# Introduction to MAPPER

### Leyda Almodóvar

You will find Mapper and instructions to download it and install it here: http://danifold.net/mapper

Or see page 2 of this document

	Welcome to the Python Mapper documentation!
annar ie an	algorithm for evolvation, analysis and visualization of data
vriat	is ryundi wapperr
• Install	Allon Demissioner
•	Requirements Developed
•	Installation
	Traubleshooting
	Mixed tins
	Compiling the documentation
Quick	start
Innut	tata
Filter	iunctions
	Mathematical definition
	Data structure
•	Filter functions in Python Mapper
<ul> <li>Cover</li> </ul>	methods
The N	lapper output window
<ul> <li>Custo</li> </ul>	m data processing in the GUI
0	Input data processing
•	Filter processing
<ul> <li>Modul</li> </ul>	e reference
<ul> <li>Copyr</li> </ul>	ight, references and citation info
۰	Copyright
۰	References
	<u>Citation info</u>

Make sure to look at http://danifold.net/mapper/installation/index.html for installing instructions and troubleshooting tips.

It is important that your data contains only numbers. Remove dollar signs, commas, NA, NaN and headers.

In order to download Mapper open up a terminal (found on the bottom of the screen):

File Edit View Termina Disk quotas for u Filesystem	I Tabs H 	lelp						
Disk quotas for u Filesystem	ser lal							
Disk quotas for u Filesystem	ser lal							
Disk quotas for u Filesystem	ser lal							
Filesystem		modovar	(uid 1539	90):				
notonno. (vol (anod	space	quota	limit	grace	files	quota	limit	grace
netappz:/vot/grad								
	953M	3200M	4000M		26034	4295m	4295m	
netapp1:/vol/csg	014	000101	007004		-	1005	1005	
natann1./val/bama	⊍K	30018M	30/20M		1	4295m	4295m	
netapp1:/vot/nome	page AK	225M	250M		2	1295m	4295m	
	-111	22311	20011		~	+23311	+2001	
lalmodovar@l-lnx0	01101[~	1% <b>h</b> g cl	one http	p://danif	old.net/	hg/map	per	

Figure 1: Open up a terminal

**Download:** Type the following in a terminal window: hg clone http://danifold.net/hg/mapper

In order to open Mapper you could double click the folder named 'mapper' and then double click the folder named 'bin' and then double click the file named 'MapperGUI.py'.

Alternatively, you could type **cd mapper** then press enter, then type **python mapper/bin/MapperGUI.py** and press enter.

2			Terminal					* <u>-</u> + ×
File Edit View Terminal	Tabs H	elp						
Disk quotas for us	ser lal	modovar	(uid 153	90):				
Filesystem	space	quota	limit	grace	files	quota	limit	grace
netappz./vot/grau	970M	3200M	4000M		26087	4295m	4295m	
netapp1:/vol/csg								
netann1./vol/bomer	0K	30618M	30720M		1	4295m	4295m	
netappi./vot/nomep	4K	225M	250M		2	4295m	4295m	
lalmodovar@l-lnx01	5101[~	]% cd ma	pper					
lalmodovar@l-lnx01	.5102[~	/mapper]	% python	mapper/b	oin/Mapp∉	erGUI.py		

Figure 2: In order to open the GUI from the terminal:

Figure 3: Once you press enter the GUI should appear and it looks like this. The next step is to load the data.

Python Mapper 🛛 🔺 🚽 🕈	\$
File Options Help	
Step 1: Input	-
Example Shapes Synthetic Shapes Load Data	
Preprocessing	
Data type: vector data, number of points: 20170, dimensionality: 2.	
Step 2: Metric	-
Euclidean	
Ambient/original metric	
O Intrinsic metric No. of nearest neighbors k 1 🔅 ε 1.0	
Minimal Is to make the data set connected: (2) Compute	
Minimal k to make the data set connected; (?)	
Step 3: Filter function	-
Filters for dissimilarity data Filters for vector data only	
kNN distance V. No. of nearest neighbors 5	
Filter transformation	٦
See histogram	
	5
min max View data	
Step 4: Mapper parameters	-
Cover Uniform 1-d cover 🗸 Intervals 15 🗘 Overlap	%
Clustering Single 🗸	
Cutoff First gap V Gap size 0.1	
Cleanup: minimal number of elements per simplex	
Quit Interrupt Generate Scale Graph Run Mappe	r
Input file selection was canceled.	_

