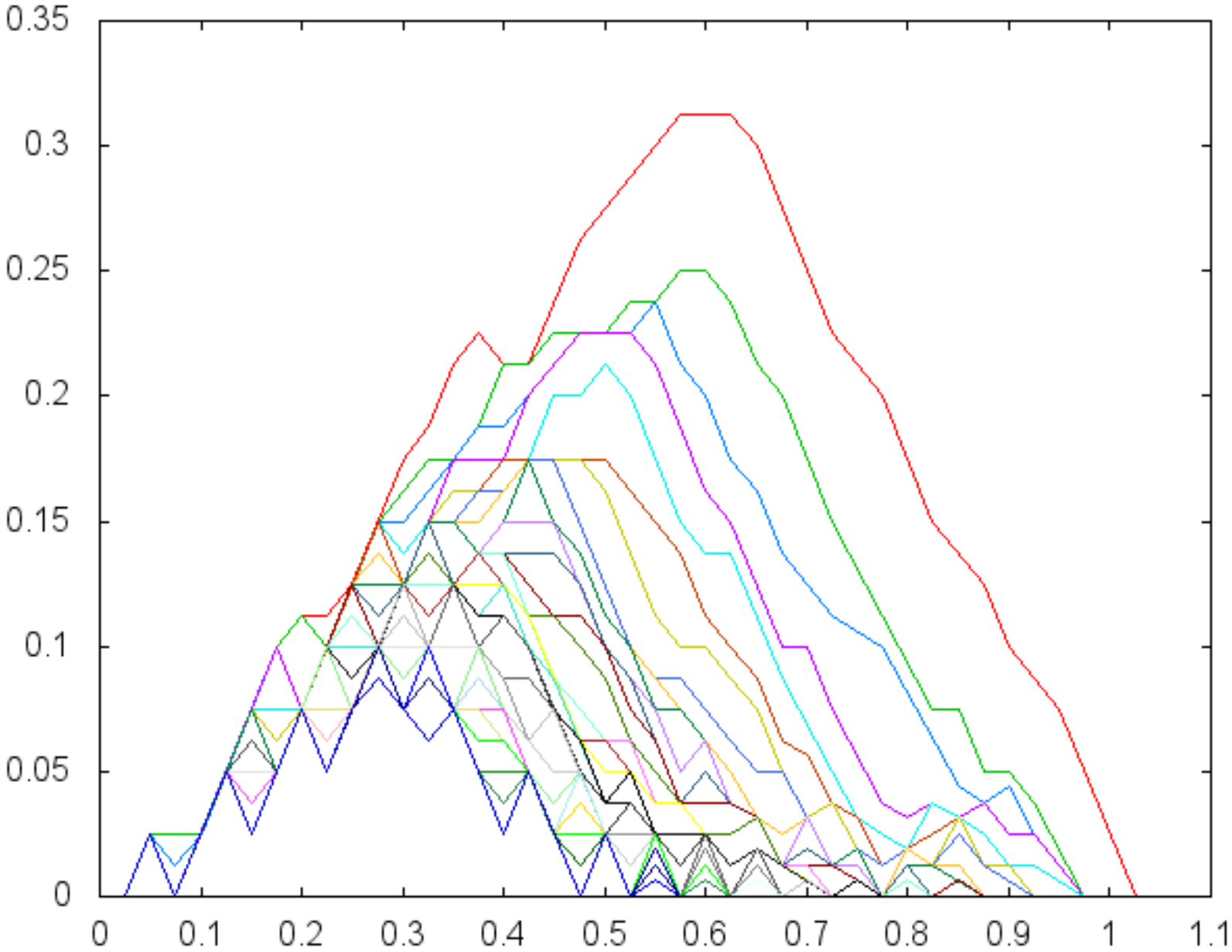
For each box in the grid...

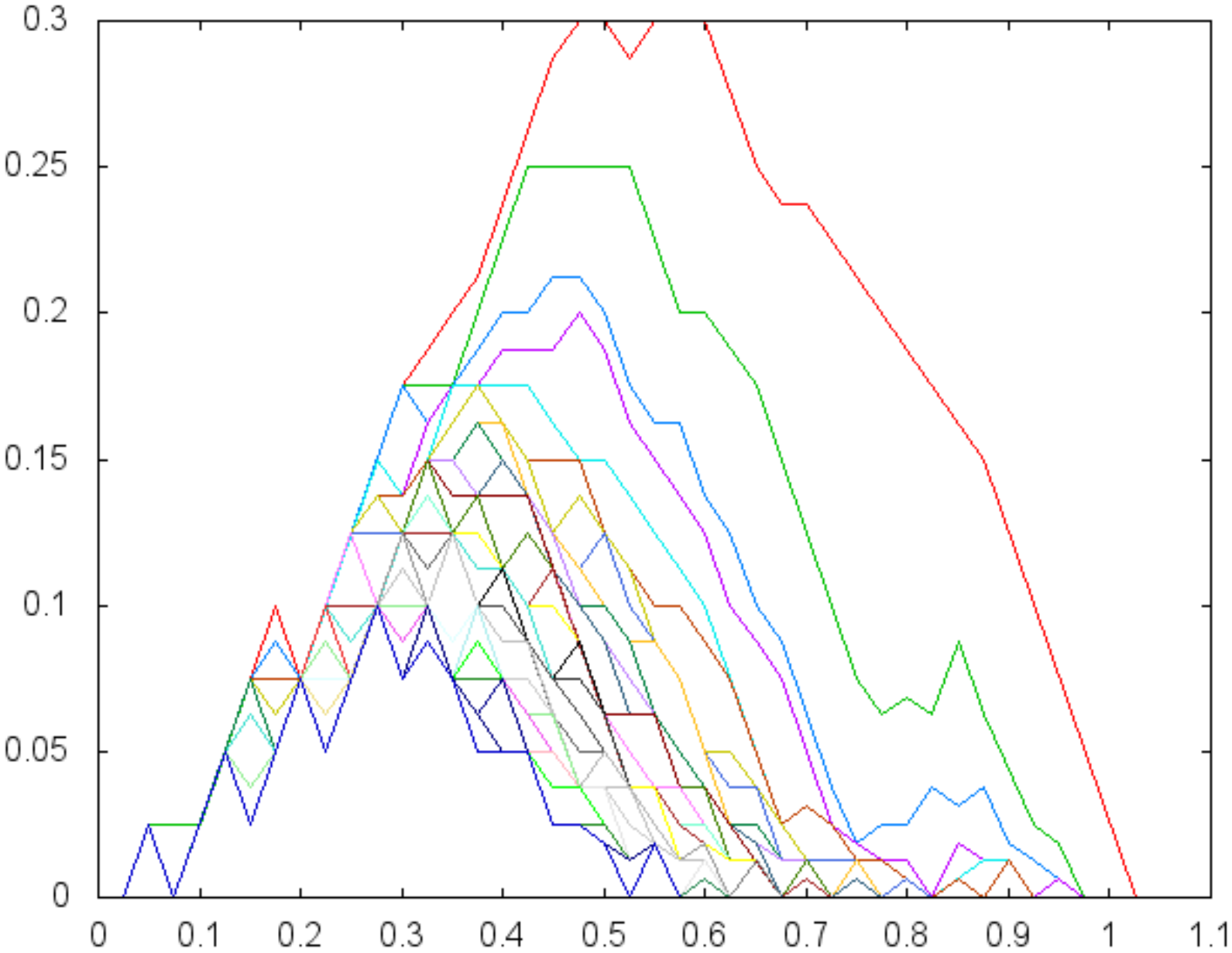
- Compute the average landscape diagram for each collection of landscape diagrams in the grid box
- Display the average landscape diagrams along each homology dimension
- Look for patterns, conjecture useful statistics, and run correlations on these statistics

