Manifold Learning

Ιωάννης Γκίνης
($s6170022$)

Επιβλέπων: Αθανάσιος Γιαννακόπουλος

Διπλωματική Εργασία

Οικονομικό Πανεπιστήμιο Αθηνών
Τμήμα Στατιστικής
Έτος σπουδών:4ο

Εξεταστές: Γεώργιος Παπαγιάννης
Στυλιανός Ξανθόπουλος

[Αυτή η σελίδα έχει αφεθεί σκοπίμως κενή]

Περίληψη

Ο σκοπός της παρούσας διπλωματικής εργασίας είναι, η μελέτη αλγορίθμων μείωσης διαστάσεων υψηλού όγκου δεδομένων(πολυμεταβλητών δεδομένων) και η εφαρμογή τους σε προσομοιωμένα δεδομένα. Οι αλγόριθμοι εκμάθησης πολλαπλοτήτων (manifold learning), που διέπουν τα δεδομένα που μελετάμε, απασχολούν την τελευταία δεκαετία σε πολύ μεγάλο βαθμό την μαθηματική και ευρύτερα την επιστημονική κοινότητα. Η συνεισφορά τους βρίσκει μεγάλο αντίκτυπο σε τομείς, όπως αυτούς της βιολογίας, των χρηματοοικονομικών και γενικότερα στην ανάλυση μεγάλου όγκου δεδομένων. Στα πρώτα κεφάλαια της παρουσας διπλωματικης γίνεται η εισαγωγή στην έννοια της πολλαπλότητας(manifold) με την παράθεση και μελέτη χαρακτηριστικών παραδειγμάτων, όπως είναι ο κύκλος ή η σφαίρα. Εισάγουμε την έννοια των λείων και διαφορίσιμών δομών, ορίζουμε το εφαπτόμενο διάνυσμα ως παραγώγιση στο χώρο των λείων συναρτήσεων σε ένα ανοικτό υποσύνολο του $\mathbb{R}^n$, καθώς και τον εφαπτόμενο χώρο. Ορίζουμε την γεωμετρία του Riemann, εισάγουμε την μετρική του και ορίζουμε την έννοια την βαθμίδας(gradient). Αναπτύσουμε την έννοια του affine connection , ώστε να είμαστε σε θέση να μιλήσουμε για το Levi-Civita connection. Με την βοήθεια του Levi-Civita connection, ορίζουμε την απόκλιση (divergence) και έτσι είμαστε σε θέση, να αποτυπώσουμε τον τελεστή Laplace-Beltrami. Συνεχίζοντας διατυπώνουμε την Φασματική θεωρία της Λαπλασιανής, μιλώντας για τους χώρους Sobolev, το φασματικό θεώρημα, το Heat equation και τέλος για τις αναλογίες που έχει η Λαπλασιανή ενός γραφήματος(graph) με τον τελεστή Laplace-Beltrami. Κατόπιν επικεντρωνόμαστε στο μαθηματικό υπόβαθρο και εισάγουμε τις γραμμικές μεθόδους μείωσης διαστάσεων(PCA, Kernel PCA και MDS), καθώς και τις μη γραμμικές(ISOMAP, LLE, Laplacian Eigenmaps και Diffusion maps). Τέλος χρησιμοποιώντας τους αλγόριθμους που έχουμε ορίσει, γίνονται κάποια παραδείγματα μείωσης διαστάσεων πάνω σε δεδομένα που η γεωμετρική δομή τους παρουσιάζεται μέσω κάποιων μη γραμμικών πολλαπλοτήτων(non-linear manifolds), όπως είναι το Swiss roll, το Swiss roll με τρύπα, την Σφαίρα και την Σφαίρα με τρύπα.

Abstract

The goal of this thesis is to represent the different methods of dimensionality reduction(linear and non-linear) applied on simulated data sets. These algorithms are considered to be a hot topic in the world of mathematics and computer science, as they play a crucial role in many scientific areas such as biology, finance and data science. In the first sections of this thesis we introduce the idea of a manifold whilst doing some distinct examples like the circle or the sphere. In addition we introduce the idea of Smooth and differentiable structures, we define the tangent vector as the derivative of the space of smooth functions on an open subset of $\mathbb{R}^n$ and having done that we define the tangent space. Also we try to describe the basics of the Riemannian geometry and the Riemannian metric which is a family of smoothly varying inner products on the tangent spaces of a smooth manifold. Then we define the Gradient and the Divergence(through the Levi-Civita connection) so we can define the Laplace-Beltrami operator. Moving on we introduce the Spectral Theory of the Laplacian starting with some preliminaries (Sobolev spaces),we define the Spectral Theorem, the Heat equation and we provide a justification for why the eigenfunctions of the Laplace Beltrami operator have properties desirable for embedding. Moving on, giving emphasis on the mathematical background we introduce the linear dimensionality reduction algorithm (PCA, Kernel PCA and MDS) and the non-linear ones (ISOMAP, LLE, Laplacian Eigenmaps and Diffusion maps). Using these algorithms that we defined, we then do some examples of dimensionality reduction on non-linear manifold like the Swiss Roll, the Swiss Roll with a hole, the Sphere and the Sphere with a hole. Finally, we compare each algorithm to see which is the best in each situation.

Περιεχόμενα

1 Manifold **1**
1.1 Τοπολογικές ιδιότητες των Manifold . . . 3
1.2 Manifolds με σύνορα . . . 4
1.3 Λείες και διαφορίσιμες δομές . . . 5

2 Εφαπτομένα διανύσματα και Εφαπτόμενοι χώροι **6**

3 Τανυστές **7**
3.1 Τανυστικά Πεδία . . . 7

4 Γεωμετρία του Riemann **8**
4.1 Τελεστής Laplace-Beltrami . . . 10

5 Φασματική θεωρία της Λαπλασιανής **14**
5.1 Χώροι Sobolev . . . 15
5.2 Ιδιοσυναρτήσεις και ιδιοτιμές . . . 17
5.3 Φασματικό θεώρημα . . . 17
5.4 Heat equation . . . 18

6 Γραμμικές τεχνικές μείωσης διαστάσεων **23**
6.1 Principal Component Analysis . . . 23
6.2 Kernel PCA . . . 25
6.3 Multidimensional Scaling . . . 27

7 Μη Γραμμική Μείωση Διαστάσεων **30**
7.1 Isomap . . . 30
7.2 Locally linear embedding . . . 35
7.3 Spectral Clustering . . . 39
7.4 Laplacian Eigenmap . . . 43
7.5 Diffusion Maps . . . 43

8 Πειράματα **46**
8.1 Swiss Roll . . . 46
8.2 Swiss Roll με τρύπα . . . 48
8.3 Σφαίρα . . . 51
8.4 Σφαίρα με τρύπα . . . 54

9 Παράρτημα **59**

[Αυτή η σελίδα έχει αφεθεί σκοπίμως κενή]

1 Manifold

Για να μπορέσουμε να ορίσουμε το Manifold αρχικά θα χρειαστεί να έχουμε ορίσει το τι είναι ένας τοπικός Ευκλείδειος χώρος.

Ορισμός 1.1. Ένας d-διάστατος τοπικός Ευκλείδειος χώρος είναι ένας τοπολογικός χώρος X τέτοιος ώστε, $\forall x \in X \ \exists \ u \subset X$ μία ανοιχτοί γειτονιά του x όπου είναι homeomorphic σε ένα ανοιχτό σύνολο $\hat{u} \subset \mathbb{R}^d$.

Επομένως τώρα μπορεί να οριστεί η έννοια της πολλαπλότητας (Manifold).

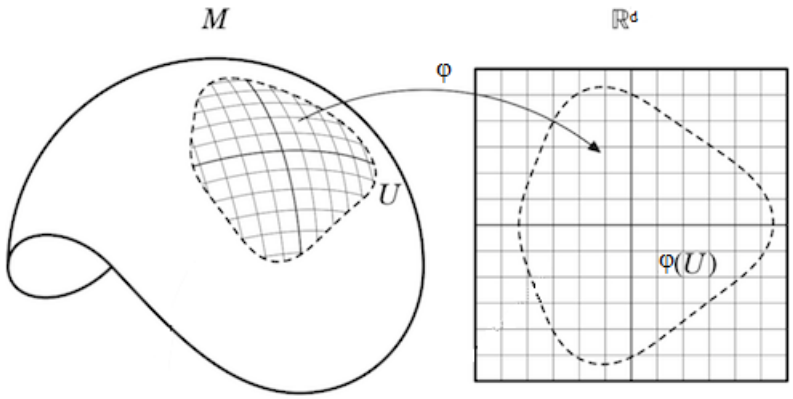
Ορισμός 1.2. Ένας d-διάστατο Manifold είναι τοπολογικός χώρος M τέτοιος ώστε

1. M είναι d-διάστατος τοπικός Ευκλείδειος χώρος
2. M είναι Hausdorff ($\forall x, y \in M, \ \exists \ u, v \subset M$ έτσι ώστε $x \in u, \ y \in v, \ u \cap v = \emptyset$)
3. M έχει αριθμήσιμη βάση

Έστω ότι για κάθε σημείο p του M έχουμε

1. ένα ανοιχτό σύνολο $U \subset M$ που περιέχει το p,
2. ένα ανοιχτό σύνολο $\tilde{U} \subset \mathbb{R}^d$
3. και τον ομοιομορφισμό $\varphi : U \to \tilde{U}$.

Το ζευγάρι (U, φ) λέγεται chart. Το σύνολο U στο chart λέγεται chart domain και η απεικόνιση φ ονομάζεται chart map. Επομένως τα component functions του φ που ορίζενται ως $\varphi(p) = \big(x^1(p), x^2(p), ..., x^n(p)\big)$ ονομάζονται local coordinates (τοπικές συντεταγμένες) στο U. Το συμπέρασμα το οποίο βγάζουμε είναι ότι με τις τοπικές συντεταγμένες ορίζουμε συντεταγμένες στο manifold. Συγκεκριμένα, έχοντας ένα chart (U, φ) στο M και chart map $\varphi : U \to \tilde{U}$ μπορούμε να δούμε ότι το U θα είναι ανοιχτό υποσύνολο του M και (όσο δουλεύουμε με το chart) ανοιχτό υποσύνολο του $\mathbb{R}^d$. Δηλαδή μπορούμε να σκεφτούμε ότι με την βοήθεια του φ μπορούμε να απεικονίσουμε την εικόνα του $U \subset M$ στο $\tilde{U} \subset \mathbb{R}^d$. Αυτό το επιδιώκουμε διότι μπορούμε να κάνουμε λογισμό στο $\mathbb{R}^d$(Σχήμα 1).



Σχήμα 1: `http://kahrstrom.com/mathematics/illustrations.php`

Παράδειγμα 1. Ένας κύκλος μπορεί να γραφτεί ως

$$S^1 = \{(\cos\theta, \sin\theta) \ : \ \theta \in [0, 2\pi)\}.$$

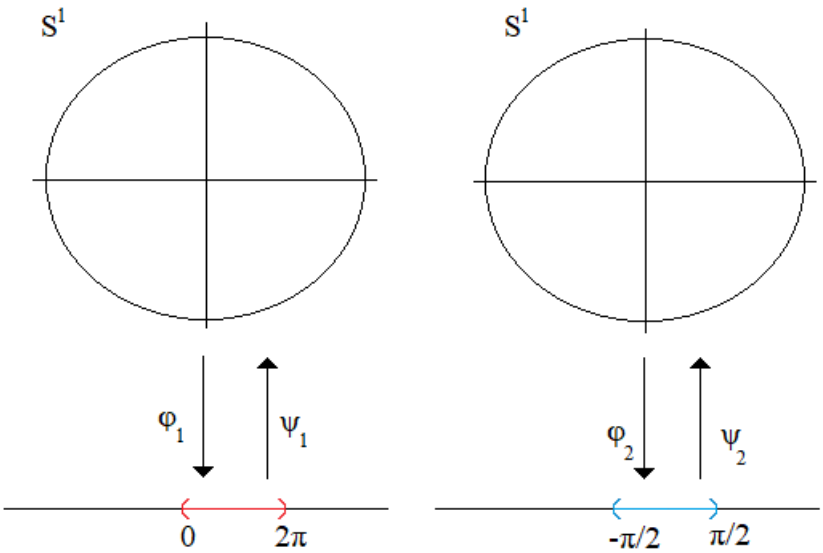
Ο κύκλος είναι ένα manifold με 1 διάσταση. Έστω

$$U_1 = \{(\cos\theta, \sin\theta) \ : \ \theta \in (0, 2\pi)\},$$

$\varphi_1 : U_1 \to (0, 2\pi)$ και $\psi_1 = \varphi_1^{-1}$ ορίζεται ως

$$\psi_1(\theta) = (\cos\theta, \sin\theta)\,.$$

Αφού το U_1 δεν καλύπτει όλο τον κύκλο, θα χρειαστούμε ένα ακόμα chart. Για παράδειγμα, μπορούμε να πάρουμε $U_2 = \{(\cos\theta, \sin\theta) \ : \ \theta \in (-\pi/2, \pi/2)\}$ και $\varphi_2 : U_2 \to (-\pi/2, \pi/2)$ και το $\psi_2 = \varphi_2^{-1}$.



Σχήμα 2

Παράδειγμα 2. Σφαίρα n διάστασης

Ξέρουμε ότι:

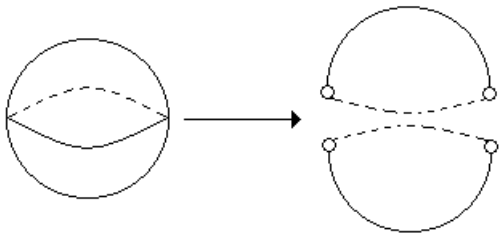
$$S^n = \{x \in \mathbb{R}^{n+1} |\ |x| = 1\}$$

Χωρίζουμε την σφαίρα σε δύο ημισφαίρια που θα τα ονομάσουμε u_i^+ και u_i^-. Συγκεκριμένα θα έχουμε

$$u_i^+ = \{x \in S^n |\ x_i > 0\}$$

$$u_i^- = \{x \in S^n |\ x_i < 0\}$$

όπου είναι ανοιχτά.



Σχήμα 3

Άρα θα έχουμε τους εξής ομοιομορφισμούς

$$\varphi_i^+ : u_i^+ \to \mathbb{R}^n,\ \ \varphi_i^+(x) = \langle x_1, ..., x_{i-1}, x_{i+1}, ..., x_n \rangle$$

και

$$\varphi_i^- : u_i^- \to \mathbb{R}^n,\ \ \varphi_i^-(x) = \langle x_1, ..., x_{i-1}, x_{i+1}, ..., x_n \rangle.$$

Τα φ_i^+, φ_i^- είναι ομοιομορφισμοί διότι το φ_i είναι γραμμικό και ως εκ τούτου είναι ομοιομορφισμός.

Πρόταση 1. Αν M, N είναι manifolds, το καρτεσιανό γινόμενο $(M \times N)$ είναι manifold με διαστάση=διάσταση(M) + διάσταση(N).

Απόδειξη

Από τον ορισμό 1.2 ξέρουμε ότι για να είναι ένας τοπολογικός χώρος manifold χρειάζεται να ισχύουν τα 1,2 και 3. Επομένως

1. Έστω $\langle x, y\rangle \in M \times N$, όπου $x \in \langle u, \varphi\rangle$, $y \in \langle v, \psi\rangle$ διότι το M και το N είναι manifolds. Άρα έχουμε ότι $\langle x, y\rangle \in \langle u \times v, \varphi \times \psi\rangle$ όπου $\varphi \times \psi : \ u \times v \to \mathbb{R}^{n+m}$ και $\langle x, y\rangle \mapsto \langle \varphi(x), \psi(y)\rangle$. Είναι εύκολο να δούμε ότι είναι ομοιομορφισμός και το $u \times v$ είναι ανοιχτό σύνολο.
2. Είναι Hausdorff διότι απότελείται από γινόμενα Hausdorff χώρων.
3. Έχει αριθμήσιμη βάση γιατί το γινόμενο 2 αριθμήσιμων βάσεων είναι μία αριθμήσιμη βάση.

Άρα τα πεπερασμένα καρτεσιανά γινόμενα από manifolds είναι manifold$(M_1 \times M_2 \times M_3 \times ... \times M_n)$.

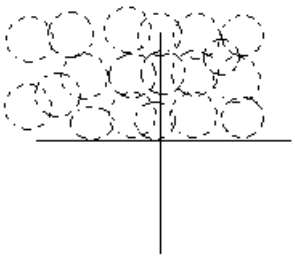
■

1.1 Τοπολογικές ιδιότητες των Manifold

Ορισμός 1.3. Λέμε ότι $U \subset X$ είναι προ-συμπαγής αν το $\bar{U}$(η κλειστότητα του U) είναι συμπαγές.

Αφού ορίσαμε την προ-συμπάγεια καταλήγουμε στο εξής θεώρημα:

Θεώρημα 1.4. Κάθε Manifold M έχει βάση που αποτελείται από προ-συμπαγή ανοιχτά σύνολα(π.χ σχήμα 4).



Σχήμα 4

Στο σχήμα 4 βλέπουμε ότι οι ανοιχτές μπάλες καλύπτουν το Manifold και ότι η κλειστότητα τους θα είναι συμπαγή. Η ένωση αυτών τον ανοιχτών συνόλων θα είναι μία βάση στο manifold μας.

Πρόταση 2.

1. Το M είναι locally path connected [1]
2. Το M είναι connected αν και μόνο αν είναι path connected
3. Τα connected components = path connected components
4. Έχει πολλά μετρήσιμα open connected components
5. Είναι τοπικά συμπαγές [2]

Τώρα θα ορίσουμε τι σημαίνει τοπικά πεπερασμένο.

Ορισμός 1.5. Έστω $\mathbf{X}$ μία συλλογή από υποσύνολα του M. Το $\mathbf{X}$ είναι τοπικά πεπερασμένο αν $\forall x \in M, \exists u \supset x$ ανοιχτό έτσι ώστε u να τέμνει μόνο πεπερασμένα στοιχεία του $\mathbf{X}$.

[1] locally path connected : Για κάθε σημείο x, μπορει να βρεθεί ένα διάστημα γύρω του έτσι ώστε, κάθε άλλο σημείο σε αυτό μπορεί να συνδεθεί από ένα συνεχές μονοπάτι.

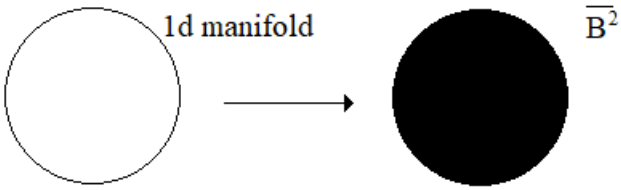
[2] Τοπικά συμπαγές: Γύρω από κάθε σημείο x μπορούμε να διαλέξουμε ένα ανοιχτό διάστημα έτσι ώστε ένα συμπαγές σύνολο το περιέχει.

Ορισμός 1.6. Δοθέντος $\mathbf{X}$ μια συλλογή, λεμέ ότι $\mathbf{Y}$ είναι εκλέπτυνση (refinement) αν για κάθε $x \in \mathbf{X}$, $\exists y \in \mathbf{Y}$ έτσι ώστε $y \subset x$.

Θεώρημα 1.7. Κάθε manifold έχει μία ανοιχτή κάλυψη όπου έχει refinement που είναι από προ-συμπαγής μπάλες και είναι τοπικά πεπερασμένες (para-compact).

1.2 Manifolds με σύνορα

Ας ξεκινήσουμε με παράδειγμα τον κύκλο. Ξέρουμε ότι ο κύκλος είναι μία πολλαπλότητα διάστασης 1. Αν πάρουμε την ένωση του ανοιχτού δίσκου με αυτό τότε θα έχω το $\bar{B}^2$(σχήμα 5).



Σχήμα 5

Άρα το $\bar{B}^2$ είναι μια κλειστή μπάλα. Το πρόβλημα που προκύπτει είναι ότι για τα σημεία που βρίσκονται στα όρια δεν μπορούμε να πάρουμε ανοιχτά σύνολα(δεν θα είναι ανοιχτό άμα το απεικονίσουμε στους πραγματικούς αριθμούς. Δηλαδή δεν μπορεί ομοιομορφικά να γίνει η απεικόνιση σε ένα ανοιχτό σύνολο στους πραγματικούς αριθμούς).