## Figure 4: Browse for data file

Python Mapper 🛛 🔺 📥 🗮
File Options Help
Step 1: Input
Example Shapes Synthetic Shapes Load Data
Preprocessing
Data type: vector data, number of points: 20170, dimensionality: 2.
Step 2: Metric
Euclidean 🗸
Ambient/original metric
Ο Intrinsic metric No. of nearest neighbors k 1 🗘 ε 1.0
Minimal k to make the data set connected: (?) Compute
Stop 2: Eiltor function
Filters for dissimilarity data Filters for vector data only
kNN distance VNo. of nearest neighbors 5
Filter transformation
See histogram
min max View data
Step 4: Mapper parameters
Cover Uniform 1-d cover VIntervals 15 🗘 Overlap %
Clustering Single V
Cutoff First gap 🗸 Gap size 0.1
Cleanup: minimal number of elements per simplex
Quit Interrupt Senerate Scale Graph Run Mapper
Input file selection was canceled.

Figure 5: Mapper tells you the number of data points and dimensionality of your data set once you select the file

Python Mapper 🗛	_ + X
File Options Help	
Step 1: Input	-
Example Shapes Synthetic Shapes Load Data	
Preprocessing	
Data type: vector data, number of points: 20170, dimensionali	ty: 2.
Step 2: Metric	-
Euclidean 🗸	
Ambient/original metric	
Ο Intrinsic metric No. of nearest neighbors k 1 🔶 ε	1.0
Minimal k to make the data set connected: (?) Compute	
Step 3: Filter function	
Filters for dissimilarity data Filters for vector data only	
KNN distance V No. of hearest heighbors	5
Filter transformation	
See	histogram
	iew data
Step // Mapper parameters	
Cover Uniform 1-d cover V Intervals 15 0ver	ap 🗌 %
Cutoff First gap V Gap size 0.1	
Cleanup: minimal number of elements per simplex	
Quit Interrupt Generate Scale Graph F	Run Mapper
Input file selection was canceled.	

Python Mapper	∧ _ + X
File Options Help	
Step 1: Input	Ξ
Example Shapes Synthetic Shapes Load Data	
Preprocessing	
Data type: vector data, number of points: 20170, dimensi	onality: 2.
Step 2: Metric	•
Euclidean 🔽	
Ambient/original metric	
🔿 Intrinsic metric 🛛 No. of nearest neighbors k 🔲 🍃	ε 1.0
Minimal k to make the data set connected: (?) Compute	
Step 3: Filter function	Ξ
Filters for dissimilarity data Filters for vector data only	
kNN distance VNo. of nearest neighbo	ors 5
Filter transformation	
	See histogram
	No. 14
min max	view data
Step 4: Mapper parameters	Ξ
Cover Uniform 1-d cover V Intervals 15 🗘 C	overlap 🦳 %
Clustering Single 🗸	
Cutoff First gap 🗸 Gap size 0.1	
Cleanup: minimal number of elements per simplex	
Quit Interrupt Øenerate Scale Graph	Run Mapper
Input file selection was canceled.	

Figure 6: Choose a metric: Euclidean, Minkowski or Chebychev

Figure 7: Choose a filter function: Eccentricity, kNN distance, Distance to a measure, Density (Gaussian Kernel), Graph Laplacian or Distance matrix eigenvector

File Options Help
Step 1: Input
Example Shapes Synthetic Shapes Load Data
Preprocessing
Data type: vector data, number of points: 20170, dimensionality: 2.
Step 2: Metric
Euclidean 🗸
Ambient/original metric
Ο Intrinsic metric No. of nearest neighbors k 1 🗘 ε 1.0
Minimal k to make the data set connected: (?) Compute
Step 3: Filter function
Filters for dissimilarity data Filters for vector data only
kNN distance VNo. of career neighbors 5
Filter transformation
See histogram
View data
min max
Step 4: Mapper parameters
Step 4: Mapper parameters           Cover       Uniform 1-d cover       Intervals         15       Overlap       %
Step 4: Mapper parameters     Image: Cover Uniform 1-d cover       Cover Uniform 1-d cover     Intervals 15       Clustering Single     Image: Cover Cov
Step 4: Mapper parameters       □         Cover       Uniform 1-d cover       ✓ Intervals 15 ° Overlap       %         Clustering       Single       ✓         Cutoff       First gap       ✓ Gap size       0.1
Step 4: Mapper parameters         Cover         Uniform 1-d cover         Intervals         15         Overlap         %         Clustering         Single         Cutoff         First gap         Gap size         0.1         Cleanup: minimal number of elements per simplex         8
Step 4: Mapper parameters       Image: Cover Uniform 1-d cover Intervals 15 Overlap %         Clustering Single        Intervals 15 Overlap %         Cutoff First gap Gap size 0.1       Image: Clustering Intervals Parameters Per simplex Parameters Per Scale Graph Run Mapper         Image: Clustering Interrupt       8 Generate Scale Graph Run Mapper