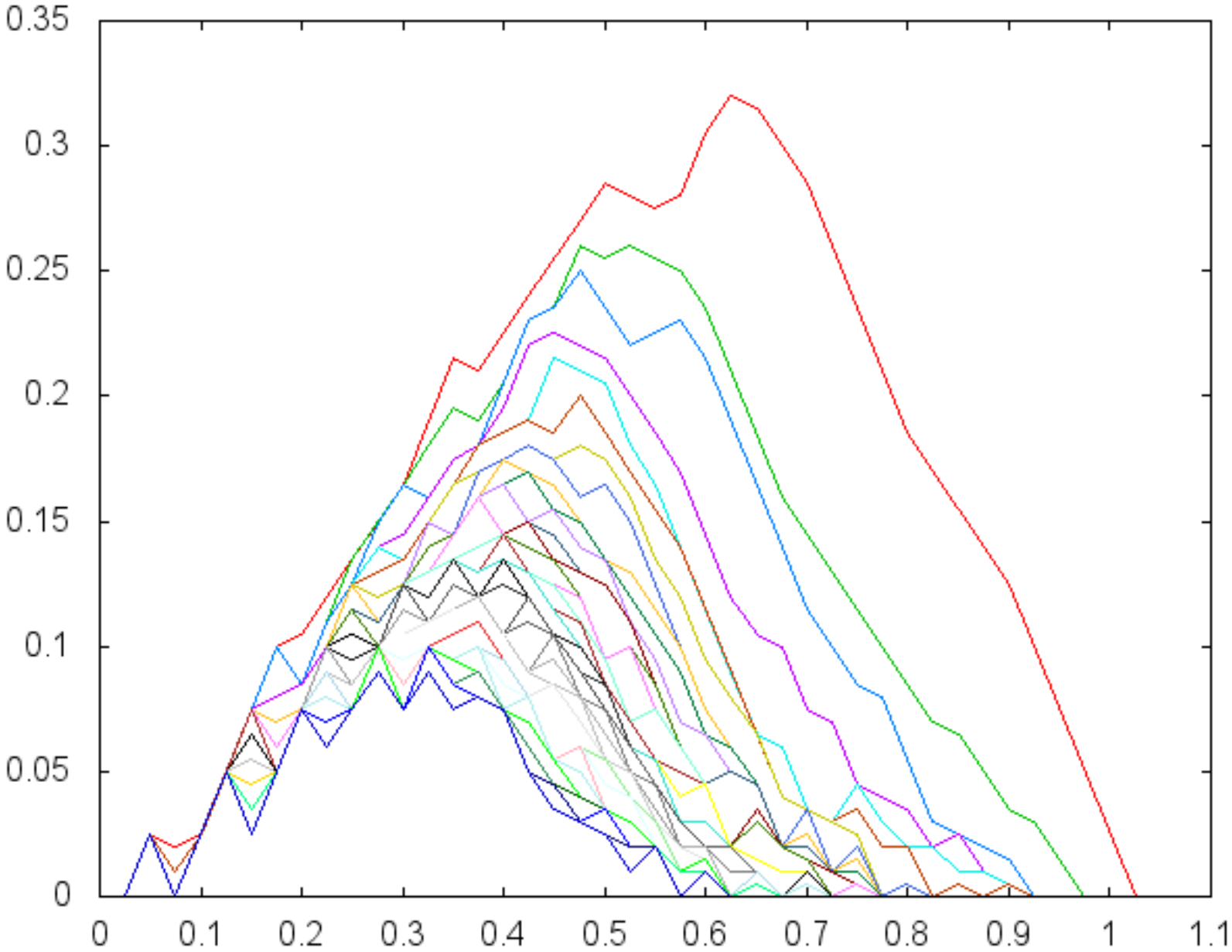

0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1

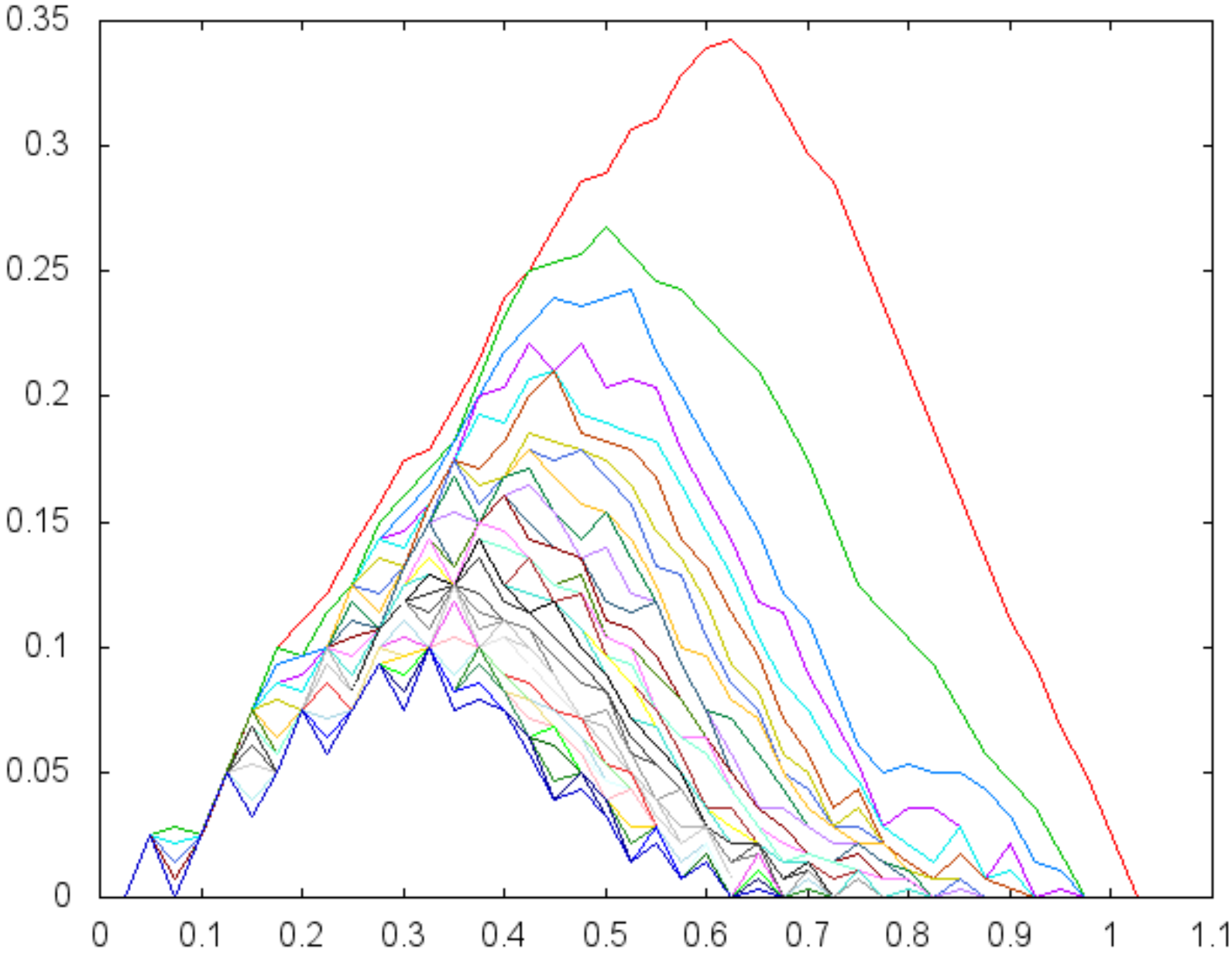

0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1

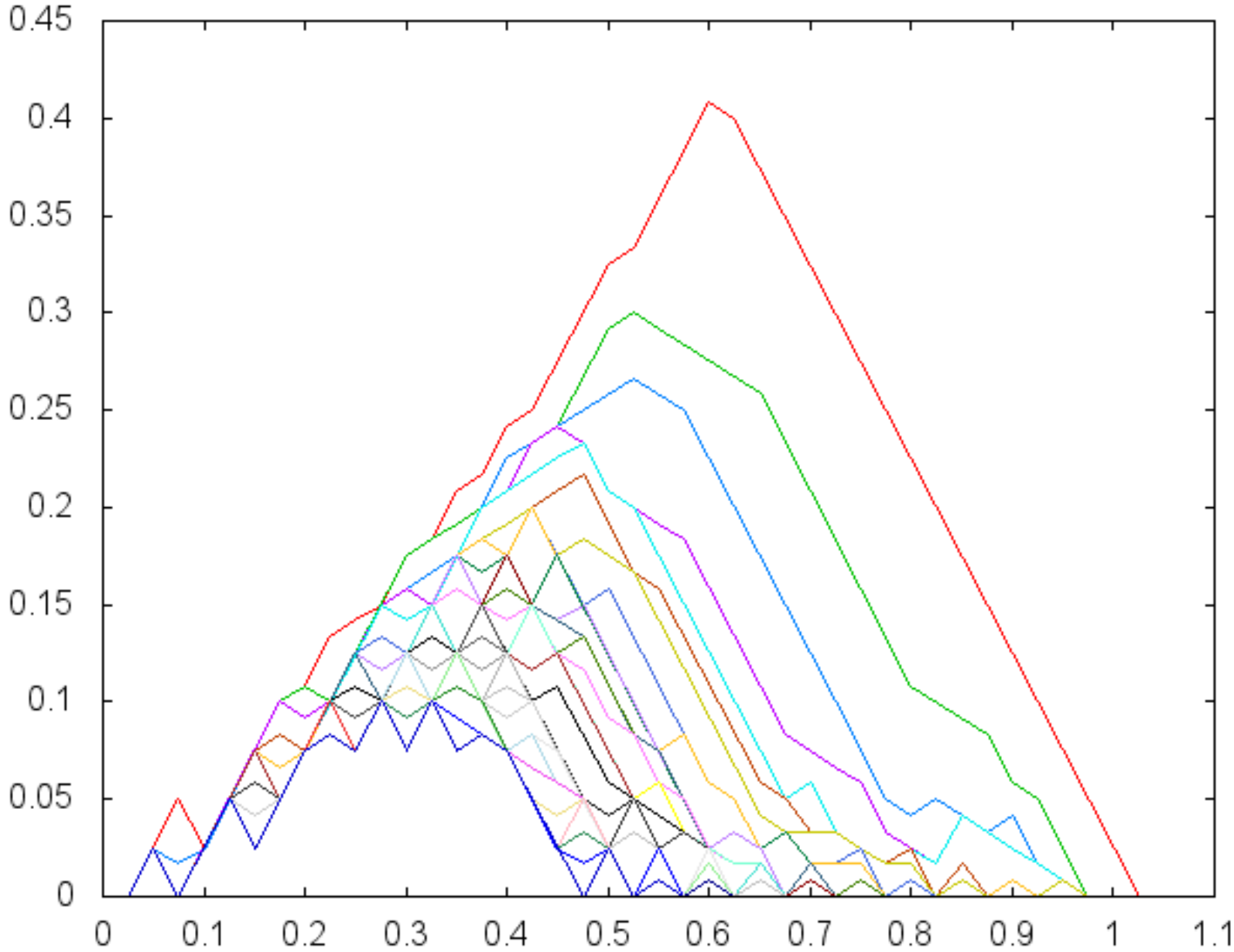

0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1