Ορισμός 1.8. Ένα manifold με σύνορο θα είναι ένας χώρος M τέτοιος ώστε

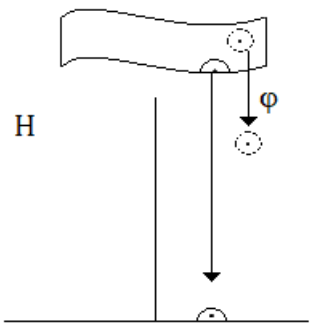
1. $\forall x \in M$, $\exists$ ανοιχτό $u \subset M$ όπου $u \supset x$ και ένας ομοιομορφισμός $\varphi : u \to \hat{u} \subset \mathcal{H}^n$(όπου $(\mathcal{H}^n)$[3] είναι το half plane)
2. Όπως και στον Ορισμό 1.2
3. Όπως και στον Ορισμό 1.2

Το chart (u, φ) λέγεται interior chart αν $\varphi(u) \cap \{x|\ x^1 = 0\} = \emptyset$ και boundary chart αν $\varphi(u) \cap \{x|\ x^1 = 0\} \neq \emptyset$. Λέμε ότι x είναι interior αν είναι στο χωρίο(domain) του interior chart . Λέμε ότι x είναι σύνορο(boundary) αν είναι στο boundary chart .
$M^\circ = \{$i-interior points$\}$
$\partial M = \{$ σύνολο από boundary points$\}$.

Ορισμός 1.9. Το manifold M (από τον ορίσμό 1.8) έχει τις εξής ιδιότητες

1. Το M έχει βάση από προ-συμπαγής μπάλες
2. Το M είναι τοπικά συμπαγές
3. Το M είναι locally path connected
4. Το M εχει μετρήσιμα connected components όπου είναι path connected components.

[3] $\mathcal{H}^n = \{x|\ x \in \mathbb{R}^n \text{και} x^1 \geq 0\}$



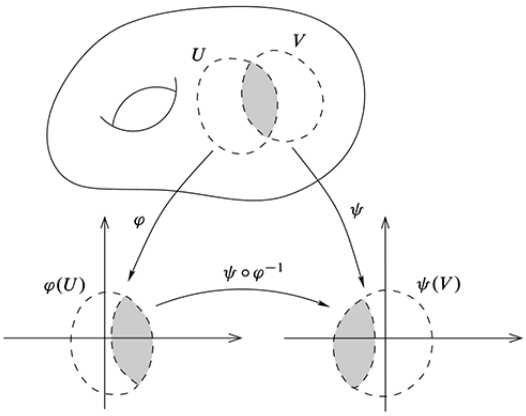
Σχήμα 6

1.3 Λείες και διαφορίσιμες δομές

Ορισμός 1.10. Μια συνάρτηση $f : u \subset \mathbb{R}^n \to v \subset \mathbb{R}^m$ είναι C^k αν $\partial^i_k[f^j]$ είναι συνεχής($\partial^i_k = \frac{\partial^k}{(\partial x^i)^k}$).

Αν αυτό ισχύει για $k \geq 0$ τοτε η συνάρτηση f είναι λεία(C^∞). Τώρα αν η f είναι αμφιμονοσήμαντη και η f^{-1} είναι λεία τότε λέγεται αμφιδιαφόριση (diffeomorphism).

Έστω $U \subset M$ και $V \subset M$ (το U και το V δεν είναι το ίδιο σύνολο), $\varphi : U \subset M \to \hat{U} \subset \mathbb{R}^n$ και $\psi : V \subset M \to \hat{V} \subset \mathbb{R}^n$. Το transition map είναι η απεικόνιση που ορίζεται ως $\psi \circ \varphi^{-1}$:(σχήμα 7).



Σχήμα 7: Απεικόνιση μετάβασης

Αφού τα φ και ψ είναι ομοιομορφισμοί, τότε το transition map είναι ομοιομορφισμός.

Ορισμός 1.11. Η συλλογή από charts όπου τα χωρία(domains) τους καλύπτουν το manifold όριζεται ως $\mathcal{A}$(atlas). Αν $(M, \mathcal{A})$ έχουν οποιαδήποτε δύο chart $(U, \varphi), (V, \psi) \in \mathcal{A}$ όπου $\varphi \circ \psi^{-1}$ και $\psi^{-1} \circ \varphi$ είναι C^k, τότε $\mathcal{A}$ είναι C^k.

2 Εφαπτομένα διανύσματα και Εφαπτόμενοι χώροι

Έστω μία σφαίρα M με εξίσωση $x^2 + y^2 + (z-1)^2 = 1$. Το σημείο $p = (0,0,0)$ ανήκει στο M και $\forall a, b \in \mathbb{R}$ όπου δεν είναι και τα δύο μηδέν, η ευθεία $\{(au, bu, o) : -\infty < u < \infty\}$ εφάπτεται στο M στο σημείο p. Το διάνυσμα $(a, b, 0)$ είναι εφαπτόμενο διάνυσμα του M στο p. Γενικότερα μπορούμε να σκεφτούμε να εφαπτόμενα διανύσματα σαν directional derivatives. Στο παράδειγμα το κάτω κομμάτι της σφαίρας μπορεί να οριστεί σαν μία γειτονιά U_p στο p όπου μπορεί να παραμετροποιηθεί από της συντεταγμένες (x, y). Μία C^1 πραγματική συνάρτηση f στο U_p θα είναι η $f(x, y)$. Έστω $\gamma(u) = (a(u), b(u))$ μία καμπύλη στο M με $|u| < u_0$ με u_0 αρκετά μικρό έτσι ώστε το $\gamma(u)$ να βρίσκεται εξ ολοκλήρου στο U_p . Η σύνθεση του γ και της f είναι μία πραγματική συνάρτηση στο $(-u_0, u_0)$.
Η παράγωγος στο $u = 0$ είναι

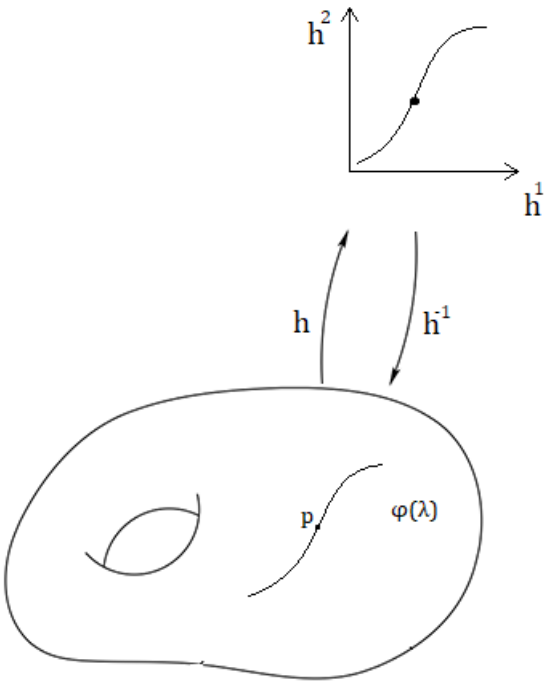
$$\frac{d}{du} f(\gamma(u))\bigg|_{u=0} = \frac{d}{du} f(a(u), b(u))\bigg|_{u=0} = \left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y}\right) f(x, y)\bigg|_{x=y=0}$$

Άρα το

$$t = \left(a\frac{\partial}{\partial x} + b\frac{\partial}{\partial y}\right)\bigg|_{x=y=0}$$

είναι το εφαπτόμενο διάνυσμα του M στο p.

Γενικά για ένα manifold ξέκινάμε ορίζοντας κάποιο σημείο που ανήκει σε αυτό(ας το πούμε πάλι p). Μέτα φτιάχνουμε το σύνολο όλων των λείων καμπυλών($\varphi(\lambda)$) που εφάπτουν το σημείο p. Επομένως θα έχουμε ότι το $\varphi : \mathbb{R} \to M$ όπου θα απεικονίζει το $\lambda \mapsto \varphi(\lambda)$. Αυτό που θέλουμε είναι να βρούμε την εικόνα αυτής της καμπύλης σε κάποιο συγκεκριμένο chart. Θα χρειαστούμε κάποιο chart h το όποιο μας απεικονίζει το manifold σε ένα υποσύνολο τον πραγματικών αριθμών. Με την βοήθεια του h δώσαμε συντεταγμένες στο ίδιο το manifold. Άρα για να φτιάξουμε το εφαπτόμενο διάνυσμα θα είναι πιο εύκολο για εμάς να χρησιμοποιήσουμε το chart (σχήμα 8).



Σχήμα 8

Για να γυρίσουμε στο manifold θα πάρουμε το h^{-1}. Ξέρουμε ότι ο διανυσματικός χώρος είναι μία γραμμική απεικόνιση. Επομένως θα χρειαστούμε μία αυθαίρετη συνάρτηση $f \in C^\infty(M)$ και $f : M \to \mathbb{R}$. Όπως είδαμε και στο παράδειγμα παραπάνω αυτό που θέλω να κάνω είναι κάνω περιορίσω την

συνάρτηση f πάνω στην καμπύλη$(f(\varphi(\lambda)))$. Το εφαπτόμενο διάνυσμα είναι

$$V_\varphi(f) = \frac{\partial}{\partial\lambda}\bigg|_{\lambda=0} f(\varphi(\lambda)).$$

Παρατηρούμε ότι το ίδιο κάναμε και στο παράδειγμα με την σφαίρα.

Επομένως τώρα μπορούμε να δώσουμε ένα πιο επίσημο ορισμό για το εφαπτόμενο διάνυσμα. Έστω M είναι ένα C^1 manifold. Δοθέντος κάποιου $p \in M$ και $F_p(M)$ είναι η οικογένεια συναρτήσεων του M που είναι C^1 στο p.

Ορισμός 2.1. Δοθέντος κάποιου $C^k - manifold\ M$, διάστασης n και $k \geq 1$ έχουμε ότι για κάθε $p \in M$, το εφαπτόμενο διάνυσμα του M στο p είναι οι C^1-καμπύλες που περνούν από το σημείο p στο M.

Άρα ο εφαπτόμενος χώρος στο p, T_pM, είναι ο διανυσματικός χώρος όλων των εφαπτόμενων διανυσμάτων στο p. Έχοντας διαλέξει ένα chart στο p με τοπικές συντεταγμένες $x_1, ..., x_d$ η βάση του T_pM είναι

$$\left(\frac{\partial}{\partial x_1}, ..., \frac{\partial}{\partial x_d}\right)\bigg|_{x(p)}$$

Έτσι $\dim T_pM = d$.

3 Τανυστές

Ορισμός 3.1. Έστω V ένας n-διαστατος διανυσματικός χώρος. Ο τανυστής k τάξης (k-τανυστής) είναι μία πολυγραμμική πραγματική συνάρτηση όριζεται στο γινόμενο $V \times ... \times V$ k φορές το V. Το σύνολο όλων των k-τανυστών είναι διανυσματικός χώρος και ορίζεται ως $\mathcal{T}^k(V^*)$[4].

Παράδειγμα 3. 1. Ο χώρος ενός 1-τανυστή $\mathcal{T}^k(V^*) = V^*$

2. Το εσωτερικό γινόμενο του $\mathbb{R}^n$ είναι ένας 2-τανυστής.

3. Η ορίζουσα είναι ένας τανυστής του $\mathbb{R}^n$.

Δοθέντος ενός τανυστή $\mathcal{T}$ και έναν τανυστή $\mathcal{S}$ ορίζουμε το τανυστικό γινόμενο ως το $(k+m)-$ τανυστή $\mathcal{T} \otimes \mathcal{S}$ που είναι

$$\mathcal{T} \otimes \mathcal{S}(u_1, ..., u_k, u_{k+1}, ..., u_{k+m}) := \mathcal{T}(u_1, ..., u_k) \cdot \mathcal{S}(u_{k+1}, ..., u_{k+m}).$$

Άμα πάρουμε k-τανυστή στο V^* αντί για το V, θα έχουμε τον χώρο $\mathcal{T}^k(V)$[5]. Αυτοί οι τανυστές λέγονται ανταλλοίωτοι(contravariant) τανυστές στο V, ενώ οι τανυστές του $\mathcal{T}^k(V^*)$ ονομάζονται συναλλοίωτοι(covariant) τανυστές στο V[6].

Ορισμός 3.2. Μικτοί (k,m)-τανυστές στο V, είναι μία πολυγραμμική συνάρτηση που ορίζεται στο $V \times ... \times V \times V^* ... \times V^*$ έχοντας k φορές το V και m φορές το V^*.

Άρα ο (k,m)-τανυστής είναι k φορές συναλλοίωτος και m φορές ανταλλοίωτος στο V. Ο χώρος όλων των (k,m)-τανυστών στο V δίνεται από $\mathcal{T}^{k,m}(V^*, V)$.

3.1 Τανυστικά Πεδία

Ορίζουμε το T_p^*M ως δυϊκό του εφαπτόμενου χώρου T_pM στο σημείο p του M[7].

Ορισμός 3.3. Ένα (k,m)-τανυστικό πεδίο είναι μία απεικόνιση που για κάθε $p \in M$ έχουμε έναν τανυστή $T \in \mathcal{T}^{k,m}(T_p^*M, T_pM)$

[4] Το V^* είναι ο δυϊκός χώρος(dual space) του V. Δηλαδή είναι ο χώρος των γραμμικών πραγματικών συναρτήσεων του V

[5] $(V^*)^* = V$

[6] Άρα οι ανταλλοίωτοι τανυστές στο V είναι οι συναλλοίωτοι τανυστές στο V^*

[7] Αυτό λέγεται και cotangent space του M στο p

4 Γεωμετρία του Riemann

Το (M, g) είναι Riemannian manifold και g είναι τύπου (0,2)-τανυστικό πεδίο στο M(όπου είναι το εσωτερικό γινόμενο σε κάθε $p \in M$)

$$g_p : T_pM \times T_pM \to \mathbb{R}$$

που είναι διγραμμικό (bilinear) και

1. $g_p(v, w) = g_p(w, v)\ \forall v, w \in T_pM$ (συμμετρικό)

2. $g_p(x, x) \geq 0$ και $g_p(x, x) = 0$ αν και μόνο αν $x = 0$(θετικά ορισμένο).

Η μετρική του Riemann g του M είναι μία λεία οικογένεια από εσωτερικά γινόμενα πάνω στους εφαπτόμενους χώρους του M. Έχοντας συντεταγμένες $x^1, x^2, ..., x^n$ για το M στο p μπορουμε να ορίσουμε :

$$g = \sum_{i,j=1}^{n} g_{ij} dx^i \otimes dx^j$$

Το g_{ij} είναι λείες συναρτήσεις όπου

$$g_{ij}(p) = g\left(\left.\frac{\partial}{\partial x^i}\right|_p, \left.\frac{\partial}{\partial x^j}\right|_p \right)$$

Δοθέντος κάποιου σύστημα συντεταγμένων και κάποιας μετρικής έχουμε:

$$G = [g_{ij}] \ \text{ και } \ G^{-1} = [g^{ij}]$$

Άρα λέμε ότι g^{ij} είναι αριθμοί τέτοιοι ώστε

$$\sum_k g^{ik} g_{kj} = \delta_{ij}.$$

Επομένως αφού ορίσαμε μια μετρική για ένα manifold μπορούμε να ορίσουμε την απόσταση. Για παράδειγμα μπορούμε να βρούμε το μήκος μιας καμπύλης ή την γωνία μεταξύ δύο μη μηδενικών διανυσμάτων.

Παράδειγμα 4. $\mathbb{R}^n$:
Το chart είναι ($\mathbb{R}^n$,ϕ) όπου $\phi(x) = x$. Η μετρική θα είναι

$$g_{\mathbb{R}^n}(x) = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}.$$

Αυτό σημαίνει ότι χρησιμοποιώντας συντεταγμένες $(x_1, ..., x_n)$, έχουμε

$$\left(\left.\frac{\partial}{\partial x_i}\right|_x, \left.\frac{\partial}{\partial x_j}\right|_x \right)_{g_{\mathbb{R}^n}} = \delta_{ij}$$

Παράδειγμα 5. Κύκλος:
Χρησιμοποιούμε το chart φ που ορίζεται από το αντίστροφο $\psi : (0, 2\pi) \to S^1$ με $\psi(\theta) = (\cos(\theta), \sin(\theta))$. Αφού όλα είναι μιας διάστασης μπορούμε να ορίσουμε την μετρική ως

$$g_{S^1}(\theta) = 1.$$

Επομένως έχουμε ότι

$$\left(\left.\frac{\partial}{\partial x_i}\right|_x, \left.\frac{\partial}{\partial x_j}\right|_x \right)_{g_{\mathbb{R}^n}} = 1$$

Παράδειγμα 6. Σφαίρα:
Έχουμε σφαίρα με σφαιρικές συντεταγμένες

$$\phi : (0, \pi) \times (0, 2\pi) \to S^2 \subset \mathbb{R}^3$$

όπου

$$\phi(\theta, \psi) = (\sin(\theta)\cos(\psi), \sin(\theta)\sin(\psi), \cos(\theta)).$$

Η μετρική g_{S^2} θα ορίζεται ως

$$g_{S^2}(\theta, \psi) = \begin{bmatrix} 1 & 0 \\ 0 & \sin^2(\theta) \end{bmatrix}.$$

Ορίζουμε $\mathrm{grad} f(p) : U \to \mathcal{X}(U)$ ως διανυσματικό πεδίο όπου

$$\langle \mathrm{grad} f, u \rangle(p) = df_p(u)$$

Εδώ θέλουμε να ορίσουμε τη βαθμίδα (gradient) συναρτήση της μετρικής.

Έστω $B = \{\frac{\partial}{\partial x^i}\}_{i=1,\dots,n}$ είναι μία βάση του εφαπτόμενου χώρου T_pU σε τοπικές συντεταγμένες για κάθε $p \in U$. Έτσι γράψουμε το $\mathrm{grad} f(p)$ ως

$$\mathrm{grad} f(p) = \sum_{i=1}^{n} \alpha_i(p) \frac{\partial}{\partial x^i}(p),$$

άρα ο στόχος μας είναι να βρούμε το $\alpha_i(p)$. Ξέρουμε ότι η απεικόνιση της βαθμίδας είναι γραμμική εξ ορισμού. Χρησιμοποιώντας την βάση B έχουμε

$$\langle \mathrm{grad} f, \frac{\partial}{\partial x^j} \rangle(p) = \sum_{i=1}^{n} \alpha_i(p) \langle \frac{\partial}{\partial x^i}, \frac{\partial}{\partial x^j} \rangle(p).$$

Καταλήγουμε ότι

$$\langle \mathrm{grad} f, \frac{\partial}{\partial x^j} \rangle = df_p\left(\frac{\partial}{\partial x^j}\right) = \frac{\partial f}{\partial x^j}(p),$$

μέσω του ορισμού της μετρικής του Riemann

$$\sum_{i=1}^{n} \alpha_i(p) \langle \frac{\partial}{\partial x^i}, \frac{\partial}{\partial x^j} \rangle(p) = \sum_{i=1}^{n} \alpha_i(p) g_{ij}(p)$$

Έτσι θα φτάσουμε στο αποτέλεσμα λύνοντας το γραμμικό συστημα

$$\frac{\partial f}{\partial x^j}(p) = \sum_{i=1}^{n} a_i(p) g_{ij}(p) \quad i = 1, \dots, n \Leftrightarrow \alpha_i(p) = \sum_{j=1}^{n} g^{ij}(p) \frac{\partial f}{\partial x^j}(p)$$

όπου $g^{ij}(p)$ ορίζουμε τον αντίστροφο του μετρικού τανυστή που είναι θετικά ημι-ορισμένος. Έτσι καταλήγουμε ότι

$$\mathrm{grad} f(p) = \sum_{i,j=1}^{n} g^{ij}(p) \frac{\partial f}{\partial x^j}(p) \frac{\partial}{\partial x^i}(p).$$

Στον $\mathbb{R}^n$ έχουμε $\nabla_g f(x) = \sum_{i,j=1}^{n} \frac{\partial f}{\partial x_i}(x) \left.\frac{\partial}{\partial x_i}\right|_x$.