Figure 8: View data (as long as it is 1-dimensional, 2-dimensional or 3-dimensional)

Python Mapper 🛛 🔺 📥 🗮					
File Options Help					
Step 1: Input					
Example Shapes Synthetic Shapes Load Data					
Preprocessing					
Data type: vector data, number of points: 20170, dimensionality: 2.					
Step 2: Metric					
Euclidean 🗸					
Ambient/original metric					
Ο Intrinsic metric No. of nearest neighbors k 1 🔷 ε 1.0					
Minimal k to make the data set connected: (?) Compute					
Step 3: Filter function					
Filters for dissimilarity data Filters for vector data only					
kNN distance V No. of nearest neighbors 5					
Filter transformation					
See histogram					
View data					
min Tax View data					
Step 4: Mapper parameters					
Cover Uniform 1-d cover V Intervals 15 🗘 Overlap %					
Clustering Single 🗸					
Cutoff First gap V Gap size 0.1					
Cleanup: minimal number of elements per simplex					
Quit Interrupt Generate Scale Graph Run Mapper					
Input file selection was canceled.					

Figure 9: Choose the type of cover, number of intervals and percentage of overlap between successive intervals

Python Mapper 🛛 🔺 🔺 🗶
File Options Help
Step 1: Input
Example Shapes Synthetic Shapes Load Data
Preprocessing
Data type: vector data, number of points: 20170, dimensionality: 2.
Step 2: Metric
Euclidean
Ambient/original metric
O Intrinsic metric Νο, of nearest neighbors k 1 📮 ε 1.0
Minimal k to make the data set connected: (2) Compute
Step 3: Filter function
Filters for dissimilarity data Filters for vector data only
kNN distance V No. of nearest neighbors 5
Filter transformation
See histogram
view data
Step 4: Mapper parameters
Cover Uniform 1-d cover 💙 Intervals 15 🗘 Overlap 🔗
Clustering Single V
Cutoff First gap V Gap size 0.1
Cleanup: minimal number of elements per simplex
Quit Interrupt Generate Scale Graph Run Mapper
Input file selection was canceled.

Figure 10: Choose clustering algorithm: Single, Complete, Average, Weighted, Median, Centroid, Ward

Python Mapper	∧ _ + X
File Options Help	
Step 1: Input	
Example Shapes Synthetic Shapes Load Data	
Preprocessing	
Data type: vector data, number of points: 20170, dimens	ionality: 2.
Step 2: Metric	
Euclidean V	
Ambient/original metric	
O Intrinsic metric No. of nearest neighbors k 1	ε 1.0
Minimal k to make the data set connected: (?) Comput	e
Step 3: Filter function	
Filters for dissimilarity data Filters for vector data only	
Filters for dissimilarity data Filters for vector data only	
kNN distance 🛛 🖌 No. of nearest neighb	ors 5
Filter transformation	
	See histogram
a construction of the second se	View data
min max	
Step 4: Mapper parameters	
Cover Uniform 1-d cover V Intervals 15	Overlap %
Clustering Single 🗸 🧲	
Cutoff First gap 💙 Gap size 0.1	
Cleanup: minimal number of elements per simplex	
Quit Interrupt Generate Scale Grap	Run Mapper
Input file selection was canceled.	

Python Mapper	∧ _ + ×
File Options Help	
Step 1: Input	
Example Shapes Synthetic Shapes Load Data	
Preprocessing	
Data type: vector data, number of points: 20170, dimensionality: 2.	
Step 2: Metric	
Euclidean 🗸	
Ambient/original metric	
🔿 Intrinsic metric 👘 No. of nearest neighbors k 🔳 💲	ε 1.0
Minimal k to make the data set connected: (?) Compute	e
Step 3: Filter function	
Hiters for dissimilarity data Hiters for vector data only	
kNN distance V. No. of nearest neighb	ors 5
Filter transformation	
	See histogram
	Nieur dete
min max	view data
Step 4: Mapper parameters	•
Cover Uniform 1-d cover V Intervals 15	Overlap %
Clustering Single 🗸	
Cutoff First gap V Gap size 0.1	
Cleanup: minimal number of elements per simplex	
Quit Interrupt Menerate Scale	Run Mapper
Input file selection was canceled.	