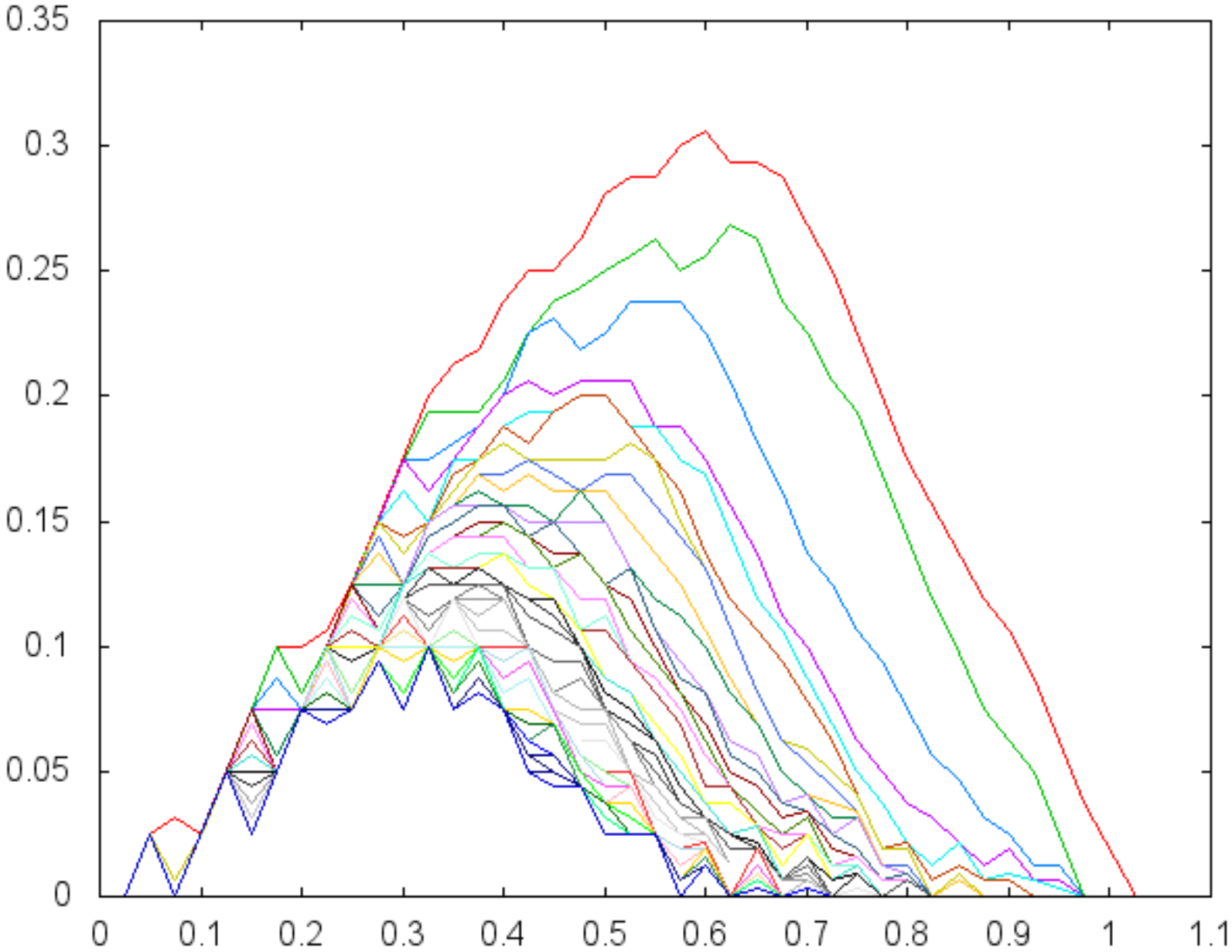

0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1

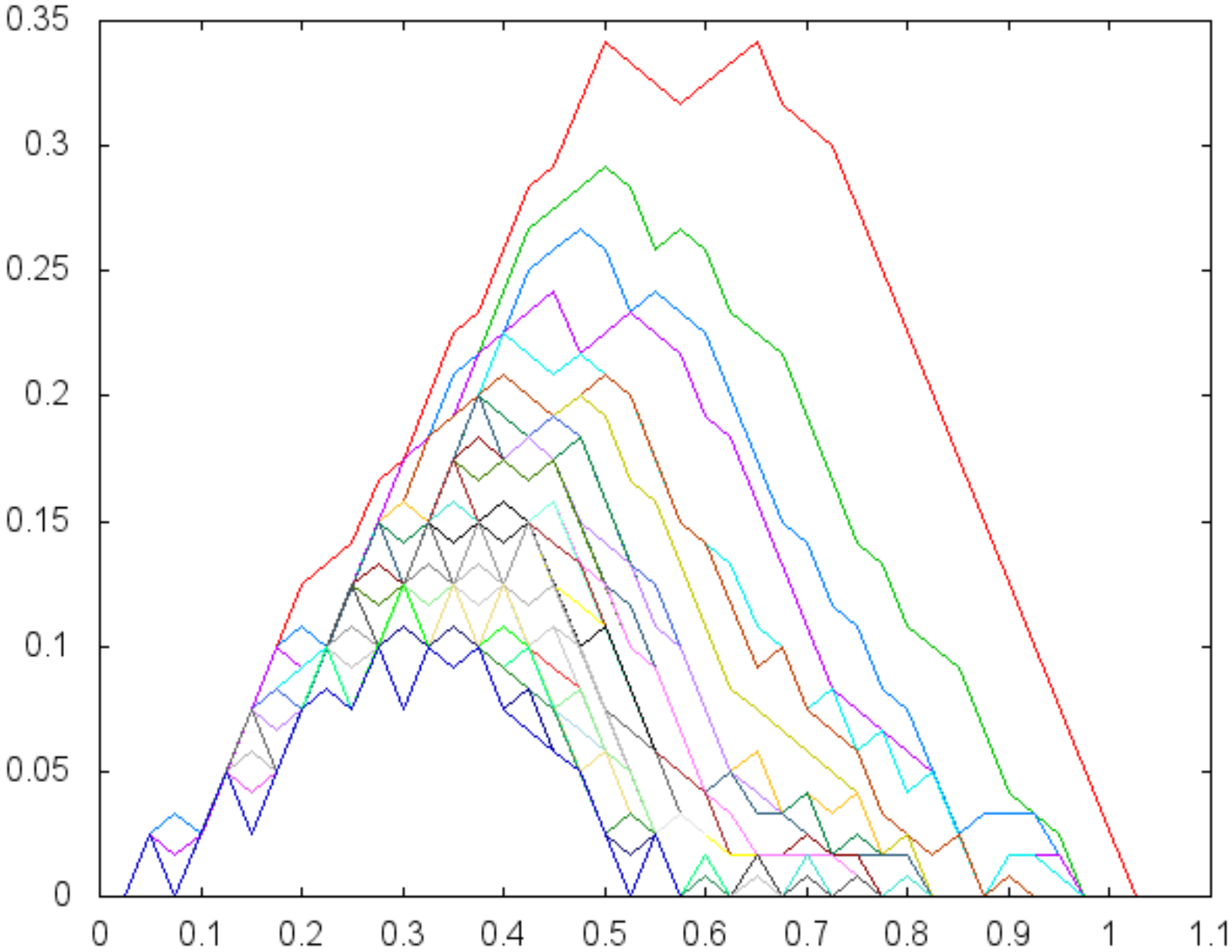

0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1

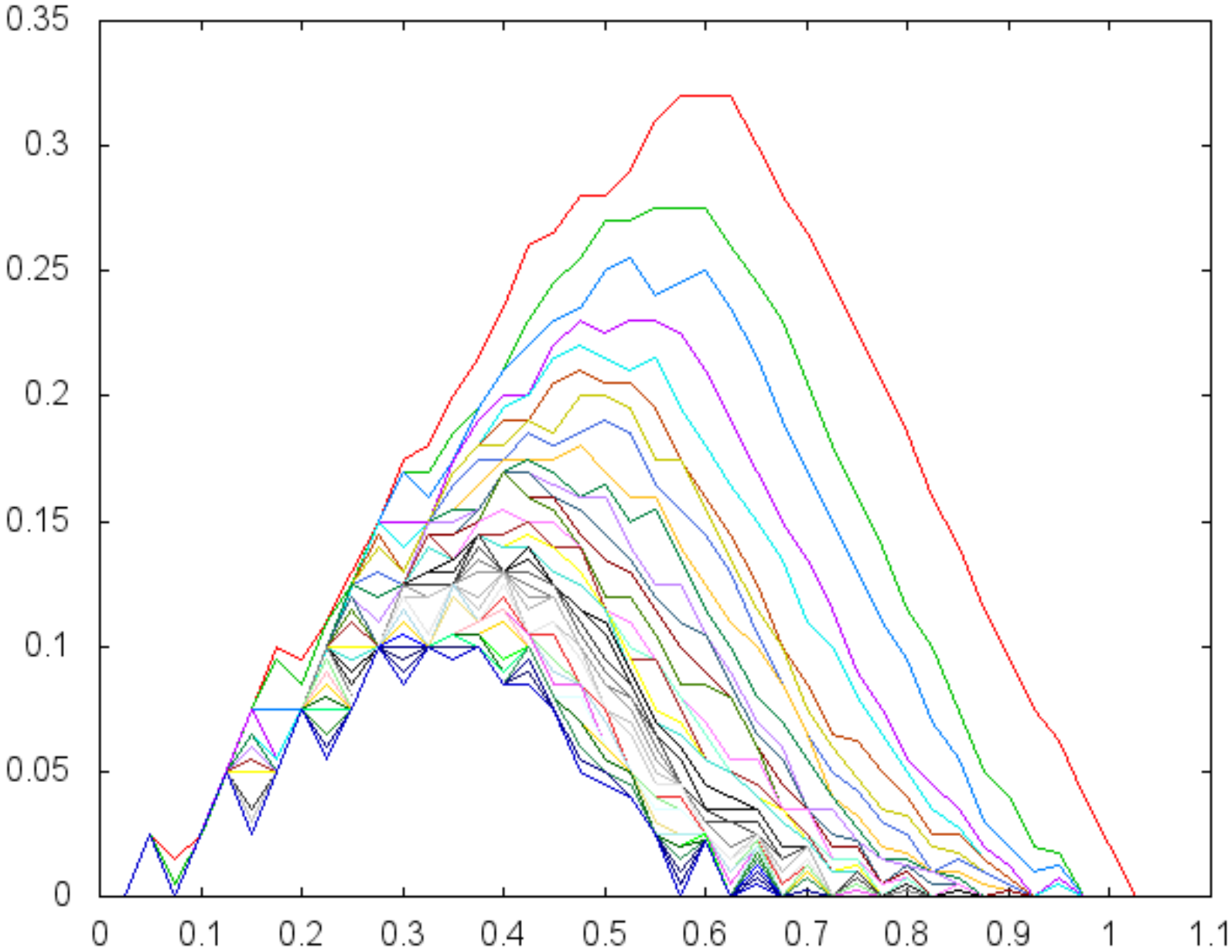

0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1