Παράδειγμα 7. Σφαίρα:
Έχουμε σφαίρα με σφαιρικές συντεταγμένες

$$\phi : (0, \pi) \times (0, 2\pi) \to S^2 \subset \mathbb{R}^3$$

όπου

$$\phi(\theta, \psi) = (\sin(\theta)\cos(\psi), \sin(\theta)\sin(\psi), \cos(\theta)).$$

Η μετρική g_{S^2} θα ορίζεται ως

$$g_{S^2}(\theta,\psi) = \begin{bmatrix} 1 & 0 \\ 0 & \sin^2(\theta) \end{bmatrix}.$$

Έπομένως η βαθμίδα θα είναι

$$\nabla_{S^2} = \frac{\partial}{\partial\theta}\theta + \frac{1}{\sin\theta}\frac{\partial}{\partial\psi}\psi$$

Έστω X είναι ένα διανυσματικό πεδίο. Ορίζουμε την βαθμίδα μιας συνάρτησης f με μορφή πινάκων ως:

$$\nabla\ f = G^{-1}df$$

όπου df είναι η βαθμίδα στον $\mathbb{R}^n$.

Απόδειξη:
Έστω X είναι ένα διανυσματικό πεδίο. Ξεκινάμε λέγοντας ότι

$$\langle \nabla f, X\rangle = df(X)$$

όπου γίνεται

$$(\nabla f)^\top gX = df^\top X.$$

Παρατηρούμε ότι έχουμε το X και από τις δύο μεριές. Επομένως

$$(\nabla f)^\top g = df^\top$$

όπου δίνει

$$g^\top \nabla f = df$$

αφού το g είναι συμμετρικός πίνακας τότε θα έχουμε

$$\nabla f = g^{-1}df \quad \text{όπου} \quad g^{-1} = G^{-1}.$$

■

4.1 Τελεστής Laplace-Beltrami

Έστω $\mathcal{X}(M)$ είναι το σύνολο των λείων διανυσματικών πεδίων στο M. Τότε ένα affine connection είναι μια απεικόνιση $\nabla : \mathcal{X}(M)\times\mathcal{X}(M) \to \mathcal{X}(M)$ που συμβολίζεται ως $(X,Y) \overset{\nabla}{\longmapsto} \nabla_X Y$ και ικανοποιεί τα παρακάτω

1. $\nabla_{fX+gZ}Y = f\nabla_X Y + g\nabla_Z Y$ όπου f, g είναι βαθμωτές συναρτήσεις του M
2. $\nabla_X(aY + bZ) = a\nabla_X Y + b\nabla_X Z$ όπου a, b σταθερά
3. $\nabla_X(fY) = (Xf)Y + f\nabla_X Y$

Το διανυσματικό πεδίο $\nabla_X Y$ είναι η συναλλοίωτη παράγωγος (covariant derivative) του Y συναρτήση του X για το affine connection ∇.

Στο $\mathbb{R}^n$, η κλασσική κατά κατεύθυνση παράγωγος (directional derivative) όρίζει ένα affine connection

$$(\nabla_X Y)_p = \lim_{t\to 0}\frac{Y_{p+tX} - Y_p}{t}$$

Σε ένα manifold υπάρχουν απείρα το πλήθος affine connections. Στην περίπτωση που έχουμε γραμμικό manifold, μπορούμε να διακρίνουμε το canonical connection. Τώρα για ένα Riemannian manifold μπορούμε να διακρίνουμε το Levi-Civita connection όπου έχει δύο ιδιότητες:

1. Συμμετρία
2. $Zg(X,Y) = g(\nabla_Z X, Y) + g(X, \nabla_Z Y) \quad \forall Z, X, Y \in \mathcal{X}(M)$ (metric compatibility).

Στην περίπτωση των γραμμικών manifold, το Levi-Civita connection μας δίνει το canonical connection, όπου είναι η κατά κατεύθυνση παράγωγος.

Για να ορίσουμε την συμμετρία ενός affine connection πρέπει να εισάγουμε την έννοια των Lie Brackets των διανυσματικών πεδίων. Έστω X, Y είναι δύο διανυσματικά πεδία του M, όπου τα πεδία ορισμού τους περιέχουν ένα ανοιχτό σύνολο u. Επίσης έχουμε ότι $f \in F(u)$(όπου $F(u)$ είναι το σύνολο των λείων πραγματικών συναρτήσεων στο u). Ορίζουμε το Lie Bracket δύο διανυσματικών πεδίων ως

$$[X, Y]f = X(Yf) - Y(Xf).$$

Επίσης έχουμε ότι:

$$[X, Y](af + bg) = a[X, Y]f + b[X, Y]g.$$

Ικανοποιεί τον κανόνα του Leibniz

$$[X, Y](fg) = f([X, Y]g) + ([X, Y]f)g$$

άρα $[X, Y]$ είναι παραγώγιση (derivation) και έτσι ορίζει εφαπτόμενο διανυσματικό πεδίο που ονομάζεται Lie Bracket του X και Y.

Ορισμός 4.1. Ένα affine connection ∇ στο M είναι συμμετρικό αν

$$\nabla_X Y - \nabla_Y X = [X, Y] \quad \forall X, Y \in \mathcal{X}(M)$$

Θεώρημα 4.2. Σε ένα Riemannian manifold M, υπάρχει ένα μοναδικό affine connection ∇ όπου ικανοποιεί τις παρακάτω ιδιότητες

1. $\nabla_X Y - \nabla_Y X = [X, Y]$
2. $Zg(X, Y) = g(\nabla_Z X, Y) + g(X, \nabla_Z Y) \quad \forall Z, X, Y \in \mathcal{X}(M)$.

Αυτό το affine connection ονομάζεται Levi-Civita connection ή Riemannian connection του M και χαρακτηρίζεται από τον τύπο Koszul

$$2g\left(\nabla_X Y, Z\right) = Xg(Y, Z) + Yg(Z, X) - Zg(X, Y) + g(Z, [X, Y]) + g(Y, [Z, X]) + g(X, [Z, Y]).$$

Απόδειξη
Πρώτα θα δείξουμε την μοναδικότητα. Έστω X, Y και Z είναι τρία λεία διανυσματικά πεδία στο M. Άρα έχουμε ότι

$$\nabla_X Y - \nabla_Y X = [X, Y]$$
$$\nabla_Y Z - \nabla_Z Y = [Y, Z]$$
$$\nabla_Z X - \nabla_X Z = [Z, X],$$

και τις παρακάτω συνθήκες

$$g(\nabla_X Y, Z) + g(Y, \nabla_X Z) = X_g(Y, Z)$$
$$g(\nabla_Y Z, X) + g(Z, \nabla_Y X) = Y_g(Z, X)$$
$$g(\nabla_Z X, Y) + g(X, \nabla_Z Y) = Z_g(X, Y).$$

Παίρνουμε το άθροισμα των δύο πρώτων παραπάνω εξισώσεων και αφαιρούμε την τρίτη. Βάζοντας και τις συνθήκες συμμετρίας έχουμε:

$$2g\left(\nabla_X Y, Z\right) = Xg(Y, Z) + Yg(Z, X) - Zg(X, Y) + g(Z, [X, Y]) + g(Y, [Z, X]) + g(X, [Z, Y]).$$

Αυτό ορίζει το εσωτερικό γινόμενο του $\nabla_x Y$ με διανυσματικό πεδίο Z. Έτσι αποδείξαμε την μοναδικότητα. Δηλώνοντας την δεξιά μεριά του τύπου με $C(X, Y, Z)$ όπου C είναι C^∞ στο X και στο Z:

$$
\begin{aligned}
C(fX,Y,hZ) &= fX(hg(Y,Z)) + Y(fhg,(Z,X)) - hZ(fg(X,Y)) \\
&+ hg(Z,[fX,Y]) + g(Y,[hZ,fX]) + fg(X,[hZ,Y]) \\
&= fhXg(Y,Z) + fhYg(Z,X) - fhZg(X,Y) + f(Xh)g(Y,Z) + fY(h)g(Z,X) \\
&+ hY(f)g(Z,X) - hZ(f)g(X,Y) + hfg(Z,[X,Y]) + hfg(Y,[Z,X]) + fhg(X,[Z,Y]) \\
&- hY(f)g(Z,X) - fX(h)g(Y,Z) + hZ(f)g(Y,X) - fY(h)g(X,Z) = \\
&= fhC(X,Y,Z)
\end{aligned}
$$

Η απεικόνιση $x \mapsto C(X,Y,Z)_x$ εξαρτάται μόνο από το Y και των τιμών X_x και Z_x του διανυσματικού πεδίου X. Έχουμε ότι

$$
\begin{aligned}
C(X,fY,Z) &= fC(X,Y,Z) + X(f)g(Y,Z) - Z(f)g(X,Y) \\
&= X(f)g(Z,Y) + Z(f)g(X,Y) \\
&= fC(X,Y,Z) + 2X(f)g(Y,Z).
\end{aligned}
$$

Επομένως αν ορίσουμε το $\nabla_X Y$ έχοντας ότι $g(\nabla_X Y, Z) = \frac{1}{2}C(X,Y,Z)$, τότε $(\nabla_X Y)_x$ εξαρτάται από το X_x και Y. Άρα

$$
g(\nabla_X(fY),Z) = fg(\nabla_X Y, Z) + X(f)g(Y,Z) = g(f\nabla_X Y + X(f)Y, Z),
$$

έτσι ∇ ικανοποιεί τον κανόνα του Leibniz.

■

Διαλέγουμε ένα chart $\varphi : U \to V$ στο M. Ο παραπάνω τύπος με X, Y και Z που δίνονται από συντεταγμένες του του εφαπτόμενου διανυσματικού πεδίου ∂_i, ∂_j και ∂_k, δίνει το παρακάτω

$$
\Gamma_{ij}^k = \frac{1}{2} g^{kl}(\partial_i g_{jl} + \partial_j g_{il} - \partial_l g_{ij}).
$$

Αν M είναι ένα Riemannian Manifold διάστασης n και $X \in \mathcal{X}(M)$ ορίζουμε

$$
\mathrm{div}X(p) = \text{το trace της γραμμικής απεικόνισης } Yp \to \nabla_Y X(p)
$$

όπου ∇ είναι το Levi-Civita connection. Αυτό θέλουμε να το εκφράσουμε σε τοπικές συντεταγμένες. Αν $\{U_\alpha, \phi_\alpha\}$ είναι ένα chart και $x^1, ..., x^n$ είναι τοπικές συντεταγμένες στο $p \in U_\alpha$ έχουμε την βάση $span\{\frac{\partial}{\partial x^i}(p)\}_{1 \le i \le n} = T_pM$ και επίσης έχουμε ότι $X(p) = \sum_{i=1}^n \alpha^i(p)\frac{\partial}{\partial x^i}(p)$. Επομένως έχουμε ότι

$$
\nabla_{\frac{\partial}{\partial x^i}(p)} = \left(\sum_{j=1}^n a^j(p) \frac{\partial}{\partial x^j}(p) \right) =
$$

$$
\sum_{j=1}^n \left(\frac{\partial \alpha^j}{\partial x^i}(p) \frac{\partial}{\partial x^j}(p) + \alpha^j(p) \nabla_{\frac{\partial}{\partial x^i}(p)} \frac{\partial}{\partial x^j}(p) \right) =
$$

$$
\sum_{j=1}^n \left(\frac{\partial \alpha^j}{\partial x^i}(p) \frac{\partial}{\partial x^j}(p) + \alpha^j(p) \sum_{k=1}^n \Gamma_{ij}^k(p) \frac{\partial}{\partial x^k}(p) \right) =
$$

$$
\sum_{k=1}^n \left(\frac{\partial \alpha^j}{\partial x^i}(p) + \sum_{j=1}^n \alpha^j(p) \Gamma_{ij}^k(p) \right) \frac{\partial}{\partial x^k}(p) = \sum_{k=1}^n b^{ik}(p) \frac{\partial}{\partial x^k}(p).
$$

Έτσι έχουμε ότι

$$
\text{trace του } Y(p) \to \nabla_Y X(p) = \sum_{i=1}^n b^{ii}(p) =
$$

$$
\sum_{i=1}^n \left(\frac{\partial \alpha^i}{\partial x^i}(p) + \sum_{j=1}^n \alpha^j(p) \Gamma_{ij}^i(p) \right).
$$

Έστω f είναι μία πραγματική συνάρτηση με $f \in C^2$, ορισμένη σε ένα Riemannian manifold M(δηλαδή ένα differantiable manifold με μετρική Riemann).
Ο τελεστής Laplace-Beltrami (Laplace-Beltrami operator) Δ είναι ένας γραμμικός διαφορικός τελεστής(linear differential operator). Έχουμε ότι

$$\Delta f = div(grad(f))$$

όπου $grad(f)$ η βαθμίδα της f και div το divergence (απόκλιση) πάνω στο manifold.

Δοθέντος $\psi : \mathbb{R}^n \to \mathbb{R}^{n+k}$ ενός submanifold M του $\mathbb{R}^{n+k}$ με

$$g_{ij} = \langle \partial_i \psi, \partial_j \psi \rangle, \quad G = [g_{ij}]$$

$$W = \sqrt{\det G} \quad g^{ij} = G^{-1}$$

με $i, j = 1, .., n$, τότε ο τελεστής Laplace-Beltrami είναι:

$$\Delta f = \frac{1}{W} \sum_{i,j} \partial_i \left(g^{ij} W \partial_j f \right).$$

Αν M είναι χωρίο(domain) σε Ευκλείδειο επίπεδο $M \subset \mathbb{R}^n$, ο τελεστής Laplace-Beltrami θα είναι η πολύ γνωστή Λαπλασιανή

$$\Delta_{g_{\mathbb{R}^n}} = \frac{\partial^2}{\partial x_1^2} + ... + \frac{\partial^2}{\partial x_n^2}.$$

Παράδειγμα 8. Κύκλος:
Αφού $g_{S^1}(\theta) = 1$, έχουμε

$$\Delta_{g_{S^1}} = \frac{\partial^2}{\partial \theta^2}.$$

Παράδειγμα 9. Σφαίρα:
Αφού

$$g_{S^2}(\theta, \psi) = \begin{bmatrix} 1 & 0 \\ 0 & \sin^2(\theta) \end{bmatrix},$$

έχουμε ότι

$$\begin{aligned}
\Delta_{g_{S^2}} &= \frac{1}{\sqrt{\det g_{S^2}}} \left(\frac{\partial}{\partial \theta} \left(g^{\theta\theta} \sqrt{|\det g_{S^2}|} \frac{\partial}{\partial \theta} \right) + \frac{\partial}{\partial \psi} \left(g^{\psi\psi} \sqrt{|\det g_{S^2}|} \frac{\partial}{\partial \psi} \right) \right) \\
&= \frac{1}{\sin\theta} \frac{\partial}{\partial \theta} \left(\left(\sin\theta \frac{\partial}{\partial \theta} \right) + \frac{\partial}{\partial \psi} \left(\frac{1}{\sin\theta} \frac{\partial}{\partial \psi} \right) \right) \\
&= \frac{1}{\sin\theta} \frac{\partial}{\partial \theta} \left(\sin\theta \frac{\partial}{\partial \psi^2} \right) - \frac{1}{\sin^2\theta} \frac{\partial^2}{\partial \psi^2}.
\end{aligned}$$

Επομένως σε σφαιρικές συντεταγμένες έχουμε ότι

$$\Delta_{g_{S^2}} = \frac{1}{\sin\theta} \frac{\partial}{\partial \theta} \left(\sin\theta \frac{\partial}{\partial \psi^2} \right) - \frac{1}{\sin^2\theta} \frac{\partial^2}{\partial \psi^2}.$$

5 Φασματική θεωρία της Λαπλασιανής

Έχοντας μια μετρική Riemann g μπορούμε να ορίσουμε το volume form du_g στο M. Δοθέντος $A \in M$,

$$Vol(A) = \int_M \mathbb{1}_{A(x)}\, du_g(x).$$

όπου $\mathbb{1}_{A(x)} = 1$ αν $x \in A$ και $\mathbb{1}_{A(x)} = 0$ αν $x \notin A$. Έχοντας τοπικές συντεταγμένες $(x_1, ..., x_n)$ το volume form είναι

$$du_g(x) = \sqrt{\det(g_{ij}(x))} dx_1 ... dx_n.$$

Ξέρουμε ότι η μετρική Riemann μας δίνει ένα εσωτερικό γινόμενο για συναρτήσεις στο M. Δίνοντας $\phi, \psi : M \to \mathbb{R}$ έχουμε

$$\langle \phi, \psi \rangle_{L^2} = \int_M \phi(x)\psi(x) du_g(x).$$

Χρησιμοποιόντας το παραπάνω μπορούμε να ορίσουτμε τον χώρο Hilbert

$$L^2(M) = \{f : M \to \mathbb{R} \text{ συνάρτηση τέτοια ώστε } \langle \phi, \psi \rangle_{L^2} < \infty\}.$$

Θεώρημα 5.1. Έστω M είναι ένα Riemannian Manifold διάστασης n με σύνορο ∂M. Η ολοκλήρωση κάτα μέρη είναι:

$$\int_M \langle \nabla f, X \rangle = -\int_M f \mathrm{div} X + \int_{\partial M} f \langle X, v \rangle$$

ή

$$\int_M (f\Delta g - g\Delta f) = -\int_{\partial M} \left(f\frac{\partial g}{\partial v} - g\frac{\partial f}{\partial g} \right)$$

στον $\mathbb{R}^n$

Θεώρημα 5.2. (Τύπος του Green). Για κάθε συνάρτηση $\psi, \phi \in C^\infty(M)$ έχουμε

$$\int_M \psi\Delta\phi\, du + \int_M \langle \nabla\psi, \nabla\phi \rangle\, du = \int_{\partial M} \psi\frac{\partial\psi}{\partial\phi}\, d\sigma$$

Απόδειξη

Ξεκινάμε έχοντας

$$\int_M \langle \nabla\psi, \nabla\phi \rangle\, du.$$

Κάνουμε ολοκλήρωση κάτα μέρη

$$\int_M \langle \nabla\psi, \nabla\phi \rangle\, du = -\int_M \psi\Delta\phi\, du + \int_{\partial M} \psi\frac{\partial\psi}{\partial\phi}\, d\sigma.$$

έτσι καταλήξαμε στον τύπο του Green.

■

Έστω (M, g) συμπαγές Riemannian manifold με σύνορο ∂M και $\psi \in C^1(M)$ και $\phi \in C^2(M)$. Τότε

$$\int_M \psi\Delta_g\phi du_g = -\int_M \langle \nabla_g\psi, \nabla_g\phi \rangle_g\, du_g + \int_{\nabla M} \phi\nabla_v\phi\, d\sigma_g.$$

Όταν $\partial M = \varnothing$, έχουμε ότι η λαπλασιανή είναι αυτοσυζυγής

$$\langle \Delta_g\phi, \psi \rangle_{L^2} = \langle \phi, \Delta_g\psi \rangle_{L^2}.$$

5.1 Χώροι Sobolev

Έστω M είναι ένα χωρίο στο Ευκλείδειο χώρο $\mathbb{R}^d$. Ορίζουμε ως $\bar{M}$ την κλειστότητα του και $\partial M := \bar{M} \backslash M$ ως το σύνορο του. Επίσης ορίζουμε ως $M_e := \mathbb{R}^d \backslash \bar{M}$ ως το εξωτερικό χωρίο.