In order to run Mapper you must make certain choices.

#### 1. Metric

- (a) Euclidean: The Euclidean metric between two points  $\mathbf{x} = (x_1, x_2, x_n)$  and  $\mathbf{y} = (y_1, y_2, y_n)$  is given by  $d(\mathbf{x}, \mathbf{y}) = |\mathbf{x} \mathbf{y}| = \sqrt{\sum_{i=1}^n |x_i y_i|^2}$ .
- (b) Minkowski: The Minskowski metric of order p between two points  $\mathbf{x} = (x_1, x_2, x_n)$ and  $\mathbf{y} = (y_1, y_2, y_n)$  is defined as  $d(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^n (|x_i - y_1|^p)^{1/p}$ .
- (c) Chebychev: The Chebychev metric between two points  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  and  $\mathbf{y} = (y_1, y_2, \dots, y_n)$  is defined as  $d(\mathbf{x}, \mathbf{y}) = \max_i |x_i y_i|$ .

#### 2. Filter function

A filter function is a function on the data set,  $f: X \to \mathbb{R}^k$ . The Mapper algorithm supports general, vector-valued functions, while the GUI is restricted to real-valued functions (the case k = 1) for simplicity.

- (a) Eccentricity
- (b) kNN distance
- (c) Distance to a measure
- (d) Density, Gaussian Kernel
- (e) Graph Laplacian
- (f) Distance matrix eigenvector

#### 3. Type of cover

(a) Uniform 1-d cover

- (b) Balanced 1-d cover
- (c) Subrange decomposition

#### 4. Clustering algorithm

- (a) Single: The distance between two clusters is defined as that of the closest pair of individuals, where it only considers pairs consisting of elements from different clusters:  $d(r, s) = min(dist(x_{ri}, x_{sj})), i \in (i, ..., n_r), j \in (1, ..., n_s).$
- (b) Complete: The distance between two clusters is defined as that of the most distant pair of individuals, where it only considers pairs consisting of individuals from different clusters:  $d(r,s) = max(dist(x_{ri}, x_{sj})), i \in (i, ..., n_r), j \in (1, ..., n_s)$ .
- (c) Average: The distance between two clusters is defined as the average of the distance between all pairs of individuals that are made up of one individual from each cluster:  $d(r,s) = \frac{1}{n_2 n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} dist(x_{ri}, x_{sj}).$
- (d) Weighted: The distance between two clusters is defined as the weighted average of the distance between all pairs of individuals that are made up of one individual from each cluster. It uses a recursive definition for the distance between two clusters:  $d(r,s) = \frac{(d(p,s)+d(q,s))}{2}$ .
- (e) Median: The distance between two clusters is defined as  $d(r,s) = \|\tilde{x}_j \tilde{x}_s\|_2$ where  $\tilde{x}_r$  and  $\tilde{x}_s$  are weighted centroids for the clusters r and s and  $\tilde{x}_r$  is defined recursively as  $\tilde{x}_r = \frac{1}{2}(\tilde{x}_p + \tilde{x}_q)$ .
- (f) Centroid: The distance used is the Squared Euclidean distance between centroids  $d(r,s) = \|\tilde{x}_j \tilde{x}_s\|_2$  where  $\tilde{x}_r = \frac{1}{n_r} \sum_{i=1}^{n_r} x_{ri}$ .
- (g) Ward: The distance used is  $d(r,s) = \sqrt{\frac{2n_r n_s}{(n_r + n_s)}} \|\tilde{x}_r \tilde{x}_s\|_2$ . The distance is defined as the incremental sum of squares, that is, the increase in the total within-cluster sum of squares as a result of joining two clusters. The within-cluster sum of squares is defined as the sum of the squares of the distances between all objects in the cluster and the centroid of the cluster.

To find more information about the covers provided by Mapper: http://danifold.net/mapper/cover.html

To find more information about the filters provided by Mapper: http://danifold.net/mapper/filters.html