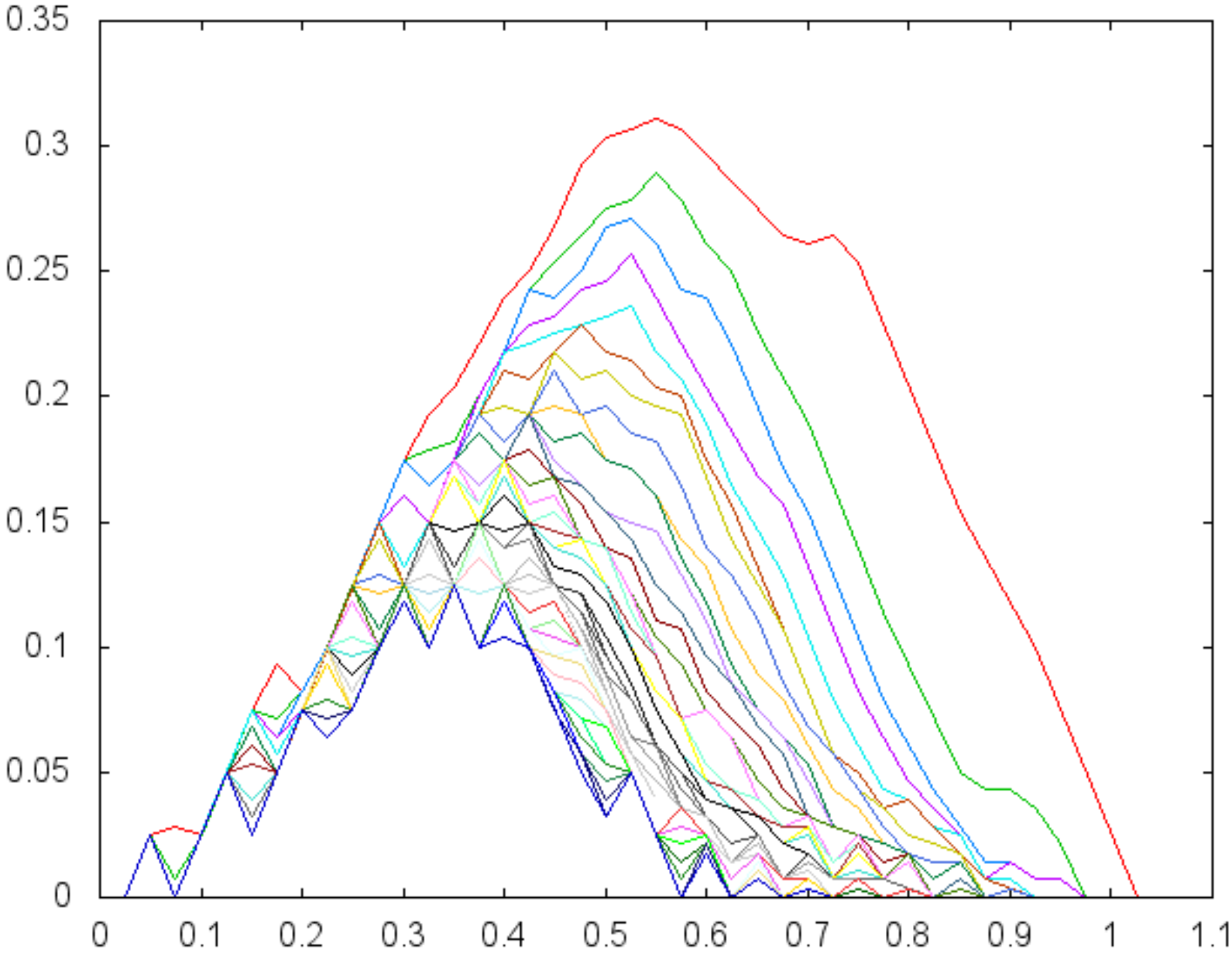

0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1

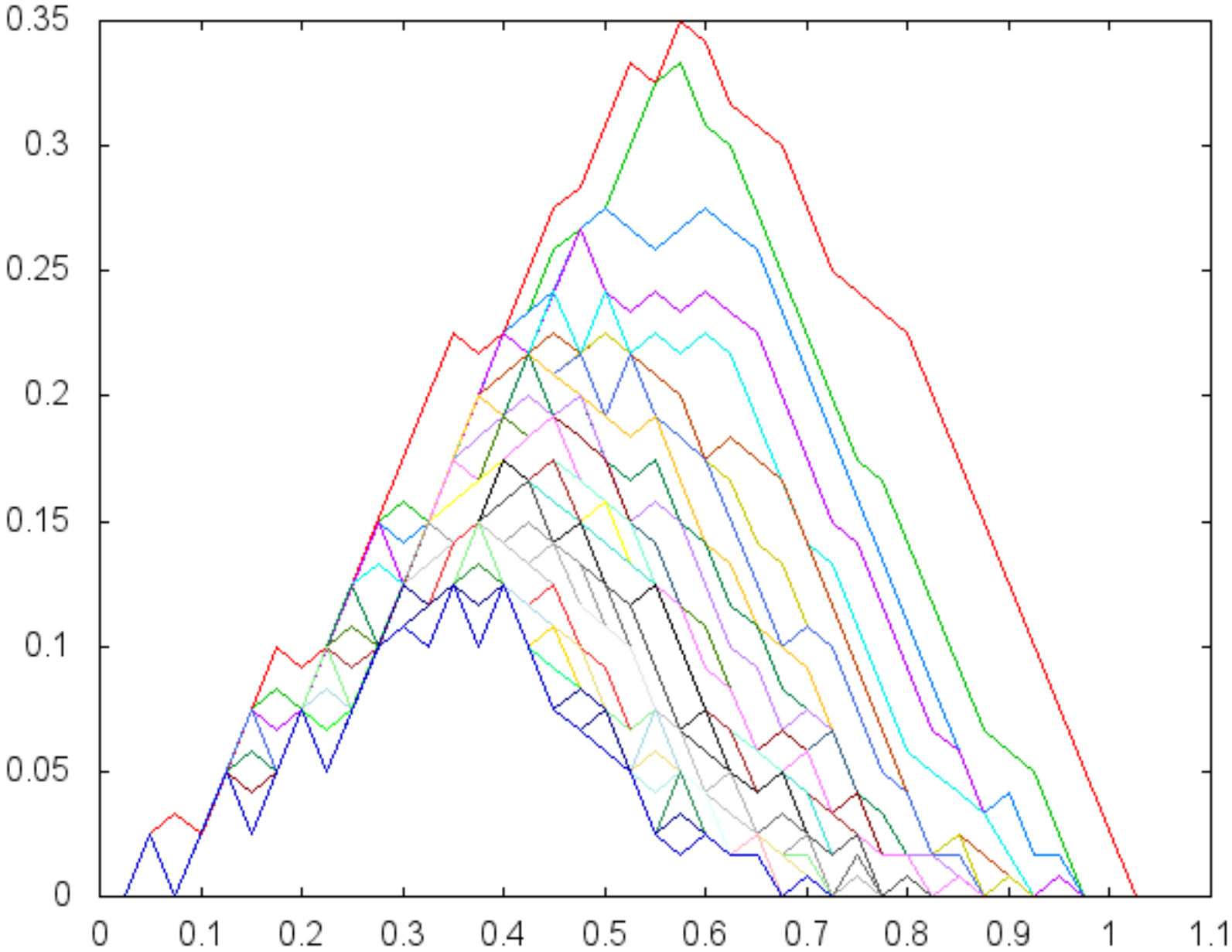

0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1


0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1


0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1


0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1


0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1


0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1

Conjecturing a statistic:

- The behavior of the first m landscape functions are (roughly) a function of the first n points as ordered by lifespan, descending.
- The average lifespan of the top n points will change as the distribution of lifespans will change.

Q: Do we see a correlation with the coefficients of the first eigenvector and the average lifespan of the top n points?

Q: How do we choose a value for n?