Έστω συναρτήσεις $u : M \to \mathbb{R}$ ορίζουμε ότι

$$D^\alpha u = \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} ... \partial x_d^{\alpha_d}}$$

είναι οι μερικές παράγωγοι τάξης $|\alpha| (|\alpha| = \sum_{i=1}^d \alpha_i)$. Εδώ, $\alpha = (\alpha_1, ..., \alpha_d)^T \in \mathbb{N}_0^d$.
Ορίζουμε ως $C^m(M), m \in \mathbb{N}_0$ τον γραμμικό χώρο των συνεχών συναρτήσεων στο M όπου οι μερικές παράγωγοι $D^\alpha u, |\alpha| \leq m$ υπάρχουν και είναι συνεχείς. Ο χώρος $C^m(M)$ είναι χώρος Banach με νόρμα

$$\|u\|_{C^m(M)} = \max_{0 \leq |\alpha| \leq m} \sup_{x \in M} |D^\alpha u(x)|.$$

Ο χώρος $C^{m,\alpha}(M), m \in \mathbb{N}_0, 0 < \alpha < 1$ ορίζεται ως τον γραμμικός χώρος των συναρτήσεων του $C^m(M)$ όπου η $m-$οστή μερική παράγωγος είναι Hölder συνεχής. Δηλαδή ότι για κάθε $\beta \in \mathbb{N}_0^d$ με $|\beta| = m$ υπάρχουν σταθερές Π_β έτσι ώστε για κάθε $x, y \in M$

$$|D^\beta u(x) - D^\beta u(y)| \leq \Pi_\beta |x - y|^\alpha.$$

Ο χώρος $C^{m,\alpha}(M)$ είναι και αυτός ένας χώρος Banach με νόρμα

$$\|u\|_{C^{m,\alpha}(M)} = \|u\|_{C^m(M)} + \max_{|\beta|=m} \sup_{x,y \in \bar{M}} \frac{|D^\beta u(x) - D^\beta u(y)|}{|x - y|^\alpha}.$$

Ο $L^P(M), p \in [1, \infty)$ ορίζεται ως τον γραμμικό χώρο Banach των $p-$οστών παραγωγίσιμων συναρτήσεων στο M και $L^\infty(M)$ ορίζεται ως τον γραμμικό χώρο Banach χώρο των ουσιωδώς φραγμένων συναρτήσεων με νόρμες

$$\|u\|_{p,M} = \left(\int_M |u(x)^p| \, dx \right)^{\frac{1}{p}}$$

και

$$\|u\|_{\infty,M} = \text{ess} \sup_{x \in M} |u(x)|.$$

Για $p = 2$ ο χώρος $L^2(M)$ είναι χώρος Hilbert με εσωτερικό γινόμενο

$$(u, v)_{0,M} = \int_M uv \, dx.$$

Οι χώροι Sobolev βασίζονται στην έννοια των ασθενών παραγώγων.

Ορισμός 5.3. Έστω $u \in L^1(M)$ και $\alpha \in \mathbb{N}_0^d$. Η συνάρτηση u λέμε ότι έχει ασθενή παράγωγο $D_w^\alpha u$ αν υπάρχει μια συνάρτηση $v \in L^1(M)$ έτσι ώστε

$$\int_M u D^\alpha \phi dx = (-1)^{|\alpha|} \int_M v \phi \, dx \quad , \quad \phi \in C_0^\infty(M)$$

Μετά λέμε ότι $D_w^\alpha u := v$.

Παράδειγμα 10. Έστω $d = 1$ και $M = (-1, +1)$. Η συνάρτηση $u(x) = |x|, x \in M$ είναι μη παραγωγίσιμη. Όμως έχει ασθενή παράγωγο $D_w^1 u$ που είναι

$$D_w^1 u = \begin{cases} -1 & , \quad x < 0 \\ +1 & , \quad x > 0 \end{cases}.$$

Για $\phi \in C_0^\infty(M)$ από ολοκλήρωση έχουμε

$$\int_{-1}^1 u(x) D^1 \phi(x) dx = \int_{-1}^0 u(x) D^1 \phi(x) dx + \int_0^1 u(x) D^1 \phi(x) dx$$

$$= -\int_{-1}^{0} D_w^1 u(x)\phi(x)dx + u\phi|_{-1}^{0} - \int_{0}^{1} D_w^1 u(x)\phi(x)dx + u\phi|_{0}^{1} =$$

$$\int_{-1}^{1} D_w^1 u(x)\phi(x)dx - [u(0)]\phi(0)$$

όπου $[u(0)] := u(+0) - u(-0)$ είναι άλματα του u στο $x = 0$. Αλλά u είναι συνεχής άρα $[u(0)] = 0$.

Ορισμός 5.4. Ένα γραμμικός χώρος $H^{m,p}(M), p \in [1, \infty]$ όπου

$$W^{m,p}(M) := \{u \in L^p(M) \mid D_w^a u \in L^p(M)\ ,\ |\alpha| \leq m\}$$

λέγεται χώρος Sobolev. Είναι ένας χώρος Banach με νόρμα

$$\|u\|_{m,p,M} = \left(\sum_{|\alpha| \leq m} \|D_w^\alpha v\|_{p,M}^p \right)^{\frac{1}{p}}, \quad p \in [1, \infty)$$

και

$$\|u\|_{m,\infty,M} = \max_{|\alpha| \leq m} \|D_w^\alpha v\|_{\infty,M}.$$

Επομένως για χωρία(domains) στο $\mathbb{R}^n$ μπορούμε να ορίσουμε το sobolev εσωτερικό γινόμενο ανάμεσα σε συναρτήσεις $\phi, \psi \in C^\infty(M)$

$$\langle \Delta_g \phi, \psi \rangle_{H^1} = \int_M \langle \nabla_g \phi, \nabla_g \psi \rangle_g + \int_M \phi\psi du_g.$$

Θεώρημα 5.5. (Ενσφηνώσεις Sobolev)
Έστω $M \subset \mathbb{R}^d$ είναι χωρίο(domain) Lipschitz, $m \in \mathbb{N}_0$ και $p \in [1, \infty]$. Υπάρχουν οι παρακάτω απεικονίσεις που είναι συνεχείς ενσφηνώσεις(continuous imbeddings)

$$W^{m,p}(M) \to L^{p^*}(M), \quad \frac{1}{p^*} = \frac{1}{p} - \frac{m}{d} \text{ αν } m < \frac{d}{p},$$

$$W^{m,p}(M) \to L^q(M), \quad q \in [1, \infty) \text{ αν } m = \frac{d}{p},$$

$$W^{m,p}(M) \to C^{0,m-\frac{d}{p}}(\bar{M}), \quad \text{αν } \frac{d}{p} < m < \frac{d}{p} + 1,$$

$$W^{m,p}(M) \to C^{0,\alpha}(\bar{M}),\ 0 < \alpha < 1 \quad \text{αν } m = \frac{d}{p} + 1,$$

$$W^{m,p}(M) \to C^{0,1}(\bar{M}), \quad \text{αν } m > \frac{d}{p} + 1.$$

Θεώρημα 5.6. (Ενσφηνώσεις Kondrasov)
Έστω $M \subset \mathbb{R}^d$ είναι χωρίο(domain) Lipschitz, $m \in \mathbb{N}_0$ και $p \in [1, \infty]$. Υπάρχουν οι παρακάτω απεικονίσεις που που οι ακόλουθες ενσφηνώσεις είναι συμπαγείς(compact imbeddings)

$$W^{m,p}(M) \to L^q(M),\ 1 \leq q \leq p^*,\ \frac{1}{p^*} = \frac{1}{p} - \frac{m}{d},\ \text{αν } m < \frac{d}{p},$$

$$W^{m,p}(M) \to L^q(M), \quad q \in [1, \infty) \text{ αν } m = \frac{d}{p},$$

$$W^{m,p}(M) \to C^0(\bar{M}), \quad \text{αν } m > \frac{d}{p}.$$

5.2 Ιδιοσυναρτήσεις και ιδιοτιμές

Η ιδιοσυνάρτηση ενός τελεστή Q είναι μία συνάρτηση f έτσι ώστε η εφαρμογή του τελεστή Q πάνω στην f μας δίνει την f επί κάποιας σταθεράς

$$Qf = kf$$

όπου k είναι σταθερά και είναι η ιδιοτιμή της ιδιοσυνάρτησης.

Παράδειγμα 11. $f(x) = e^x$ με Q να είναι η παράγωγος.

Για αρχή αυτό που κάνουμε είναι να εφαρμόσουμε το Q στην f. Η παράγωγος του e^x είναι πάλι e^x

$$f^{'}(x) = e^x.$$

Άρα βλέπουμε ότι $f^{'}(x) = kf(x)$ όπου $k = 1$. Επομένως η f είναι ιδιοσυνάρτηση με ιδιοτιμή το $k = 1$.

Παράδειγμα 12. $f(x) = e^{x^2}$ με Q να είναι η παράγωγος.

Η παράγωγος του e^x είναι $2xe^{x^2}$

$$f^{'}(x) = 2xe^{x^2}.$$

Παρατηρούμε ότι η παραπάνω συνάρτηση δεν είναι ιδιοσυνάρτηση διότι $f^{'}(x) \neq kf(x)$.

Παράδειγμα 13. Ιδιοσυναρτήσεις και ιδιοτιμές για κύκλο

Οι ιδιοσυναρτήσεις είναι

$$1, \quad \cos(\theta), \quad \sin(\theta), \quad \cos(2\theta), \quad \sin(2\theta), ..., \quad \cos(k\theta), \quad \sin(k\theta), ...$$

με ιδιοτιμές $0, \quad 1, \quad 1, \quad 4, \quad 4, ..., \quad k^2, \quad k^2, ...$ αντίστοιχα.

5.3 Φασματικό θεώρημα

Έστω $\alpha : H \times H \to \mathbb{R}$ είναι μία διγραμμική μορφή και υπάρχουν σταθερές $\beta, \rho > 0$ έτσι ώστε

$$|\alpha(u, v)| \leq \beta \|u\| \|v\|, \quad \forall u, v \in H \quad \text{(boundedness)}$$

$$\alpha(u, v) \geq \rho \|u\|^2, \quad \forall u \in H \quad \text{(coercivity)}.$$

Για κάθε $f \in H^*$, υπάρχει μοναδικό $u \in H$ έτσι ώστε

$$\alpha(u, v) = \langle f, v \rangle, \quad \forall\ v \in H.$$

Ας θεωρήσουμε τώρα την διγραμμική μορφή

$$\alpha(\phi, \psi) = \langle\, \phi, \psi \rangle_{H^1}.$$

Επίσης έχουμε ότι:

$$\alpha(\phi, \phi) = \|\phi\|^2_{H^1} \geq 0.$$

Η διγραμμική μορφή (bilinear form) α είναι συνεχείς και συμμετρική στο $H^1(M)$. Τότε υπάρχει βάση $\{\phi_j\}$ του $L^2(M)$ και ιδιοτιμές $\{\gamma_i\}$ που ικανοποιούν ότι

$$\alpha(\phi_j, \psi) = \gamma_j \langle\, \phi_j, \psi \rangle_{L^2}$$

για κάθε $\psi \in H^1(M)$. Αφού

$$\alpha(\phi_j, \psi) = \int_M \langle\, \nabla_g \phi_j, \nabla_g \psi \rangle_g \, du_g + \langle\, \phi_j, \psi \rangle_{L^2}$$

για όλα $\psi \in H^1(M)$. Αφού $\alpha(\phi_j, \psi) = \int_M \langle\, \nabla_g \phi, \nabla_g \psi \rangle_g du_g + \langle \phi_j, \psi \rangle_{L^2}$ έχουμε

$$\int_M \langle\, \nabla_g \phi_j, \nabla_g \psi \rangle_g \, du_g = (\gamma_j + 1) + \langle\, \phi_j, \psi \rangle_{L^2}$$

για κάθε $\psi \in H^1(M)$. Συγκεκριμένα

$$\langle \Delta_g \phi_j, \nabla_g \psi \rangle_{L^2} = (\gamma_j + 1) \langle \phi_j, \psi \rangle_{L^2}$$

για όλα τα $\psi \in H^1(M)$, άρα

$$-\Delta_g \phi_j = \lambda_j \phi_j$$

για $\lambda_j = \gamma_j + 1$.

Θεώρημα 5.7. Για (M, g) συμπαγές χωρίς σύνορο(boundary), υπάρχει ορθοκανονική βάση$\{\phi_1, \phi_2, ...\}$ του $L^2(M)$ που αποτελείται από ιδιοσυναρτήσεις του Δ_g με ϕ_j έχοντας ιδιοτιμές λ_j όπου

$$0 = \lambda_1 \leq \lambda_2 \leq ... \to \infty.$$

Ορισμός 5.8. Άν μας ενδιαφέρει να καταλάβουμε πως διαδίδεται η θερμότητα στο $\Omega \subset \mathbb{R}^n$ τότε μπορούμε να λύσουμε την εξίσωση θερμότητας

$$\Delta u(x,t) = \frac{1}{c} \frac{\partial}{\partial t} u(x,t)$$

όπου το c είναι η αγωγιμότητα του υλικού από το οποίο είναι κατασκευασμένο το Ω και $u(x,t)$ είναι η θερμοκρασία στο σημείο $x \in \Omega$ σε χρόνο t.

Αφού η λύσεις της εξίσωσης θερμότητας είναι της μορφής $u(x,t) = \sum_i a_i e^{-\lambda_i t} \phi_i(x)$(όπως θα δούμε παρακάτω) τότε είναι φανερό ότι τα λ_1, ϕ_1 δίνουν την μεγαλύτερη πληροφορία δίοτι το $e^{-\lambda_1 t} \phi_1(x)$ μειώνεται με τον πιο αργό ρυθμό καθώς περνά ο χρόνος. Επομένως η γεωμετρία του Ω πρεπεί να είναι στο λ_1 σε κάποιο βαθμό. Για παράδειγμα όσο μεγαλύτερο σύνορο $\partial\Omega$ είναι, τόσο πιο γρήγορα μειώνεται η θερμότητα. Έτσι αν έχουμε ένα χωρίο(domain) Ω και μία μπάλα B με το ίδιο εμβαδό τότε περιμένουμε η θερμότητα στο Ω να διαχυθεί πιο γρήγορα από αυτή του B. Έτσι έχουμε ότι

$$\lambda_1(\Omega) \geq \lambda_1(B)$$

όπου αυτή είναι η ανισότητα Faber-Krahn(1923).

5.4 Heat equation

Έχοντας ότι (M, g) είναι συμπαγές Riemannian manifold χωρίς σύνορο. Ο Heat operator είναι

$$-\Delta_g + \partial_t$$

που επιδρά σε συναρτήσεις στο $C(M \times (0, \infty))$ όπου είναι C^2 στο M και C^1 στο $(0, \infty)$.

Το ομογενές heat equation είναι

$$\begin{cases} (-\Delta_g + \partial_t) u(x,t) = 0 & (x,t) \in M \times (0, +\infty) \\ u(x,0) = f(x) & x \in M \end{cases}.$$

Η συνάρτηση $u(x,t)$ μας δίνει την θερμοκρασία στο σημείο x και σε χρόνο t υποθέτοντας ότι η αρχική θερμοκρασία στο manifold δίνεται από την συνάρτηση $f(x)$.

Το φάσμα δεν καθορίζει γενικά τη γεωμετρία ενός Manifold. Ωστόσο, κάποιες πληροφορίες μπορούν να εξαχθούν από το φάσμα.

Ορισμός 5.9. Έστω M ένα Riemannian Manifold . Το Heat Kernel, ή αλλιώς το fundamental solution για την εξίσωση θερμότητας είναι μία συνάρτηση $K(0, \infty) \times M \times M \to M$ όπου ικανοποιεί τα εξής

- $K(t, x, y)$ είναι C^2 στο x, y και C^1 στο t
- Έχουμε ότι $\frac{\partial K}{\partial t} + \Delta_2(K) = 0$, όπου Δ_2 είναι η λαπλασιανή συναρτήση της δεύτερης μεταβλητής

- $\lim_{t\to 0^+} \int_M K(t,x,y)\,dy = f(x)$ για κάθε συμπαγές συνάρτηση f στο M.

Ξέρουμε ότι ο τελεστής Laplace-Beltrami είναι

$$\Delta_g = \mathrm{div}_g \nabla_g$$

Έτσι η εξίσωση θερμότητας είναι

$$\Delta_g u = \gamma_h \partial_t u$$

με σταθερά το γ_h. Η λύση μπορεί να γραφτεί με την βοήθεια του φάσματος του Δ_g:

$$u(x,t) = \sum_i \alpha_i e^{-\lambda_i t} \phi_i(x)$$

όπου $\Delta_g \phi_l = -\lambda_l \phi_l$ είναι η εξίσωση Helmholtz στο M και τα α_i εξαρτώνται από τις συνθήκες της εξίσωσης θερμότητας.

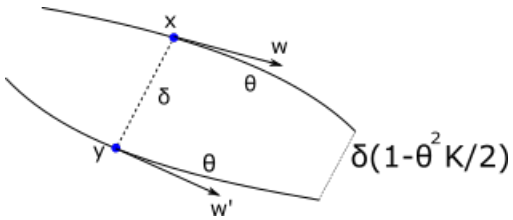
Επίσης μπορούμε να υπολογίσουμε τα kernels των μερικών διαφορικών εξισώσεων ($K_t(x,y,t)$) έτσι ώστε

$$f(x,y) = \int_M K(x,y,t) f(y,0)\,dy$$

είναι λύση της εξίσωσης θερμότητας. Αν λ_i είναι το φάσμα του M και ξ_i είναι οι ιδιοσυναρτήσεις όπου είναι κανονικές και σχηματίζουν μία ορθοκανονική βάση του $L^2(M)$, τότε μπορούμε να πούμε ότι

$$K(t,x,y) = \sum_i e^{-\lambda_i t} \xi_i(x) \xi_i(y).$$

Έστω ότι $x \in M$ και έστω u είναι το εφαπτόμενο διάνυσμα στο x με νόρμα $\delta = \|u\|$. Το y είναι το τελευταίο σημείο του u και $d(x,y) = \delta$. Το w είναι ένα άλλο εφαπτόμενο διάνυσμα στο x με νόρμα θ (το w είναι ορθογώνιο του u). Το $w^{'}$ είναι το ίδιο με το w αλλά στο σημείο y. Εδώ θέλουμε να



Σχήμα 9

βρούμε την απόσταση μεταξύ των τελευταίων σημείων του w και $w^{'}$. Αν είμασταν στον ευκλείδειο χώρο τότε η απόσταση θα ήταν ίση με το δ. Έχουμε ότι

$$d(\exp_x \theta w, \exp_y \theta w^{'}) = \delta \left(1 - \frac{\theta^2}{2} K(u,w) + o(\theta^3 + \theta^2 \delta)\right)$$

όσο $(\theta, \delta) \to 0$. Το $K(u,w)$ είναι η τμηματική καμπυλότηταα (sectional curvature) με κατεύθυνση (u,w).

Ορισμός 5.10. Έστω ότι έχουμε ένα λείο $N-$διάστατο Riemannian Manifold και x ένα σημείο που ανήκει σε αυτό. Το u είναι το εφαπτόμενο διάνυσμα στο x. Ορίζουμε $Ric(u)$(ή $Ric(u,u)$) την καμπυλότητα Ricci στο u ως N Φορές την μέση τιμή του $K(u,w)$ όπου w βρίσκεται στην μοναδιαία σφαίρα στο εφαπτόμενο επίπεδο του x.

Αν S_x είναι το σύνολο των σημείων μιας σφαίρας ακτίνας θ στο εφαπτόμενο επίπεδο στο x και S_y είναι το σύνολο των σημείων μιας σφαίρας ακτίνας θ στο εφαπτόμενο επίπεδο στο y. Αν απεικονίσουμε το S_x στο S_y χρησιμοποιώντας παράλληλη μεταφορά(parallel transport), κάτα μέσο όρο η απόσταση που διανοίουν τα σημεία είναι

$$\delta \left(1 - \frac{\theta^2}{2N} Ric(u) + o(\theta^3 + \theta^2 \delta)\right) \text{ όσο } (\theta, \delta) \to 0$$

Αν χρησιμοποιήσουμε μπάλες αντί για σφαίρες τότε ο όρος $\frac{\theta^2}{2N}$ θα γίνει $\frac{\theta^2}{2(N+2)}$.

Ορισμός 5.11. Ένας τοπολογικός χώρος X είναι:

- Πλήρως μετρικοποιήσιμους (completely metrizable) αν υπάρχει μια μετρική d που ορίζει την τοπολογία του X έτσι ώστε (X, d) είναι πλήρες
- Polish αν είναι διαχωρίσιμος και πλήρως μετρικοποιήσιμος.

Έστω (X, d) ένας Polish μετρικός χώρος. Ο τυχαίος περίπατος m στο X είναι η οικογένεια από πιθανότητες m_x στο X για κάθε $x \in X$ όπου ικανοποιούν τα παρακάτω

1. το m_x εξαρτάται από το σημείο $x \in X$
2. για κάποια $z \in X$ και για κάθε $x \in X$ έχουμε $\int_x d(x, y) dm_x(y) < +\infty$

Η Μαρκοβιανή αλυσίδα κάνει άλματα από το σημείο $x \in X$ σε ένα τυχαίο σημείο που το διαλέγουμε ανάλογα με το m_x. Η πιθανότητα μετάβασης της Μαρκοβιανής αλυσίδας από το x στο y στο βήμα n είναι

$$dm_x^{*n}(y) = \int_{z \in X} dm_z(y) dm_x^{*(n-1)}(z)$$

όπου $m_x^{*1} = m_x$. Έχουμε ότι

$$\int_{y \in X} dm_x^{*n}(y) = \int_{z \in X} \left(\int_{y \in X} dm_z(y) \right) dm_x^{*(n-1)}(z) = \int_{z \in X} dm_x^{*(n-1)}(z) = 1.$$

Δίνοντας μια αρχική κατανομή μ στο X, το μέτρο $\mu * m$ είναι

$$d(\mu * m)(y) = \int_X dm_x(y) d\mu(x)$$

όπου είναι η νέα κατανομή μετά από το άλμα. Ο τελεστής Laplace σε σχέση με το m (Δ_m) ορίζεται ως

$$\Delta_m f(x) = \int_X (f(y) - f(x))\, dm_x(y).$$

Το μέτρο Radon v στο X είναι αναλοίωτο[8] για τον τυχαίο περίπατο αν

$$dv(x) = \int_X dv(y) dm_y(x),$$

άρα για κάθε $v-$μετρήσιμο σύνολο A έχουμε ότι το A είναι m_x-μετρήσιμο για σχεδόν όλα τα $x \in X$ και $x \mapsto m_x(A)$ είναι $v-$μετρήσιμο και

$$v(A) = \int_X m_x(A) dv(x).$$

Δυϊκά, η Μαρκοβιανή αλυσίδα ορίζει έναν τελεστή M σε μια φραγμένη συνάρτηση στο X από

$$(Mf)(x) = \int_{y \in X} f(y) dm_x(y)$$

και βλέπουμε $\int Mf\, d\mu = \int f d(\mu * m)$. Εδώ το $M^n f(x)$ είναι η αναμενόμενη τιμή του f στο τελικό σημείο του τυχαίου περιπάτου με $n-$βήματα. Τώρα αν η f είναι σταθερή τότε $Mf = f$. Εμείς θα πάρουμε το v ως ένα αναλοίωτο μέτρο πιθανότητας ($v(X) = 1$).

Γενικά το κριτήριο το οποίο χρησιμοποιούμε για να συγκρίνουμε μέτρα πιθανότητας είναι το total variation distance

$$\|\mu - \mu'\|_\tau = \sup_{A \subset X} |\mu(A) - \mu'(A)| = \frac{1}{2} \int_X d|\mu - \mu'|$$

Όταν έχουμε συνεχή χρόνο ο τυχαίος περίπατος έχει πιθανότητα dt να κάνει άλμα από ένα σημείο x σε ένα νέο σημείο που πάρθηκε με την βοήθεια του m_x. Σε αυτήν την περίπτωση αντί για το M θα έχουμε

$$M_t = \lim((1 - \theta) Id + \theta M)^{\frac{t}{\theta}} = e^{t(M - Id)}$$

[8] Ένα μέτρο v είναι αναλοίωτο αν $v * m = v$

όπου συγκλίνει ως τελεστής του $L^2(v)$ αφού $\|M\| < 1$. Στον συνεχή χρόνο οι πιθανότητες μετάβασης σε χρόνο t είναι

$$dp_x^t(y) = \sum_{k\in\mathbb{N}} e^{-t}\frac{t^k}{k!}dm_x^{*t}(y)$$

Ο τελεστής Laplace είναι

$$\Delta = M - Id$$

Εδώ θα δουλέψουμε με τον τελεστή νόρμας του $e^{t\Delta}$ στο $L_0^2(v)$

$$\frac{d}{dt}|_x Var_v(e^{t\Delta}f) = 2\langle \nabla f, \Delta f\rangle_{L_0^2(v)} = -\int (f(y) - f(x))^2\, dv(x)dm_x(y)$$

όπου το τελευταίος όρος λέγεται μορφή Dirichlet.

Ανισότητα Poincaré: Η ανισότητα $Var_v(e^{t\Delta}f) \leq e^{-2\lambda t}Var_v f$ ισχύει για κάθε $f \in L^2(v)$ αν και μόνο αν

$$Var_v f \leq \frac{1}{2\lambda}\int\int (f(y) - f(x))^2\, dv(x)dm_x(y)$$

όπου ισχύει για κάθε συνάρτηση $f \in L^2(v)$.

Για κάθε $x \in X$ με $v(x) > 0$ έχουμε

$$\|p_x^t - v\|_\tau \leq \frac{1}{2}\sqrt{\frac{1 - v(x)}{v(x)}}e^{-\lambda t}$$

Η καλύτερη τιμή του λ λέγεται φασματικό χάσμα(spectral gap).

Αν X είναι ένα Riemannian Manifold και το m_x να είναι το μέτρο περιορισμένο στην μπάλα με ακτίνα θ γύρω από το X. Ο διακριτός τελεστής Laplace του τυχαίου περίπατου θα είναι μια προσέγγιση του τελεστή Laplace-Beltrami πολλαπλασιασμένος με τον όρο $\frac{\theta^2}{2(N+2)}$. Έτσι ο τυχαίος περίπατος θα προσεγγίσει μία κίνηση Brown. Έχοντας μία λεία συνάρτηση f όπου $\int(f(y) - f(x))^2$ και $Var_{m_x}f$ προσεγγίζουνε το $\|\nabla f\|^2$ και η ανισότητα Poincaré είναι

$$Var f \leq \frac{1}{2}\int \|\nabla f\|^2.$$

Η λαπλασιανή ενός γραφήματος(graph) είναι ανάλογη με τον τελεστή Laplace-Beltrami στο Manifold. Έστω M είναι ένα $m-$διάστατο Riemannian Manifold. Το manifold αυτό θέλουμε να το πάμε στον $\mathbb{R}^n$. Έστω μια απεικόνιση $f : M \to \mathbb{R}$ που είναι παραγωγίσιμη. Έστω δύο γειτονικά σημεία $x, y \in M$.

Η βαθμίδα $\nabla f(x)$ είναι ένα διάνυσμα του εφαπτόμενου χώρου T_xM έτσι ώστε δοθέντος ενός άλλου διανύσματος $z \in T_xM$, $df(z) = \langle \nabla f(x), z\rangle_M$. Έστω $l = dist_M(x, y)$. Το $\gamma(t)$ είναι η γεωδαισιακή με $x = \gamma(0)$ και $y = \gamma(l)$. Τότε

$$f(y) = f(x) + \int_0^l df(\gamma^{'}(t))\, dt = f(x) + \int_0^l \langle \nabla f(\gamma(t)), \gamma^{'}(t)\rangle\, dt.$$

Από την γενίκευση της ανισότητας Caushy-Schwarz έχουμε

$$\langle \nabla f(\gamma(t)), \gamma^{'}(t)\rangle \leq \|\nabla f(\gamma(t))\| \left\|\gamma^{'}(t)\right\| = \|\nabla f(\gamma(t))\|$$

αφού $\gamma(t)$ είναι η γεωδαισιακή τότε $\left\|\gamma^{'}(t)\right\| = 1$. Επίσης έχουμε ότι

$$\|\nabla f(\gamma(t))\| = \|\nabla f(x)\| + o(t).$$

Ολοκληρώνοντας έχουμε

$$|f(y) - f(x)| \leq l\,\|\nabla f(x)\| + o(l).$$

Το M είναι ισομετρικά εμβαπτισμένο στον $\mathbb{R}^l$ τότε

$$dist_M(x,y) = \|x-y\|_{\mathbb{R}^l} + o(\|x-y\|_{\mathbb{R}^l})$$

και

$$|f(y)-f(x)| \leq \|\nabla f(x)\| \, \|y-x\| + o(\|x-y\|)$$

Από το παραπάνω καταλαβαίνουμε ότι το $\|\nabla f(x)\|$ μας δίνει το πόσο μακριά η f απεικονίζει τα γειτονικά σημεία.
Επομένως θέλουμε να ελαχιστοποιήσουμε

$$\underset{\|f\|_{L^2(M)}=1}{\arg\min} \int_M \|\nabla f(x)\|^2 .$$

Ελαχιστοποιώντας το $\int_M \|\nabla f(x)\|^2$ σε ένα manifold είναι σαν την ελαχιστοποίηση $\frac{1}{2}\sum_{u_i \in A, u_j \in \bar{A}} w(u_i,u_j)(f_i - f_j)^2$ σε ένα γράφημα(graph). Επίσης έχουμε ότι

$$\int_M \langle X, \nabla f \rangle = \int_M div(X) f.$$

άρα

$$\int_M \|\nabla f(x)\|^2 = \int_M \Delta(f) f.$$

Η f που ελαχιστοποιεί το $\int_M \|\nabla f(x)\|^2$ θα είναι μια ιδιοσυνάρτηση του Δ. Όπως και με το γράφημα(graph) έτσι και εδώ έχουμε ιδιοτιμές $\lambda_0 \leq \lambda_1 \leq \lambda_2 \leq ...$, και τα f_i θα είναι η ιδιοσυναρτήσεις για κάθε ιδιοτιμή. Πάλι όπως και με την περίπτωση με το γράφημα έτσι και εδώ θα απορρίψουμε την ιδιοσυνάρτηση που αντιστοιχεί στην ιδιοτιμή με τιμή μηδέν και θα πάρουμε την ιδιοσυνάρτηση της αμέσως επόμενης ελάχιστης ιδιοτιμής.

6 Γραμμικές τεχνικές μείωσης διαστάσεων

6.1 Principal Component Analysis

Η μέθοδος Principal component analysis είναι μια αρκετά παλιά μέθοδος (1901) αλλά αποτελεί την βάση κάποιων άλλων μεθόδων που θα καλύψουμε παρακάτω.
Το PCA ειναι μία μέθοδος για μείωση της διάστασης. Άρα χρησιμοποιόντας PCA μπορούμε να μειώσουμε την διάσταση των δεδομένων από έναν υψηλής διάστασης χώρο σε έναν χώρα χαμηλότερης διάστασης χωρίς να χάνουμε πολύ πληροφορία.

Έχοντας κάποια δεδομένα σε d-διάστατο χώρο μπορούμε να μετασχηματίσουμε τα δεδομένα σε έναν p-διάστατο χώρο όπου $p \leq d$.

Έστω ότι έχω n-data points $X = [x_1 ... \ x_n]_{d \times n}$ και $x_i \in \mathbb{R}^d$. Αυτό που θέλουμε να κάνουμε είναι να προβάλουμε αυτά τα d σημεία σε ένα διάνυσμα u_1. Έχουμε (το $u^\top$ είναι $1 \times n$ διάνυσμα)

$$u_1^\top X$$

έτσι ώστε το $Var(u_1^\top X)$ είναι μέγιστο. Συγκεκριμένα θα έχουμε ότι (S είναι $d \times d$ είναι το covariance matrix του δείγματος)

$$\max_{u_1} Var(u_1^T X) = \max_{u_1} u_1^T S u_1$$

Άρα θα χρειαστεί να βρούμε το μέγιστο του $u_1^\top S u_1$. Όμως το $u_1^\top S u_1$ είναι δευτεροβάθμια (δεν έχει πάνω όριο) άρα δεν γίνεται να βρεθεί το μέγιστο. Επομένως θα βάλουμε κάποιον περιορισμό. Ο περιορισμός είναι $u_1^\top u_1 = 1$. Άρα θα έχουμε

$$\max_{u_1} u_1^\top S u_1 \ \text{ με } \ u_1^\top u_1 = 1.$$

Για το παραπάνω μπορούμε να χρησιμοποιήσουμε τον πολλαπλασιαστή Lagrange

$$L(u_1, \lambda_1) = u_1^\top S u_1 - \lambda_1 (u_1^\top u_1 - 1)$$

και να βρούμε το σαγματικό σημείο

$$\frac{\partial L}{\partial u_1} = 2Su_1 - 2\lambda_1 u_1 = 0 \Rightarrow Su_1 = \lambda_1 u_1.$$

Επομένως το u_1 είναι ιδιοδιάνυσμα του S και το λ_1 είναι ιδιοτιμή του S. Από αυτό καταλαβαίνουμε ότι τα ιδιοδιανύσματα και οι ιδιοτιμές του S θα είναι σαγματικά σημεία της Langrangian ($\frac{\partial L}{\partial u_1} = 0$). Άρα κάθε ζευγάρι $(u_1, \lambda_1), (u_2, \lambda_2), ..., (u_d, \lambda_d)$ είναι σαγματικά σημεία. Ποιο όμως μεγιστοποιεί το objective function;

Λέμε ότι

$$u^\top S u = u^\top \lambda u = \lambda u^\top u = \lambda.$$

Από το παραπάνω καταλαβαίνουμε ότι αν εισάγω οποιοδήποτε ιδιοδιάνυσμα του πίνακα S στο $\max_{u_1} u_1^T S u_1$, τότε η τιμή που θα πάρουμε θα είναι η ιδιοτιμή που αντιστοιχεί στο ιδιοδιάνυσμα. Επομένως για να το μεγιστοποιήσουμε θα επιλέξουμε το ιδιοδιάνυσμα που αντιστοιχεί στην μεγαλύτερη ιδιοτιμή.

Ξέρουμε ότι το S έχει το πολύ d ιδιοτιμές και d ιδιοδιανύσματα

$$\lambda_1 > \lambda_2 > \lambda_3 > ... > \lambda_d$$

$$u_1 \ u_2 \ u_3 ... u_d.$$

Επομένως το ιδιοδιάνυσμα του πίνακας συνδιακύμανσης S που αντιστοιχεί στην μέγιστη ιδιοτιμή θα είναι η 1η κύρια συνιστώσα (Principal component).

Έστω $X_{d \times n}$ πίνακας, έχουμε (singular value decomposition)

$$X = U \Sigma V^\top$$

όπου το $U_{d\times d}$ είναι ιδιοδιανύσματα του $XX^\top$. Το $V_{n\times n}$ είναι ιδιοδιανύσματα το $X^\top X$ και το $\Sigma_{d\times n}$ είναι ένας διαγώνιος πίνακας όπου τα διαγώνια στοιχεία είναι ιδιοτιμές του $XX^\top$ ή του $X^\top X$.
Ας υποθέσουμε ότι το X είναι ένας πίνακας δεδομένων. Αν κεντροποιήσουμε το X

$$\tilde{X} = (X_{d\times n} - M_{d\times n})$$

όπου $X = [x_1...x_n]$, $M = [\bar{x}\ \bar{x}...\bar{x}]$ και $\bar{x} = \frac{1}{d}\sum_{i=1}^{d} x_i$, το $\tilde{X}\tilde{X}^\top$ θα είναι το covariance matrix του X.

Αν κάνουμε singular value decomposition στο $\tilde{X}$ θα έχουμε

$$\tilde{X} = U\Sigma V^\top$$

όπου το U είναι ιδιοδιανύσματα του $\tilde{X}\tilde{X}^\top$ και το V είναι ιδιοδιανύσματα του $\tilde{X}^\top\tilde{X}$.

Αλγόριθμος Principal Component Analysis:

Input:

- Πίνακας δεδομένων $X_{d\times n}$ όπου στις γραμμές βρίσκονται οι μεταβλητές
- Τον αριθμό των διαστάσεων που θα κρατήσουμε (έστω p)

1. Βρίσκουμε την μέση τιμή όλων των x_i: $\mu = \frac{1}{d}\sum_{i=1}^{d} x_i$
2. $D = \sum_{i=1}^{d} x_i - \mu$
3. Υπολογίζουμε το $Z = \frac{1}{\sqrt{n-1}}D^\top$ και κάνουμε SVD στον πίνακα Z ($Z = U\Sigma V^\top$) όπου τα διαγώνια στοιχεία του Σ είναι οι τετραγωνικές ρίζες των ταξινομημένων ιδιοτιμών και τα principal components θα είναι οι στήλες του U.
4. Διαλέγουμε τα ιδιοδιανύσματα των μεγαλύτερων ιδιοτιμών $W = [U_1, U_2, ..., U_p]$
5. $Y = W^\top D$.

Output:

- Ο πίνακας Y διαστάσεων $p \times n$.

Μέχρι τώρα έχουμε πει ότι μπορούμε να κάνουμε PCA χρησιμοποιόντας singular value decomposition

$$X = U\Sigma V^\top$$

όπου U είναι ιδιοδιανύσματα του $XX^\top$ και V είναι ιδιοδιανύσματα του $X^\top X$(το $XX^\top$ είναι $d \times d$ πίνακας και το $X^\top X$ είναι $n \times n$).
Υπάρχουν κάποιες περιπτώσεις όπου οι διαστάσεις είναι μεγαλύτερες από τον αριθμό των δεδομένων. Για παράδειγμα μπορεί να έχουμε δεδομένα με γονιδιακές εκφράσεις, όπου έχουμε πολλά γονίδια για κάθε άτομο αλλά έχουμε ένα αρκετά μικρό δείγμα ατόμων. Άρα θα μας είναι δύσκολο (υπολογιστικά) να κάνουμε decomposition στον πίνακα $XX^\top$.

Αυτό που μπορούμε να κάνουμε είναι να γράψουμε το U συναρτήση του X, Σ, V^T

$$X = U\Sigma V^\top$$

Τώρα έστω ότι μειώνουμε τις διαστάσεις(Dimensionality Reduction) του πίνακα U και κρατάμε τα πρώτα p ιδιοδιανύσματα που αντιστοιχούν στις πρώτες p τιμές του πίνακα Σ. Αφού το $V^\top$ είναι ορθοκανονικός πίνακας μπορούμε να πολλαπλασιάσουμε και τις δύο μεριές με V από τα δεξιά. Άρα

$$XV = U\Sigma$$

όπου οι διαστάσεις των πινάκων είναι

$$X_{d\times n}\ \ U_{d\times p}\ \ \Sigma_{p\times p}\ \ V_{n\times p}$$

Ξέρουμε ότι το Σ είναι ένας διαγώνιος πίνακας με όλες τις διαγώνιες τιμές θετικές. Άρα μπορούμε να υπολογίσουμε τον ψευδοαντίστροφο [9] του πίνακα Σ.

$$XV\Sigma^{-1} = U$$

Επομένως μπορούμε να χρησιμοποιήσουμε των παραπάνω αλγόριθμο απλά όπου υπάρχει το U θα έχουμε $XV\Sigma^{-1}$.

6.2 Kernel PCA

Μέχρι τώρα τα πάντα ήταν γραμμικά. Τι γίνεται όταν τα δεδομένα βρίσκονται σε ένα submanifold κάποιου χώρου ;
Σε αυτήν την περίπτωση θα βοηθήσει το Kernel PCA. Πριν όμως μιλήσουμε για Kernel PCA καλό είναι να εξηγήσουμε τι είναι οι μέθοδοι Kernel.

Η ιδέα των μεθόδων Kernel είναι ότι σε υψηλές διαστάσεις η δομή των δεδομένων μπορεί να αναλυθεί πιο εύκολα [10]. Για παράδειγμα έχουμε έναν αλγόριθμο που είναι γραμμικός και δεν μπορούμε (ή δεν θέλουμε) να αλλάξουμε, τότε μπορούμε να αλλάξουμε τα δεδομένα μας έτσι ώστε να μπορέσει να γίνει η ανάλυση πιο εύκολα.
Γενικά όμως σε μεγάλες διαστάσεις υπάρχουν δυσκολίες υπολογιστικά και στατιστικά [11].

Έστω ότι έχουμε ένα feature space $\mathcal{V}$ έτσι ώστε

$$\Phi : X \to \mathcal{V}$$

και

$$X \to \Phi(X)$$

Το X (X =δεδομένα) το απεικονίζουμε σε έναν χώρο έτσι ώστε η διάσταση των δεδομένων στο χώρο αυτό είναι υψηλότερη από αυτού του X.

Ορισμός 6.1. Kernel είναι μία συνάρτηση k όπου για κάθε $x, y \in X$ έχουμε ότι

$$K(x, y) = \langle \Phi(x), \Phi(y) \rangle$$

όπου $\langle ., . \rangle$ είναι το εσωτερικό γινόμενο και Φ είναι η απεικόνιση που μας πάει από το X στο feature space $\mathcal{V}$.

Παράδειγμα 14. Έχω

$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \overset{\Phi}{\longmapsto} \Phi(\underline{x}) = \begin{bmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2}x_1x_2 \end{bmatrix}$ με x_1, x_2 να είναι scalars. Φαίνεται ότι στον 3-διάστατο χώρο τα σημεία γίνονται γραμμικά διαχωρίσιμα. Έτσι μπορούν να διαχωριστούν από ένα γραμμικό υπερεπίπεδο(Σχήμα 10).