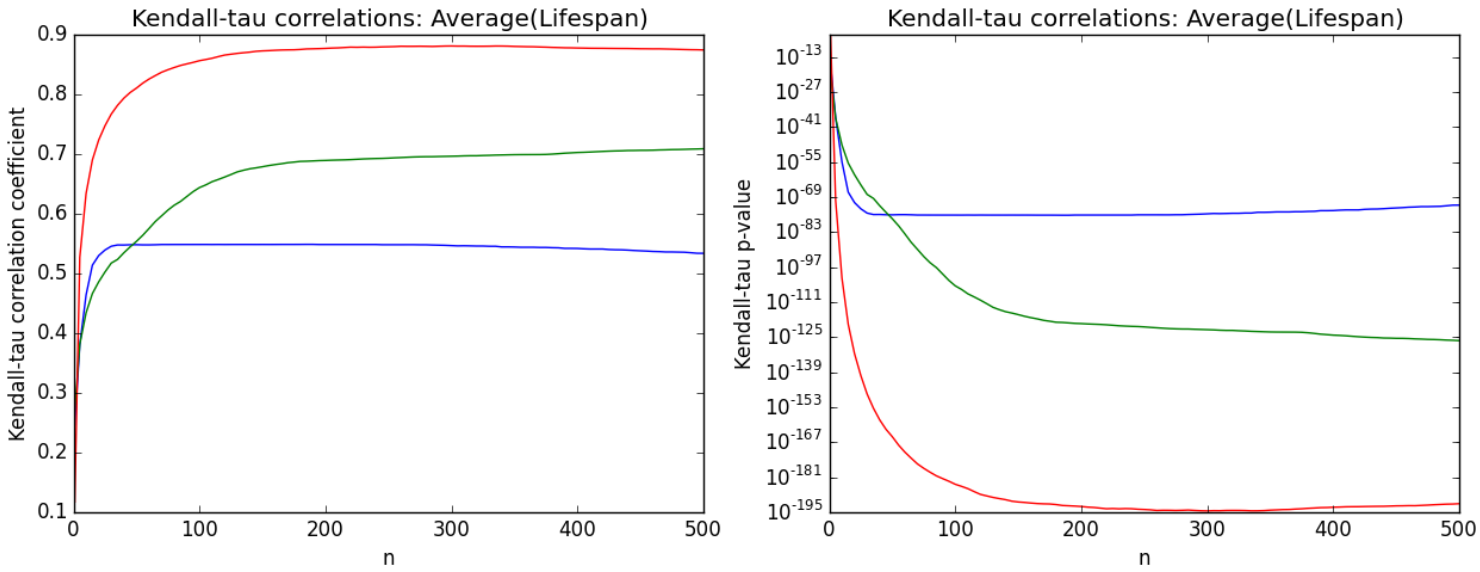
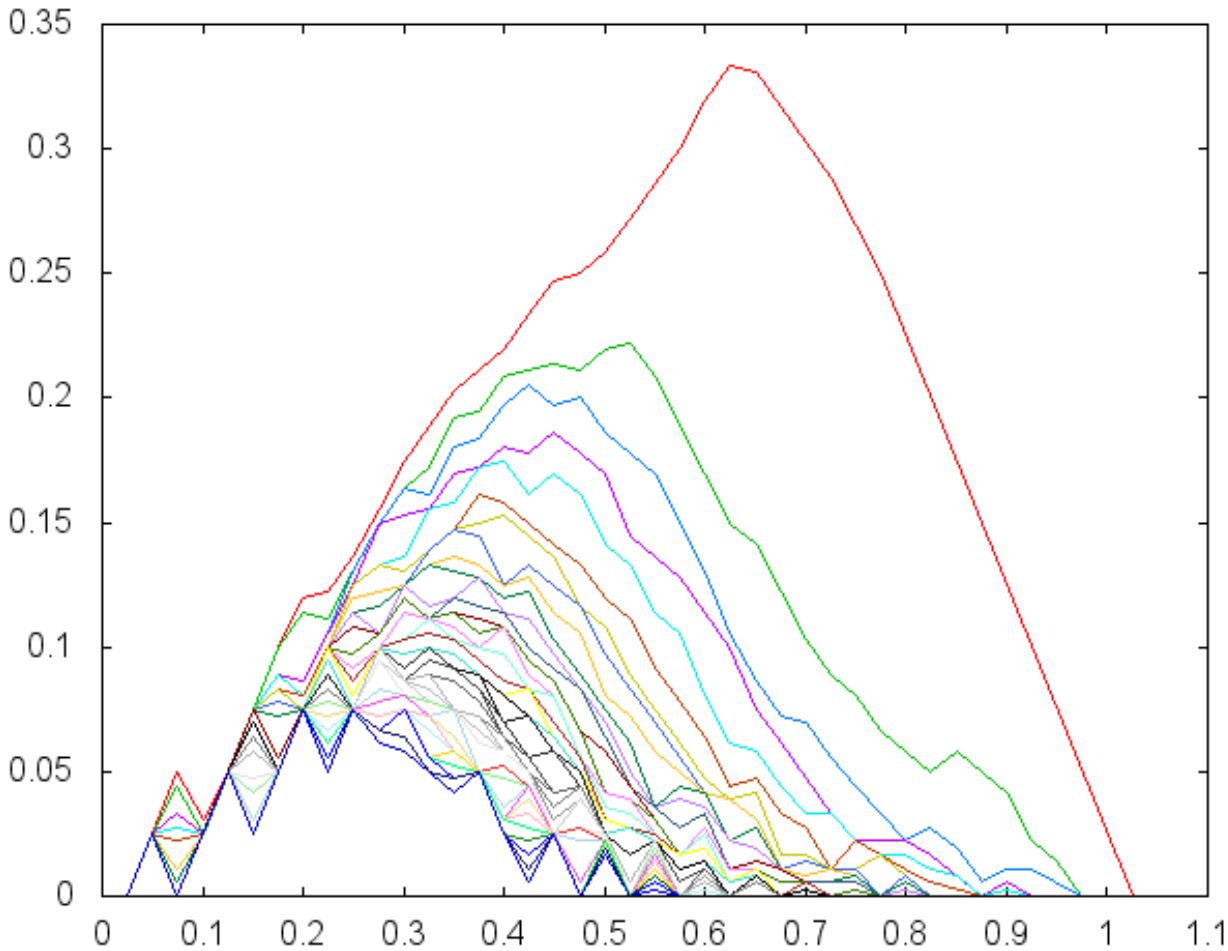