Εφαρμόζω αυτήν την απεικόνιση στο y

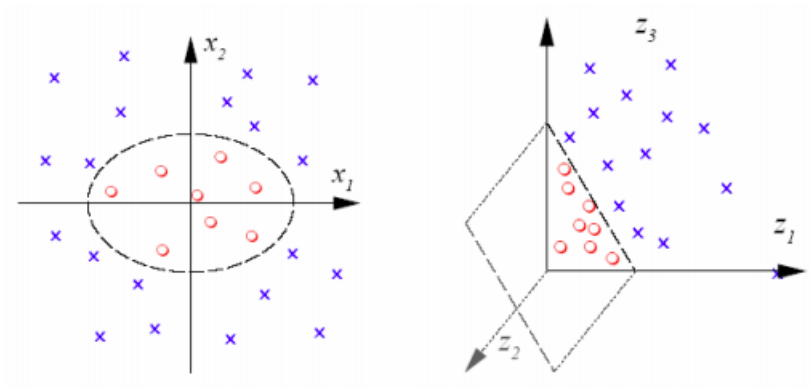
$\underline{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \overset{\Phi}{\longmapsto} \Phi(\underline{z}) = \begin{bmatrix} z_1^2 \\ z_2^2 \\ \sqrt{2}z_1z_2 \end{bmatrix}$

Άρα έχουμε ότι $\Phi(\underline{x})^\top \Phi(\underline{z}) = [x_1^2 \ x_2^2 \ \sqrt{2}x_1x_2] \begin{bmatrix} z_1^2 \\ z_2^2 \\ \sqrt{2}z_1z_2 \end{bmatrix} = x_1^2z_1^2 + x_2^2z_2^2 + 2x_1x_2z_1z_2$

[9] Ο πίνακας Σ από μόνος του δεν είναι αντιστρέψιμος. Ξέρουμε ότι Σ είναι ένας διαγώνιος πίνακας που περιέχει τις τιμές $\lambda_1, ..., \lambda_p$ και $n - p$ φορές την τιμή 0. Επομένως μπορούμε να κάνουμε truncate το πίνακα Σ έτσι ώστε να γίνει τετραγωνικός πίνακας

[10] Blessing of dimensionality

[11] Curse of dimensionality



Σχήμα 10

Η συνάρτηση $K(\underline{x}, \underline{z})$ η οποία θα μας δώσει το ίδιο αποτέλεσμα είναι

$$K(x,y) = (\langle x, y\rangle)^2 = \left([x_1 \ x_2]\begin{bmatrix} z_1 \\ z_2 \end{bmatrix}\right)^2 = (x_1 z_1 + x_2 z_2)^2 = x_1^2 z_1^2 + x_2^2 z_2^2 + 2x_1 z_1 x_2 z_2$$

Άρα βρήκαμε μία συνάρτηση έτσι ώστε αυτή η συνάρτηση μπορεί να υπολογίσει το γινόμενο (dot product) μεταξύ των $\Phi(\underline{x})^\top$ και $\Phi(\underline{z})$.

Υπάρχουν πολλές Kernel συναρτήσεις όπως

1. Linear Kernel
$$k_{ij} = \langle x_i, x_j\rangle$$
2. Gaussian Kernel
$$k_{ij} = \exp^{\frac{-\|x_i - x_j\|^2}{2\sigma^2}}$$
3. Polynomial Kernel
$$k_{ij} = (1 + \langle x_i, x_j\rangle)^p$$

Κάθε Kernel συνάρτηση θα έχει ένα αντίστοιχο Φ. Όταν εφαρμόζουμε Kernel συνάρτηση στα δεδομένα είναι σαν να εφαρμόζουμε το Φ και να υπολογίζουμε το γινόμενο (dot product).

Τώρα θα χρειαστεί ο αλγόριθμος να αλλάξει έτσι ώστε να μην έχουμε προβλήματα υπολογιστικά που προκύπτουν από το dimensionality. Η αλλαγή που πρέπει να γίνει είναι ο αλγόριθμος να εξαρτάται μόνο από το γινόμενο και να μην εξαρτάται από τις συντεταγμένες του κάθε σημείου. Άρα σε αυτόν τον αλγόριθμο θα υπολογίζουμε τα γινόμενα των εικόνων των σημείων x, y μέσω της απεικόνισης που ορίζεται απο την Kernel συνάρτηση[12].

Μέχρι τώρα έχουμε μάθει ότι το PCA γίνεται με ανάλυση του πίνακα X σε ιδιάζουσες τιμές(singular value decomposition στο X), για το Kernel PCA έχουμε

$$\Phi(X) = U\Sigma V^\top$$

όπου U περιέχει τα ιδιοδιανύσματα του $\Phi(X)\Phi(X)^\top$. Έστω ότι $\Phi(X)$ είναι πίνακας διάστασης $t \times n$, τότε οι διάσταση του feature space θα είναι (t)(όπου t πολύ μεγάλο). Αφού το $\Phi(X)$ είναι πίνακας διαστάσεων $t \times n$ τότε ο πίνακας U θα είναι διαστάσεων $t \times t$. Άρα ο αλγόριθμος PCA θα είναι μη πρακτικός.
Για μπορέσουμε να κάνουμε PCA θα χρειαστεί ο πίνακας U να μην εξαρτάται από το t. Έστω ότι έχουμε $K(x,y) = \Phi(x)^\top\Phi(y)$ όπου $K(.,.)$ είναι kernel. Έχοντας μια τέτοια συνάρτηση μπορούμε να υπολογίσουμε τον πίνακας K όπου $K = \Phi(X)^\top\Phi(X)$ και ο K είναι πίνακας διαστάσεων $n \times n$.

[12] Οι χώροι που προβάλουμε τα δεδομένα είναι χώροι Hilbert

Αλγόριθμος Kernel Principal Component Analysis:

Input:

- Πίνακας δεδομένων $X_{d\times n}$ όπου στις γραμμές βρίσκονται οι μεταβλητές
- Τον αριθμό των διαστάσεων που θα κρατήσουμε (έστω p)

1. Διαλέγουμε την συνάρτηση kernel
2. Υπολογίζουμε $K = \Phi(X)^\top \Phi(X)$ χρησιμοποιόντας Kernel k και $V =$ ιδιοδιανύσματα του $\Phi(X)^\top \Phi(X)$ που αντιστοιχούν στις πρώτες p ιδιοτιμές. Έστω $\Sigma =$ διαγώνιος πίνακας που αποτελείται από τετραγωνικές ρίζες των πρώτων p ιδιοτιμών.
3. $Y = U^\top \Phi(X) = \Sigma V^\top$

Output:

- Ο πίνακας Y διαστάσεων $p \times n$.

Τέλος το συμπέρασμα το οποίο προκύπτει είναι ότι ο αλγόριθμος PCA είναι μια ειδική περίπτωση του Kernel PCA με πολυωνυμικό kernel πρώτης τάξης.

6.3 Multidimensional Scaling

Το MDS είναι ακόμη ένας τρόπος που απεικονίζει τον αρχικά μεγάλης διάστασης χώρο σε έναν χαμηλότερης διάστασης με την διατήρηση των pairwise αποστάσεων.

Ορισμός 6.2. Ένας $t \times t$ πίνακας $D^{(X)}$ λέγεται πίνακας συνάφειας(affinity matrix ή distance matrix) αν είναι συμμετρικός, $d_{ii} = 0$ και $d_{ij} > 0,\ i \neq j$.

Δοθέντος ενός πίνακα συνάφειας $D^{(X)}$, το MDS προσπαθεί να βρει t σημεία δεδομένων $y_1, ..., y_t$ σε d διαστάσεις, έτσι ώστε αν $d_{ij}^{(Y)}$ ορίζει την ευκλείδεια απόσταση μεταξύ y_i και y_j, τότε $D^{(Y)}$ είναι παρόμοιο με το $D^{(X)}$. Συγκεκριμένα, θεωρούμε MDS, όπου ελαχιστοποιεί

$$\min_Y \sum_{i=1}^{t} \sum_{i=1}^{t} \left(d_{ij}^{(X)} - d_{ij}^{(Y)} \right)^2$$

όπου $d_{ij}^{(X)} = \|x_i - x_j\|$ και $d_{ij}^{(Y)} = \|y_i - y_j\|$.

Ο πίνακας $D^{(X)}$ μπορεί να μετατραπεί σε kernel πίνακα K

$$K = -\frac{1}{2} H D^{(X)} H$$

όπου $H = I - \frac{1}{t} ee^\top$ και e είναι διάνυσμα στήλη με την τιμή 1.

Θεώρημα 6.3. Έστω D ο πίνακας συνάφειας και ορίζουμε $K = -\frac{1}{2} H D^{(X)} H$, τότε ο D είναι ευκλείδειος πίνακας συνάφειας [13] αν και μόνο αν K είναι θετικά ημι-ορισμένος.

Αφού K είναι θετικά ημι-ορισμένος, μπορεί να γραφτεί ως $K = X^\top X$. Τώρα το

$$\min_Y \sum_{i=1}^{t} \sum_{i=1}^{t} \left(d_{ij}^{(X)} - d_{ij}^{(Y)} \right)^2$$

[13] Euclidean distance matrix

μπορεί να γραφτεί ως

$$\min_{Y} \sum_{i=1}^{t} \sum_{i=1}^{t} \left(x_i^\top x_j - y_i^\top y_j\right)^2$$

Η νόρμα μπορεί να γίνει το trace

$$\min_{Y} Tr(X^\top X - Y^\top Y)^2$$

Κάνοντας SVD(Singular Value Decomposition) στο $X^\top X$ και $Y^\top Y$ έχουμε

$$X^\top X = V \Lambda V^\top$$

$$Y^\top Y = Q \hat{\Lambda} Q^\top$$

Αφού $Y^\top Y$ είναι θετικα ημι-ορισμένος, το $\hat{\Lambda}$ δεν έχει αρνητικές τιμές και έτσι

$$Y = \hat{\Lambda}^{\frac{1}{2}} Q^\top.$$

Οι παραπάνω ορισμοί μας βοηθούν να γράψουμε την cost function ως

$$\min_{Q,\hat{\Lambda}} Tr(V \Lambda V^\top - Q \hat{\Lambda} Q^\top)^2 =$$

$$= \min_{Q,\hat{\Lambda}} Tr(\Lambda - V^\top Q \hat{\Lambda} Q^\top V)^2$$

Έστω $C = V^\top Q$, άρα

$$\min_{Q,\hat{\Lambda}} Tr(\Lambda - V^\top Q \hat{\Lambda} Q^\top V)^2 =$$

$$= \min_{C,\hat{\Lambda}} Tr(\Lambda - C \hat{\Lambda} C^\top)^2$$

Αναλύουμε το τετράγωνο

$$\min_{C,\hat{\Lambda}} Tr(\Lambda^2 + C \hat{\Lambda} C^\top C \hat{\Lambda} C^\top - 2 \Lambda C \hat{\Lambda} C^\top)$$

Έχοντας σταθερό $\hat{\Lambda}$ μπορούμε να ελαχιστοποιήσουμε το C και έχουμε ότι $C = I$.
Επομένως το min θα γινει είναι

$$\min_{\hat{\Lambda}} Tr(\Lambda^2 + \hat{\Lambda} \hat{\Lambda} - 2 \Lambda \hat{\Lambda}) =$$

$$= \min_{\hat{\Lambda}} Tr(\Lambda - \hat{\Lambda})^2$$

Για ελαχιστοποιήσουμε το $Tr(\Lambda - \hat{\Lambda})^2$ θα πρέπει να κάνουμε το $\hat{\Lambda}$ να είναι τα πρώτα d διαγώνια στοιχεία του Λ. Ξέροντας ότι $C = I$ έχουμε

$$I = V^\top Q.$$

Επομένως

$$Y = \hat{\Lambda}^{\frac{1}{2}} V^\top$$

όπου V είναι ιδιοδιανύσματα του $X^\top X$ που αντιστοιχεί στις πρώτες d ιδιοτιμές και $\hat{\Lambda}$ είναι οι πρώτες d ιδιοτιμές του $X^\top X$.

Αλγόριθμος Multidimensional Scaling:

Input:

- Πίνακας δεδομένων $X_{d\times n}$ όπου στις γραμμές βρίσκονται οι μεταβλητές
- Τον αριθμό των διαστάσεων που θα κρατήσουμε (έστω p)

1. Φτιάχνουμε των πίνακα των τετραγώνων των αποστάσεων $D^{(X)}$
2. Υπολογίζουμε τον πίνακα $K = -\frac{1}{2}HD^{(X)}H$ όπου $H = I - \frac{1}{n}ee^\top$
3. Υπολογίζουμε τα ιδιοδιανύσματα (V_i) και τις ιδιοτιμές (λ_i) του πίνακα K.
4. Βάζουμε σε φθίνουσα σειρά τα ιδιοδιανύσματα βάση των ιδιοτιμών
5. Υπολογίζουμε τουν πίνακα $Y = \hat{\Lambda}^{\frac{1}{2}}V^\top$ όπου $\hat{\Lambda}$ διαγώνιος πίνακας με στοιχεία λ_i οι ιδιοτιμές του K και V πίνακας με στήλες τα ιδιοδιανύσματα V_i του K.

Output:

- Ο πίνακας Y διαστάσεων $p \times n$.

7 Μη Γραμμική Μείωση Διαστάσεων

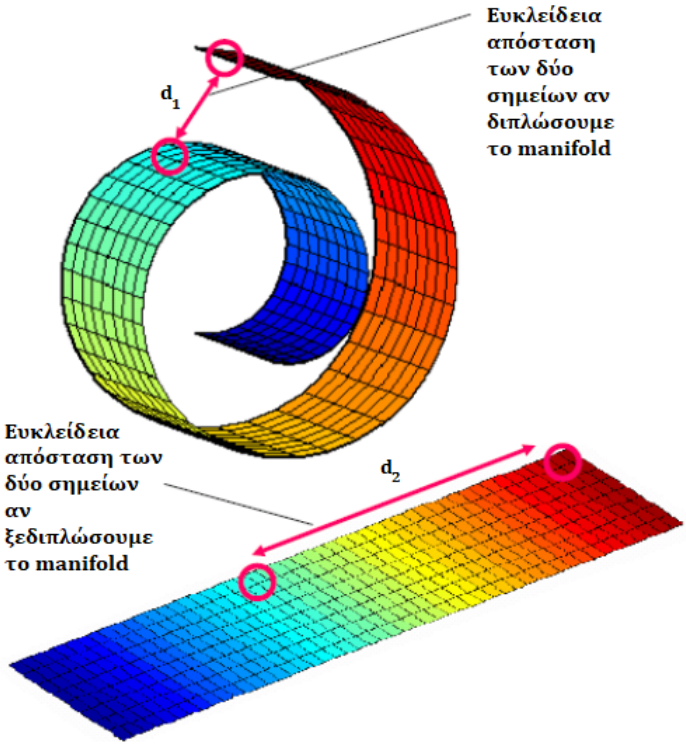
7.1 Isomap

Μέχρι τώρα όλοι οι αλγόριθμοι που έχουμε πει ασχολούνται με δεδομένα που βρίσκονται σε κάποιον υπόχωρο(subspace). Αν τα δεδομένα βρίσκονται σε sub-manifold τότε το MDS δεν είναι ο επιθυμητός αλγόριθμος διότι δεν λαμβάνει υπόψην του την καμπυλότητα και την γεωμετρία των δεδομένων. Το Isomap είναι ίδιο με το MDS αλλά αντί για ευκλείδεια απόσταση χρησιμοποιούμε την γεωδαισιακή απόσταση(geodesic distance) μεταξύ σημείων.

Το ερώτημα είναι γιατί να χρησιμοποιήσουμε γεωδαισιακές αποστάσεις αντί για ευκλείδειες ;

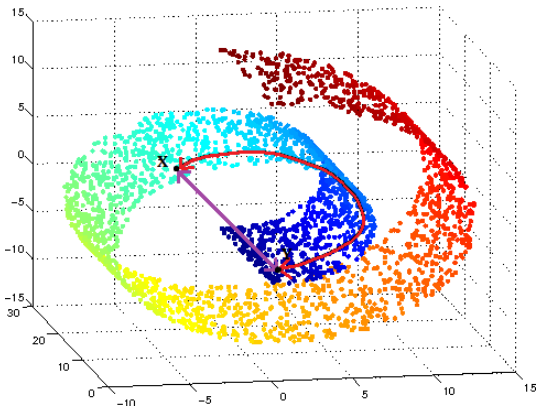
Για να απαντήσουμε σε αυτό το ερώτημα μπορούμε να δώσουμε ένα παράδειγμα.

Παράδειγμα 15.



Σχήμα 11: `https://www.cs.cmu.edu/~bapoczos/Classes/ML10715_2015Fall/slides/ManifoldLearning.pdf`

Στο παράπανω σχήμα βλέπουμε κάποια δεδομένα που έχουν την μορφή ενός Swiss roll. Αν διαλέξουμε 2 σημεία πάνω σε αυτό το manifold και υπολογίσουμε την ευκλείδεια απόσταση μεταξύ τους θα δούμε ότι αυτή η απόσταση αλλάζει άμα ξεδιπλώσουμε το manifold . Αυτό συμβαίνει διότι το Swiss roll είναι ένα μη γραμμικό manifold.



Σχήμα 12: `https://www.semanticscholar.org/paper/Algorithms-for-manifold-learning-Cayton/100dcf6aa83ac559c83518c8a41676b1a3a55fc0`

Παρατηρούμε ότι η γεωδαισιακή απόσταση μεταξύ των σημείων X και Y θα παραμείνει ίδια αν ξεδιπλώσουμε το manifold.

Τώρα το πρόβλημα που προκύπτει είναι το πως μπορούμε να υπολογίσουμε την γεωδαισιακή απόσταση μεταξύ δύο σημείων. Αυτό που μπορούμε να κάνουμε είναι να βρούμε μια εκτίμηση αυτής της απόστασης με την βοήθεια των k κοντινότερων γειτόνων ή διαλέγοντας όλα τα σημεία που βρίσκονται σε ακτίνα ϵ. Αυτές οι γειτονιές απεικονίζονται με την βοήθεια ενός γραφήματος $G(V, E)$ όπου κάθε σημείο συνδέεται με τους κοντινότερους του γειτονες με ακμές που έχουν βάρος $d_X(i, j)$ μεταξύ γειτόνων.

Ορίζουμε το γράφημα $G = (V, E)$ και οι αποστάσεις του θα είναι οι εξής:

- $d^{(G)}(i, j) = d^{(X)}(i, j)$, αν τα σημεία i, j είναι συνδεδεμένα με μία ακμή
- αλλιώς $d^{(G)}(i, j) = \infty$

Ο αλγόριθμος Isomap εκτιμά την γεωδαισιακή απόσταση $(d^{(M)}(i, j))$ ως το μήκος του συντομότερου μονοπατιού $d^{(G)}(i, j)$ στο γράφημα G. Για την εύρεση του μήκους του συντομότερου μονοπατιού θα χρησιμοποιήσουμε τον αλγόριθμο Dijkstra ή τον αλγόριθμο Floyd–Warshall.

Αλγόριθμος Dijkstra:
Input:

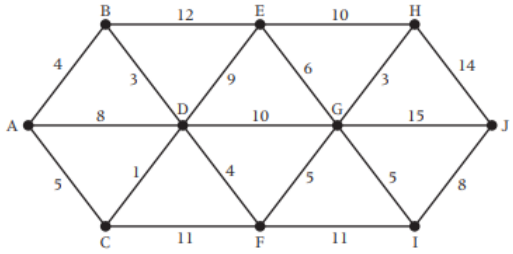
- Ένα γράφημα(Weighted Graph) $G(V, E)$

1. Βάζουμε τον αρχικό κόμβο ως 0
2. Κρατάμε σταθερή αυτήν την τιμή
3. Βρίσκουμε την απόσταση κάθε κόμβου που συνδέεται με τον αρχικό κόμβο
4. Κρατάμε τον κόμβο με την μικρότερη απόσταση
5. Από αυτόν το κόμβο, βρίσκουμε τις αποστάσεις των κόμβων που συνδέονται με αυτόν τον κόμβο
6. Αν η απόσταση είναι μικρότερη από την απόσταση που υπάρχει στην κόμβο, τότε σβήνουμε την απόσταση αυτήν και γράφουμε την νέα απόσταση. Αν δεν υπάρχει απόσταση σε αυτόν το κόμβο τότε γράφουμε την απόσταση που βρήκαμε.
7. Επαναλαμβάνουμε τον αλγόριθμο από το βήμα 4 μέχρι μέχρι να βρούμε το μήκος του συντομότερου μονοπατιού μεταξύ των δύο σημείων που θέλουμε

Output:

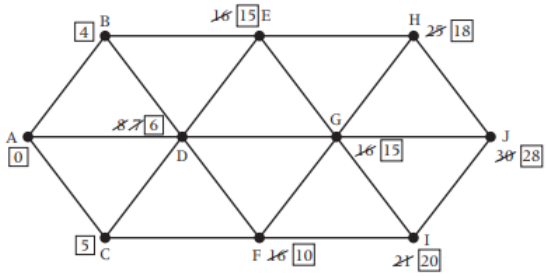
- Μήκοι των συντομότερων μονοπατιών μεταξύ σημείων.

Παράδειγμα 16. Εδώ θα βρούμε την μικρότερη απόσταση από το A στο J στο παρακάτω γράφημα.



Σχήμα 13: `https://prodtest.pearsonschoolsandfecolleges.co.uk/secondary/Mathematics/16plus/AdvancingMathsForAQA2ndEdition/Samples/SampleMaterial/Chp-02%20023-043.pdf`

Κάνοντας τον αλγόριθμος Dijkstra η λύση είναι:



Σχήμα 14: `https://prodtest.pearsonschoolsandfecolleges.co.uk/secondary/Mathematics/16plus/AdvancingMathsForAQA2ndEdition/Samples/SampleMaterial/Chp-02%20023-043.pdf`

όπου έχουμε την απόσταση 28 ($ACDFGIJ$) μεταξύ του A και του J.

Αλγόριθμος Floyd–Warshall:
Input:

- Ένα γράφημα(Weighted Graph) G

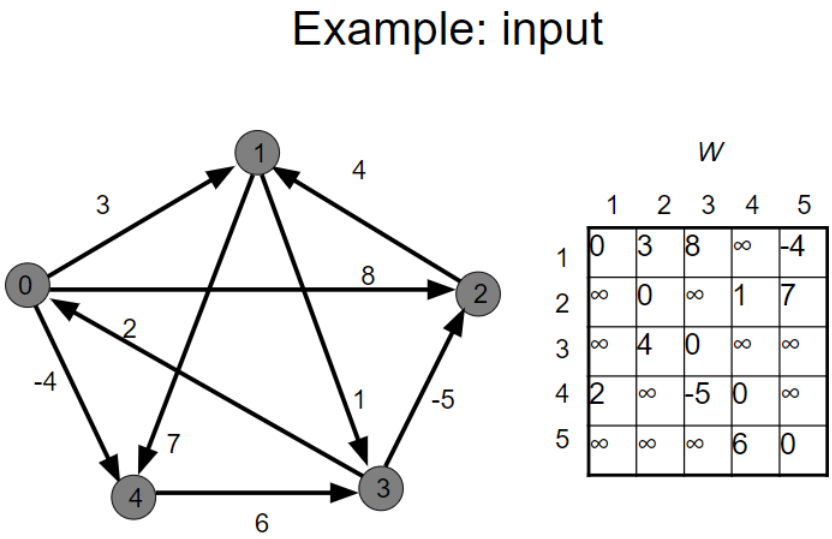
Έστω ότι έχουμε έναν πίνακα διάστασης $|V| \times |V|$ όπου $|V|$ ο αριθμός των κόμβων

```
for κάθε ακμή (u, v) do
    dist[u][v] ← w(u, v) // Το βάρος της ακμής (u, v)
end for
for Κάθε κόμβο v do
    dist[v][v] ← 0
end for
for k από 1 μέχρι |V| do
    for i από 1 μέχρι |V| do
        for j από 1 μέχρι |V| do
            if dist[i][j] > dist[i][k] + dist[k][j] then
                dist[i][j] ← dist[i][k] + dist[k][j]
            end if
        end for
    end for
end for
```

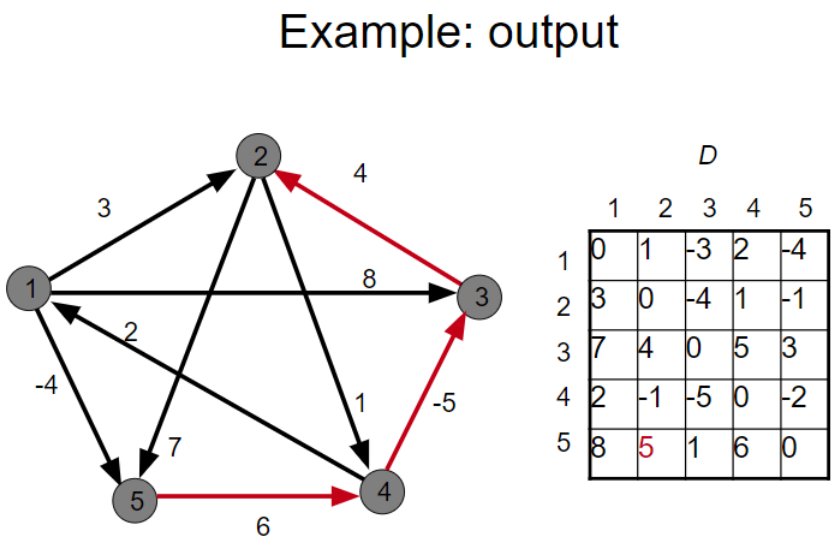
Output:

- Μήκοι των συντομότερων μονοπατιών μεταξύ σημείων.

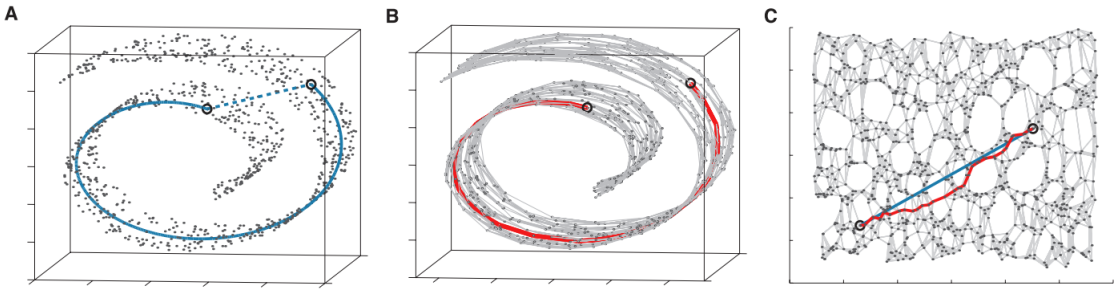
Παράδειγμα 17.



Σχήμα 15: https://github.com/epomp447/Floyd-Warshall-Algorithm-Java-



Σχήμα 16: https://github.com/epomp447/Floyd-Warshall-Algorithm-Java-



Σχήμα 17

Από τα παραπάνω τρία σχήματα Βλέπουμε την εκτίμηση της γεωδαισιακής απόστασης(κόκκινη γραμμή) μεταξύ δύο σημείων μέσω του μικρότερου μονοπατιού που βρίσκεται με την βοήθεια του αλγόριθμου Dijkstra ή του αλγόριθμου Floyd–Warshall(Σχήμα 17.c).

Αλγόριθμος ISOMAP:
Input:

- Πίνακας δεδομένων $X_{d\times n}$ όπου στις γραμμές βρίσκονται οι μεταβλητές
- Τον αριθμό των διαστάσεων που θα κρατήσουμε (έστω p)
- Την παράμετρο k(k-nearest neighbors) ή την ακτίνα ϵ

1. Κατασκευάζουμε το γράφημα γειτνίασης σημείων $G(V, E)$ χρησιμοποιώντας την παράμετρο k ή την παράμετρο ϵ.
2. Υπολογίζουμε την μικρότερη διαδρομή μεταξύ όλων των σημείων σαν μία εκτίμηση της γεωδαισιακής απόστασης (D^G). Αυτό μπορεί να γίνει με διάφορους αλγόριθμους (Dijkstra's algorithm, Floyd–Warshall algorithm)
3. Εφαρμόζουμε Multidimensional scaling (MDS) στον πίνακα $D^{(G)}$

Output:

- Ο πίνακας Y διαστάσεων $p \times n$.

Η επιλογή της παραμέτρου k η οποία εκφράζει όπως είπαμε το πλήθος των γειτόνων που θα έχει το κάθε σημείο ορίζεται από τον χρήστη. Η μη σωστή επιλογή της παραμέτρου k μπορεί να οδηγήσει σε λάθος αποτελέσματα.

Μία μέθοδος εκτίμησης της μπορεί να πραγματοποιηθεί από την εύρεση ενός μέτρου που θα υποδεικνύει την ''ποιότητα'' της απεικόνισης από τον αρχικό χώρο στον χώρο μειωμένων διαστάσεων. Δηλαδή πόσο καλά αντιπροσωπεύεται η μεγάλης διάστασης γεωμετρική δομή των δεδομένων στο αρχικό χώρο στον εμβαπτισμένο(embedded) χώρο.
Ορίζουμε λοιπόν το residual variance ως εξής:

$$1 - \rho^2_{D^{(G)}(k)D^Y}$$

όπου $\rho_{D^{(G)}(k)D^Y}$ είναι ο συντελεστής γραμμικής συσχέτισης μεταξύ όλων των στοιχείων των πινάκων $D^{(G)}(k)$ και D^Y. Οι πίνακες $D^{(G)}(k)$ και D^Y εκφράζουν τις γεωδαισιακές αποστάσεις που ορίζονται από τα συντομότερα μονοπάτια του αρχικού χώρου X τα οποία είναι συναρτήσει της παραμέτρου k και τις ευκλείδιες αποστάσεις στον χαμηλότερων διαστάσεων χώρο Y αντίστοιχα.

Όσο χαμηλότερο είναι το residual variance τόσο καλύτερη είναι η αναπαράσταση του αρχικού χώρου στον χώρο μειωμένων διαστάσεων και επομένως τόσο κατάλληλη η επιλογή του k. Η επιλογή του επομένως μπορεί να οριστεί ως :

$$k_{optimal} = \arg\min_k (1 - \rho^2_{D^{(G)}(k)D^Y}).$$

7.2 Locally linear embedding

Το Locally linear embedding (LLE) είναι μια ακόμη μέθοδος που αντιμετωπίζει το πρόβλημα της μη-γραμμικής μείωσης διαστάσεων. Η σκέψη πίσω από το LLE είναι ότι η γεωμετρία ενός manifold τοπικά μπορεί να αποδωθεί από γραμμικές συναρτήσεις. Στο LLE χρειαζεται να βελτιστοποιήσουμε 2 συναρτήσεις. Η μία συνάρτηση είναι

$$\epsilon(W) = \sum_{i=1}^{t} \|x_i - \sum_{j=1}^{K} W_{ij} x_j\|^2$$

και η άλλη είναι

$$\Phi(Y) = \sum_{i=1}^{t} \|y_i - \sum_{j=1}^{t} W_{ij} y_j\|^2.$$

Σε μεγάλης διάστασης χώρο παρόμοια με το Isomap θα φτιάξουμε τους k-κοντινότερους γείτονες για κάθε σημείο (x_i). Η 1η συνάρτηση είναι μια συνάρτηση που προσπαθεί να φτιάξει κάθε σημείο x_i από τους k-κοντινότερους γείτονες. Άρα θέλουμε

$$x_i \approx W_{i1}x_{i1} + W_{i2}x_{i2} + ... + W_{ik}x_{ik}.$$

Επομένως ο άγνωστος στην 1η συνάρτηση είναι τα βάρη.
Ας υποθέσουμε ότι λύσαμε αυτήν την συνάρτηση και ξέρουμε τα βάρη. Παρατηρούμε ότι η 2η συνάρτηση είναι παρόμοια με την 1η συνάρτηση. Όμως στην 1η συνάρτηση το x_i το γνωρίζουμε και δεν γνωρίζουμε τα βάρη W_{ij}, ενώ στην 2η συνάρτηση ξέρουμε τα βάρη W_{ij} και δεν γνωρίζουμε τα y_i. Επομένως αυτό που κάνουμε είναι ότι παίρνουμε τα βάρη που βρήκαμε στην 1η συνάρτηση (τα κρατάμε σταθερά) και τα βάζουμε στην 2η συνάρτηση και θέλουμε να βρούμε τα y_i όπου ικανοποιούν

$$y_i \approx W_{i1}y_{i1} + W_{i2}y_{i2} + ... + W_{it}y_{it}.$$

Αλγόριθμος Locally linear embedding (LLE):
Input:

- Πίνακας δεδομένων $X_{d\times n}$ όπου στις γραμμές βρίσκονται οι μεταβλητές
- Τον αριθμό των διαστάσεων που θα κρατήσουμε (έστω p)
- Την παράμετρο k(k-nearest neighbors)

1. Βρίσκουμε τους $k-$κοντινότερους γείτονες για κάθε σημείο x_i μέσω ευκλείδειων αποστάσεων.
2. Υπολογίζουμε τα w_{ij} έτσι ώστε

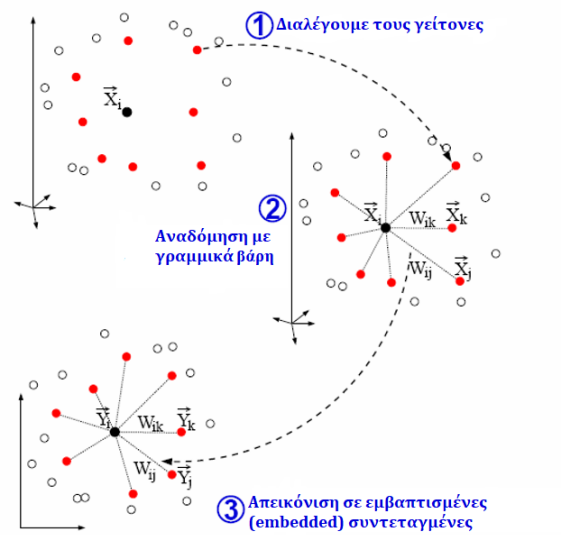
$$\min_{w_{ij}} \sum_{i=1}^{n} |x_i - \sum w_{ij} x_{u_i(j)}|^2$$

   όπου $u_i(j)$ είναι ο δείκτης του j-οστού γείτονα του x_i έτσι $\sum_{j=1}^{K} w_{ij} = 1$.
3. Υπολογίζουμε τα σημεία Y_i στον ελαττωμένο χώρο $\mathbb{R}^p$ όπου μπορούν να βρεθούν με την βοήθεια των W_{ij}, ελαχιστόποιούμε

$$\min_{Y} \sum_{i=1}^{n} |y_i - \sum_{j=1}^{n} w_{ij} y_j|^2 \ \text{ με } \ \frac{1}{n} YY^\top = I \ \text{ και } \ \sum_{i=1}^{n} y_i = 0.$$

Output:

- Ο πίνακας Y διαστάσεων $p \times n$.



Σχήμα 18

Συγκεκριμένα για τον αλγόριθμο Locally linear embedding (LLE) έχουμε:

1. Βρίσκουμε τους $k-$κοντινότερους γείτονες για κάθε σημείο x_i.
2. Υπολογίζουμε τα w_{ij} έτσι ώστε

$$\min_{w_{ij}} \sum_{i=1}^{n} |x_i - \sum w_{ij} x_{u_i(j)}|^2$$

όπου $u_i(j)$ είναι ο δείκτης του j-οστού γείτονα του x_i έτσι $\sum_{j=1}^{K} w_{ij} = 1$. Πως θα βρω το παραπάνω min ·

Έστω ότι έχουμε

$$|x_i - \sum w_{ij} x_{u_i(j)}|^2$$

Ας ορίσουμε το U_i σαν πίνακα όλων των γειτώνων του σημείου i

$$U_i = [x_{u_i(1)} \; x_{u_i(2)} \; ... \; x_{u_i(k)}]_{d \times K}.$$

Επίσης ορίζουμε το διάνυσμα w_i

$$w_i = \begin{bmatrix} w_{i1} \\ w_{i2} \\ \vdots \\ w_{iK} \end{bmatrix}_{K \times 1}$$

και e είναι ένα $K \times 1$ πινακας με μονάδες

$$e = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{K \times 1}$$

Με την βοήθεια των παραπάνω μπορώ να γράψω το άθροισμα $\sum w_{ij} x_{u_i(j)}$ σε μορφή πίνακα

$$\sum w_{ij} x_{u_i(j)} = U_i w_i$$

Άρα η συνάρτηση θα είναι

$$|x_i - U_i w_i|^2$$

οπου το w_i είναι άγνωστο. Θα φτιάξουμε έναν πίνακα αυτής της μορφής

$$[x_i \; x_i \; ... \; x_i]_{d\times K},$$

αν θέλω να γράψω τον παραπάνω πίνακα συναρτήση του e τότε έχω

$$[x_i \; x_i \; ... \; x_i]_{d\times K} = x_i - e^\top$$

Τώρα μπορούμε να ισχυριστούμε ότι

$$\min x_i = x_i e^\top w_i.$$

Ξαναγράφουμε την συνάρτηση

$$\min |x_i e^\top w_i - U_i w_i|^2 = |(x_i e^\top - U_i) w_i|^2.$$

Αν αναλύσουμε την νόρμα

$$\min w_i^\top (x_i e^\top - U_i)^\top (x_i e^\top - U_i) w_i.$$

Το κομμάτι $(x_i e^\top - U_i)^\top (x_i e^\top - U_i)$ είναι γνωστό διότι εξαρτάται από τα x_i, e και το U_i όπου όλα είναι γνωστά. Ας ονομάσουμε τον γνωστό όρο ως G

$$G = (x_i e^\top - U_i)^\top (x_i e^\top - U_i)$$

Το G είναι γνωστό και ονομάζεται Gram matrix. Έχουμε

$$\min w_i^\top G w_i.$$

Άμα δεν έχουμε κανέναν περιορισμό τότε η λύση της παραπάνω συνάρτησης είναι 0. Όμως εμείς έχουμε τον περιορισμό ότι $\sum_{j=1}^K w_{ij} = 1$. Ο περιορισμός θα γίνει

$$\sum_{j=1}^K w_{ij} = e^\top w_i = 1$$

Γράφουμε την Lagrangian

$$L(w_i, \lambda) = w_i^\top G w_i - \lambda(e^\top w_i - 1),$$

παίρνουμε την παράγωγο της Lagrangian συναρτήση του w_i ίση με το 0

$$\frac{\partial L}{\partial w_i} = 2Gw_i - \lambda e = 0$$

$$2Gw_i = \lambda e \Rightarrow Gw_i = \frac{\lambda}{2} e$$

όπου e είναι γνωστό, G είναι γνωστό, ενώ w_i, λ είναι άγνωστα. Αν γνωρίζουμε το λ μπορούμε να υπολογίσουμε το w_i πολλαπλασιάζοντας και τις δύο μεριές με G^{-1}. Αφού ξέρουμε ότι $\sum_{j=1}^K w_{ij} = 1$ τότε μπορούμε να θέσουμε ένα arbitrary(αυθαίρετο) λ λύνοντας ως προς w_i. Άρα μπορούμε να λύσουμε για καθε λ και μπορούμε να κανονικόποιήσουμε το w_i έτσι ώστε το άθροισμα τους να μας κάνει 1.

3. Ελαχιστοποιούμε

$$\min_Y \sum_{i=1}^n |y_i - \sum_{j=1}^n w_{ij} y_j|^2 \;\; \text{με} \;\; \frac{1}{n} Y Y^\top = I \;\; \text{και} \;\; \sum_{i=1}^n y_i = 0.$$

Έχουμε ότι $Y = [y_1 \; ... \; y_n]_{p\times n}$. Ας ορίσουμε

$$I = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}_{n\times n}$$

και ορίζουμε

$$I_{:i} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

όπου είναι παντού μηδέν εκτός στο σημείο i που είναι ίσο με 1. Επιπλέον ορίζουμε τον πίνακα W που είναι ένας $n \times n$ πίνακας με τα βάρη. Άρα έχουμε

$$W_{:i} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ w_1 \\ \vdots \\ w_k \\ \vdots \end{bmatrix}$$

όπου τα w_1 έως τα w_k είναι βάρη των $k-$κοντινότερων γειτώνων του x_i και μηδέν αλλού. Άρα έχοντας αυτούς τους συμβολισμούς (notations) μπορούμε να γράψουμε το $\min_Y \sum_{i=1}^n |y_i - \sum_{j=1}^n w_{ij} y_j|^2$ με διαφορετική μορφή.