Figure: blue=$H0$, green=$H1$, red=$H2$

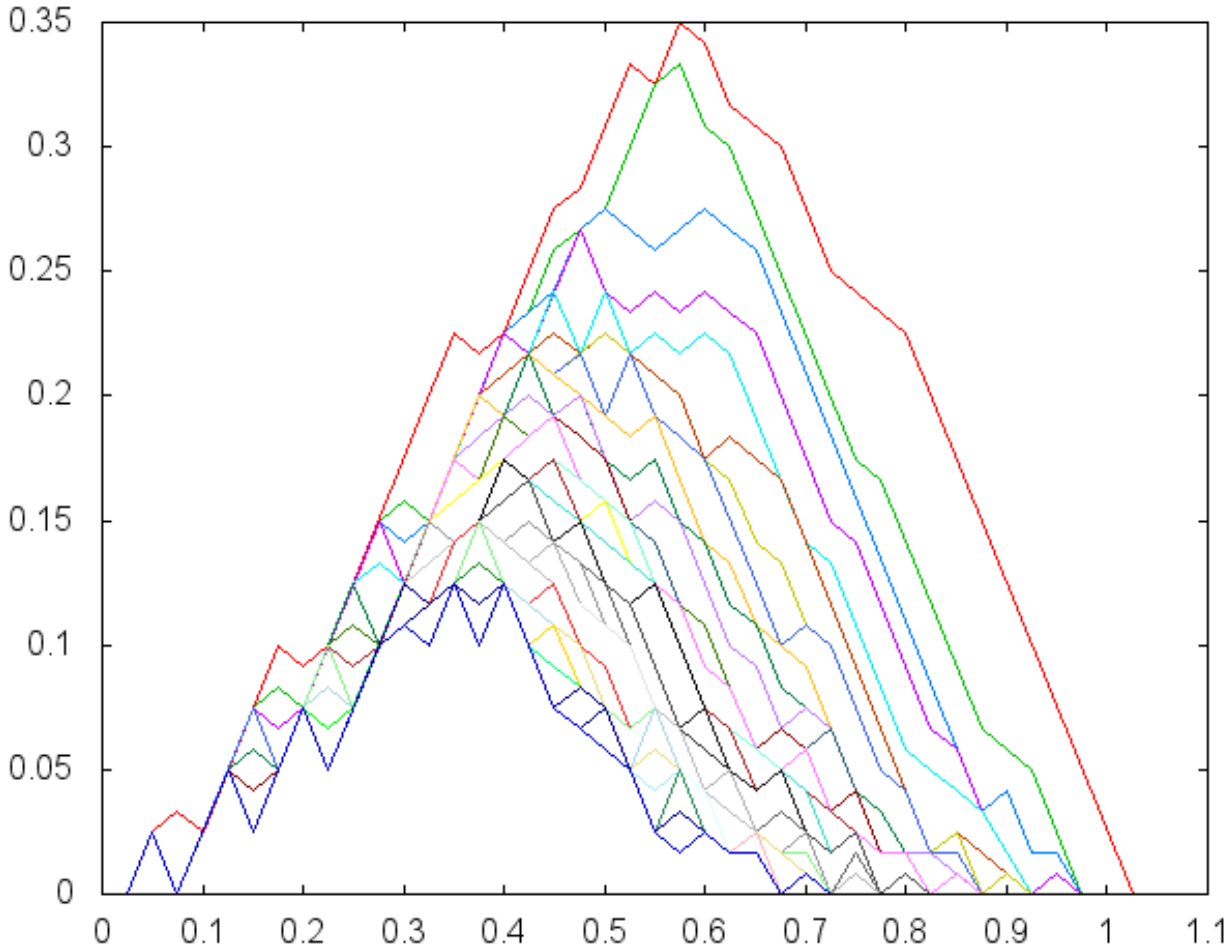
Conclusion: This statistic in $H2$ captures the separation along the dominant eigenvector for suitable choices of n.

Interpretation of statistic: For each n, capturing the average vortex "depth" over the top n "deepest" vortices.

Question to consider: How much did the choice of the Landscape L^2 metric influence the diffusion map embedding?



Question to consider: How much did the choice of the Landscape L^2 metric influence the diffusion map embedding?



Further lines of inquiry:

- How do the diffusion map projections change with different Landscape L^p metrics?
- What about d_B and d_{W^p}?
  - Does enstrophy lie along the dominant eigenvectors?
  - Will we see other, more suitable families of statistics to use?
- Can we find a statistic from the persistence diagrams that correlates with enstrophy?

New challenge: Families upon families of statistics!

Enter the lasso?

Regression Shrinkage and Selection via the Lasso

By ROBERT TIBSHIRANI†

University of Toronto, Canada

[Received January 1994. Revised January 1995]

SUMMARY

We propose a new method for estimation in linear models. The 'lasso' minimizes the residual sum of squares subject to the sum of the absolute value of the coefficients being less than a constant. Because of the nature of this constraint it tends to produce some coefficients that are exactly 0 and hence gives interpretable models. Our simulation studies suggest that the lasso enjoys some of the favourable properties of both subset selection and ridge regression. It produces interpretable models like subset selection and exhibits the stability of ridge regression. There is also an interesting relationship with recent work in adaptive function estimation by Donoho and Johnstone. The lasso idea is quite general and can be applied in a variety of statistical models: extensions to generalized regression models and tree-based models are briefly described.

Keywords: QUADRATIC PROGRAMMING; REGRESSION; SHRINKAGE; SUBSET SELECTION

1. INTRODUCTION

Consider the usual regression situation: we have data $(\mathbf{x}^i, y_i)$, $i = 1, 2, \ldots, N$, where $\mathbf{x}^i = (x_{i1}, \ldots, x_{ip})^T$ and y_i are the regressors and response for the ith observation. The ordinary least squares (OLS) estimates are obtained by minimizing the residual

Many thanks to...

My collaborators:

- Rutgers University: Miroslav Kramár, Konstantin Mischaikow
- Georgia Institute of Technology: Jeffrey Tithof, Balachandra Suri, Michael F. Schatz
- Virginia Tech: Mu Xu, Mark Paul
- Nagoya University: Takashi Ishihara
- University of Pennsylvania: Pawel Dlotko

The software creators:

- Vidit Nanda (Perseus)
- Pawel Dlotko (Persistence Landscape Toolbox)
- Shaun Harker (Subsampling/Cluster-delegator)
- Miro Kramár (Diffusion Map projection)
- Jonathan Reeve (who taught me all about scripting)

Funding: NSF, AFOSR, and DARPA.