Το y_i με την βοήθεια αυτών που γράψαμε είναι

$$y_i = YI_{:i},$$

το $\sum_{j=1}^n w_{ij} y_j$ θα γραφτεί ως

$$\sum_{j=1}^n w_{ij} y_j = YW_{:i}.$$

Επομένως θα έχουμε

$$\min_Y \sum_{i=1}^n |YI_{:i} - YW_{:i}|^2 = \min_Y |YI - YW|^2$$

$$\min_Y |Y(I-W)|^2 = \min_Y Tr[(I-W)^\top Y^\top Y(I-W)].$$

Αφού έχουμε trace τότε

$$\min_Y Tr[(I-W)^\top Y^\top Y(I-W)] = \min_Y Tr[Y(I-W)(I-W)^\top Y^\top].$$

Ας ονομάσουμε $(I-W)(I-W)^\top = M$. Η συγκεκριμένη βελτιστοποίηση είναι παρόμοια με τις βελτιστοποιήσεις που έχουμε κάνει έως τώρα. Χωρίς κανεναν περιορισμό η λύση θα είναι το 0. Ας χρησιμοποιήσουμε τον 1ο περιορισμό ($\frac{1}{n} YY^\top = I$). Με αυτόν τον περιορισμό η λύση θα είναι τα p ελάχιστα ιδιοδιανύσματα του M.

Χρησιμοποιούμε πολλαπλασιαστής Lagrange(Lagrange multiplier)

$$L(Y) = Tr(YMY^\top) + Tr(\Lambda \left(\frac{1}{n} YY^{\top - I}) \right)$$

Παίρνουμε την παράγωγο ίση με το 0

$$\frac{\partial L}{\partial Y} = MY^\top + \frac{1}{n} \Lambda Y^\top = 0$$

Άρα έχουμε ότι

$$MY^\top = -\frac{1}{n}\Lambda Y^\top$$

όπου σημαίνει ότι οι στήλες του $Y^\top$ είναι ιδιοτιμές του M. Το M θα έχει πάντα μια ιδιοτιμή ίση με το 0. Άρα το ιδιοδιάνυσμα που αντιστοιχεί σε αυτήν την ιδιοτιμή είναι $\begin{bmatrix}1\\ \vdots \\ 1\end{bmatrix}$. Διαλέγουμε το $p+1$ πρώτο ελάχιστο ιδιοδιάνυσμα και αγνοούμε αυτό που αντιστοιχεί στην ιδιοτιμή με τιμή 0.

7.3 Spectral Clustering

Έστω ότι έχουμε $\{x_i\}_{i=1}^n$. Φτιάχνουμε το weighted γράφημα (graph) $G = (V, E)$ όπου E είναι το σύνολο των ακμών(edges) και $V = (u_1, ..., u_n)$ το σύνολο κόμβων(vertices). Το weighted γράφημα(graph) είναι ένα γράφημα με $w : V \times V \to \mathbb{R}$. Επίσης τα βάρη χρειάζεται να ακολουθούν τις παρακάτω συνθήκες

1. $w(u_i, u_j) = w(u_j, u_i)$ για κάθε $u_i, u_j \in V$
2. $w(u_i, u_j) \geq 0$ για κάθε $u_i, u_j \in V$

Έχουμε d_{u_i} ως

$$d_{u_i} = \sum_{u_j \in V} w(u_i, u_j)$$

Επιπλέον ορίζουμε τον πίνακα $T_{n\times n}$ όπου είναι ένας διαγώνιος πίνακας με $T_{ii} = d_{u_i}$ και τον πίνακα $L_{n\times n}$ έτσι ώστε

$$L_{ij} = \begin{cases} d_{u_i} - w(u_i, u_j) & \text{αν } \ u_i = u_j \\ -w(u_i, u_j) & \text{αν } u_i \text{ και } u_j \text{ είναι γειτονικά} \\ 0 & \text{αλλού} \end{cases}$$

και η λαπλασιανή του γραφήματος(graph) G ορίζεται ως

$$L = T^{-1/2} L T^{-1/2}.$$

Η πιθανότητα μετάβασης $P(u_i, u_j)$ είναι

$$P(u_i, u_j) = \frac{w(u_i, u_j)}{d_{u_i}} \quad \text{για κάθε} \quad u_i, u_j \in V$$

όπου είναι ο τυχαίος περίπατος στο weighted γράφημα(graph). Ο τυχαίος περίπατος έχει πιθανότητα $P(u_i, u_j)$ να κινηθεί από το u_i στο u_j.

Έχοντας ένα γράφημα(graph) θέλουμε να το «κόψουμε» σε δύο κομμάτια. Οι κόμβοι του ενός υπογραφήματος πρέπει να είναι παρόμοιες μεταξύ τους (το ίδιο ισχύει και για το 2ο υπογράφημα).
Έστω ότι το γράφημα είναι συνεκτικό(connected). Τότε για να το κόψω σε δύο κομμάτια χρειάζεται να κόψω κάποιες ακμές (edges). Άρα επιλέγω ποιο υποσύνολο από ακμές πρέπει να κοπεί για να κάνουμε 2 γραφήματα που ικανοποιούν τις παραπάνω προϋποθέσεις.
Ένας τρόπος είναι να διαλέξω ένα σύνολο από ακμές έτσι ώστε τα άθροισμα με τα βάρη σε αυτές τις ακμές να είναι μικρό(διότι τα βάρη δίχνουν την ομοιότητα μεταξύ σημείων). Δηλαδή θέλω να κόψω ακμές που έχουν μικρά βάρη.

Ορίζουμε

$$cut(A, \bar{A}) = \sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j)$$

όπου $\bar{A} = V - A = V \backslash A$ και $V = A \cup \bar{A}$.

Άρα έχουμε

$$\min cut(A, \bar{A})$$

ή

$$\min \sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j)$$

Όμως το cut δεν δουλεύει καλά άμα έχουμε ακραίες τιμές στα δεδομένα μας. Επομένως θα ορίσουμε το ratiocut

$$ratiocut(A, \bar{A}) = \frac{cut(A, \bar{A})}{|A|} + \frac{cut(\bar{A}, A)}{|\bar{A}|}$$

Ορίζουμε το διάνυσμα $f \in \mathbb{R}^n$ και

$$f_i = \begin{cases} \sqrt{\frac{|\bar{A}|}{|A|}} & \text{αν} \quad u_i \in A \\ -\sqrt{\frac{|A|}{|\bar{A}|}} & \text{αν} \quad u_i \in \bar{A} \end{cases}$$

Πρόταση 3. Η ελαχιστοποίηση του $\sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j)(f_i - f_j)^2$ είναι το ίδιο με την ελαχιστοποίηση του ratiocut.

Απόδειξη :

$$\sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j)(f_i - f_j)^2 = \sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j) \left(\sqrt{\frac{|\bar{A}|}{|A|}} + \sqrt{\frac{|A|}{|\bar{A}|}} \right)^2 + \sum_{u_i \in \bar{A}, u_j \in A} w(u_i, u_j) \left(-\sqrt{\frac{|A|}{|\bar{A}|}} - \sqrt{\frac{|\bar{A}|}{|A|}} \right)^2 =$$

$$= \sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j) \left(\frac{|\bar{A}|}{|A|} + \frac{|A|}{|\bar{A}|} + 2 \right) + \sum_{u_i \in \bar{A}, u_j \in A} w(u_i, u_j) \left(\frac{|A|}{|\bar{A}|} + \frac{|\bar{A}|}{|A|} + 2 \right),$$

άρα έχουμε

$$\left(\frac{|\bar{A}|}{|A|} + \frac{|A|}{|\bar{A}|} + 2 \right) \left[\sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j) + \sum_{u_i \in \bar{A}, u_j \in A} w(u_i, u_j) \right].$$

Ξέρουμε ότι $cut(A, \bar{A}) = \sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j)$, $cut(\bar{A}, A) = \sum_{u_i \in \bar{A}, u_j \in A} w(u_i, u_j)$ και $cut(A, \bar{A}) = cut(\bar{A}, A)$. Άρα έχουμε ότι

$$\left(\frac{|\bar{A}|}{|A|} + \frac{|A|}{|\bar{A}|} + 2 \right) \left[2cut(A, \bar{A}) \right]$$

τον πρώτο όρο των γράφουμε ως

$$\left(\frac{|\bar{A}|}{|A|} + \frac{|A|}{|\bar{A}|} + 2 \right) = \left(\frac{|\bar{A}|}{|A|} + \frac{|A|}{|\bar{A}|} + \frac{|A|}{|A|} + \frac{|\bar{A}|}{|\bar{A}|} \right)$$

Επομένως έχουμε

$$2 \left(\frac{|\bar{A}| + |A|}{|A|} + \frac{|\bar{A}| + |A|}{|\bar{A}|} \right) cut(A, \bar{A}).$$

Επίσης έχουμε ότι $|\bar{A}| + |A| = n$

$$2n \left(\frac{1}{|A|} + \frac{1}{|\bar{A}|} \right) cut(A, \bar{A})$$

$$2n \left(\frac{cut(A, \bar{A})}{|A|} + \frac{cut(A, \bar{A})}{|\bar{A}|} \right) = 2n \cdot ratiocut$$

■

Άρα πρέπει να κάνουμε την παρακάτω βελτιστοποίηση

$$\min_{f_i, f_j} \sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j)(f_i - f_j)^2.$$

Πρόταση 4. $\frac{1}{2} \sum_{u_i \in A, u_j \in \bar{A}} w(u_i, u_j)(f_i - f_j)^2 = f^\top L f$ όπου L είναι η λαπλασιανή του γραφήματος G.

Η λαπλασιανή ενός γραφήματος (graph) ξέρουμε ότι είναι

$$L = D - W$$

όπου W είναι $n \times n$ πίνακας με τα βάρη

$$W = \begin{bmatrix} w_{11} & \dots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{n1} & \dots & w_{nn} \end{bmatrix}_{n \times n}$$

και

$$D = \begin{bmatrix} d_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & d_n \end{bmatrix}_{n \times n}$$

διαγώνιος πίνακας όπου $d_i = \sum_j w_{ij}$.

Έχουμε ότι

$$f^\top L f = f^\top (D - W) f = f^\top D f - f^\top W f = \frac{1}{2} \left(2 \sum_i d_i f_i^2 - 2 \sum_{ij} w_{ij} f_i f_j \right) =$$

$$= \frac{1}{2} \left(\sum_i d_i f_i^2 + \sum_j d_j f_j^2 - 2 \sum_{ij} w_{ij} f_i f_j \right) = \frac{1}{2} \left(\sum_i f_i^2 \sum_j w_{ij} + \sum_j f_j^2 \sum_i w_{ij} - 2 \sum_{ij} w_{ij} f_i f_j \right) =$$

$$= \frac{1}{2} \left(\sum_{ij} w_{ij} f_i^2 + \sum_{ij} w_{ij} f_j^2 - 2 \sum_{ij} w_{ij} f_i f_j \right) = \frac{1}{2} \sum_{ij} w_{ij} (f_i - f_j)^2.$$

Άρα έχουμε

$$\min_f f^\top L f.$$

Χωρίς περιορισμούς η λύση του παραπάνω είναι το μηδέν. Μπορούμε να πούμε ότι

$$\min_f f^\top L f \quad \text{έτσι ώστε} \quad f^\top f = 1.$$

Εδώ η λύση είναι το ελάχιστο ιδιοδιάνυσμα του πίνακα L. Το πρώτο ιδιοδιάνυσμα έχει ιδιοτιμή ίση με το μηδέν. Επομένως αγνοούμε το πρώτο ιδιοδιάνυσμα και θα ξεκινάμε από το δεύτερο μικρότερο. Άρα το f θα είναι το δεύτερο μικρότερο ιδιοδιάνυσμα του L.

Άρα αυτό κάνουμε άμα θέλουμε να χωρίσουμε τα δεδομένα σε δύο clusters. Άμα θέλουμε να φτιάξουμε παραπάνω από δύο clusters τότε θα πάρουμε παραπάνω ιδιοδιανύσματα του πίνακα L.

- Για παράδειγμα αν θέλουμε 3 clusters θα πάρουμε 2 ιδιοδιανύσματα
- k-clusters θα πάρουμε (k-1) ιδιοδιανύσματα κ.λ.π.

Υπολογίζουμε την ομοιότητα κατά ζεύγη $s_{ij} = s(x_i, x_j)$ με την βοήθεια μιας συνάρτησης ομοιότητας όπου είναι συμμετρική και μη αρνητική και έχουμε τον πίνακα $S = (s_{ij})_{i,j=1,\dots,n}$.

Αλγόριθμος Spectral Clustering:
Input:

- Ένας πίνακα $S \in \mathbb{R}^{n \times n}$, και k clusters.

1. Φτιάχνουμε το Weighted Graph και τον πίνακα $W_{n \times n}$
2. Υπολογίζουμε την λαπλασιανή L
3. Υπολογίζουμε το πρώτα k ιδιοδιανύσματα $u_1, ..., u_k$ της λαπλασιανής(L)
4. Έστω $U \in \mathbb{R}^{n \times k}$ είναι ο πίνακας που περιέχει τα ιδιοδιανύσματα $u_1, ..., u_k$ ως στήλες
5. Για $i = 1, ..., n$ έχουμε $y_i \in \mathbb{R}^k$ όπου y_i είναι η i γραμμή του πίνακα U
6. Κάνουμε cluster τα σημεία y_i με τον αλγόριθμο k-means σε Clusters $C_1, ..., C_k$.

Output:

- Καταλήγουμε στα clusters $A_1, ..., A_k$ με $A_i = \{j | y_j \in C_i\}$.

7.4 Laplacian Eigenmap

Το Laplacian Eigenmap είναι παρόμοιο με το isomap και το LLE διότι είναι και αυτό με την σειρά του ένας τρόπος μη-γραμμικού dimensionality reduction.

Αλγόριθμος Laplacian Eigenmap:
Input:

- Πίνακας δεδομένων $X_{d\times n}$
- Τον αριθμό των διαστάσεων που θα κρατήσουμε (έστω p)
- Την παράμετρο k(k-nearest neighbors) ή την ακτίνα ϵ

1. Φτιάχνουμε τό weighted γράφημα (graph) $G = (V, E)$. Διαλέγουμε την απόσταση που θα χρησιμοποιήσουμε:
   - Έστω $\epsilon > 0$ ένωνουμε ακμές (edges) μεταξύ i και j αν $\|x_i - x_j\|^2 < \epsilon$.
   - k-κοντινότερους γείτονες.
2. Διαλέγουμε τα βάρη.
   Για παράδειγμα:
   - Ορίζουμε $w_{ij} = \begin{cases} 1 & \text{αν } i \text{ και } j \text{ συνδέονται (connected)} \\ 0 & \text{αλλού} \end{cases}$
   - ή $w_{ij} = e^{-\frac{\|x_i - x_j\|^2}{t}}$ για κάποιο $t > 0$.
3. Ελαχιστοποιούμε
   $$\min \sum_Y w_{ij}|y_i - y_j|^2$$
   όπου $x \in \mathbb{R}^d$ και $y \in \mathbb{R}^p$ με $p < d$. (Παρατηρούμε ότι αυτή η βελτιστοποιήση μοιάζει με την βελτιστοποίηση που είχαμε στο spectral clustering).

   Επομένως έχουμε ότι
   $$\min Trace(Y^\top LY) \quad \text{έτσι ώστε} \quad Y^\top Y = I$$
   όπου Y είναι ο πίνακας $p \times n$ με τα εμβαπτισμένα δεδομένα $Y = [y_1, ..., y_n] \in \mathbb{R}^{n\times p}$.

Output:

- Ο πίνακας Y διαστάσεων $p \times n$.

7.5 Diffusion Maps

Όπως ορίσαμε και παραπάνω έτσι θα ορίσουμε τα δεδομένα $X = \{x_1, x_2, ..., x_n\}$ όπου είναι ένα σύνολο από κόμβους(vertices) του σταθμισμένου(weighted) γραφήματος με βάρη w. Ο στόχος και εδώ είναι να πάμε από τα σημεία $x_i \in \mathbb{R}^d$ στα $y_i \in \mathbb{R}^p$ όπου $p < d$ έτσι ώστε το $Y = \{y_1, y_2, ..., y_N\}$.

Έχουμε ήδη αναφέρει την πιθανότητα μετάβασης

$$P(x_i, x_j) = \frac{w(x_i, x_j)}{d_{x_i}}$$

όπου

$$w(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{\epsilon}}$$

και

$$d_{x_i} = \sum_{x_j \in X} w(x_i, x_j)$$

Ορίζουμε το κανονικοποιημένο $w(x_i, x_j)$ ως

$$\tilde{w}(x_i, x_j) = \frac{w(x_i, x_j)}{q^\alpha(x_i) q^\alpha(x_j)}$$

όπου $q^\alpha(x_i) = \sum_{x_k \in X} w(x_i, x_k)$.

Η ασυμπτωτική συμπεριφορά $\epsilon \to 0$ μας δίνει διαφορετικούς απειροστικούς(infinitesimal) τελεστές για διαφορετικές τιμές του α:

- $\alpha = 0$: Graph Laplacian
- $\alpha = 1$: Προσέγγιση του τελεστή Laplace-Beltrami

Έχουμε ότι

$$d^{(\alpha)}(x_j) = \sum_{x_j \in X} \tilde{w}(x_i, x_j)$$

Αλγόριθμος Diffusion Map:
Input:

- Πίνακας δεδομένων $X_{d \times n}$
- Τον αριθμό των διαστάσεων που θα κρατήσουμε (έστω p)
- Την παράμετρο ϵ
- Την παράμετρο t

1. Φτιάχνουμε τον πίνακα W όπου $W_{ij} = e^{-\frac{\|x_i - x_j\|^2}{\epsilon}}$
2. Κανονικοποιούμε το $W(x_i, x_j)$ με βοήθεια του $q^\alpha(x_i) q^\alpha(x_j)$ όπου $\tilde{W} = \sqrt{q}\ W\ \frac{1}{\sqrt{q}}$
3. Φτιάχνουμε τον πίνακα μετάβασης $P = \sqrt{d}\ \tilde{W}\ \frac{1}{\sqrt{d}}$
4. Κανουμε SVD(Singular Value Decomposition) στο P

$$P = U \Sigma V^\top$$

5. Με την βοήθεια του βήματος 4 βρίσκουμε τις ιδιοτιμές (λ_i) και τα ιδιοδιανύσματα (ϕ_j) του τελεστή Laplace-Beltrami.
6. Το diffusion map ορίζεται ως

$$x_i \to y_i = \left(\lambda_2^{t/2} \phi_2(i), \lambda_3^{t/2} \phi_3(i), ..., \lambda_p^{t/2} \phi_p(i)\right)^\top$$

   όπου $\phi_j(i)$ είναι η i component του j ιδιόδιανυσματος ϕ_j και $\lambda_i = \Sigma_{jj}$

Output:

- Ο πίνακας Y διαστάσεων $p \times n$.

Ορισμός 7.1. Ορίζουμε την οικογένεια των αποστάσεων διάχυσης(Diffusion distances) D_t με $t \geq 1$ ως

$$D_t(x,y) = \sqrt{\sum_i^n \lambda_i^{2t} \left(\phi_i(x) - \phi_i(y)\right)^2}$$

όπου η παράμετρος t ελέγχει την ευαισθησία της μετρικής της διάχυσης (diffusion) D_t.

Αφού $0 \leq \lambda_i \leq 1$, μία μεγάλη τιμή στο t θα κάνει το φάσμα να αποσυνθεθεί πολύ πιο γρήγορα και έτσι το D_t θα εξαρτάται μόνο από μερικά ιδιοδιανύσματα ϕ_i.

Το $D_t(x,y)$ είναι ένας τρόπος να βρούμε την συνεκτικότητα(connectivity) μεταξύ των σημείων x και y. Τα diffusion maps Φ_t είναι

$$x \to \Phi_t(x) = \left(\lambda_0^{t/2}\phi_0(x), \lambda_1^{t/2}\phi_1(x), \lambda_2^{t/2}\phi_2(x), \lambda_3^{t/2}\phi_3(x), ...\right)^\top$$

αφού ϕ_0 είναι σταθερά έχουμε

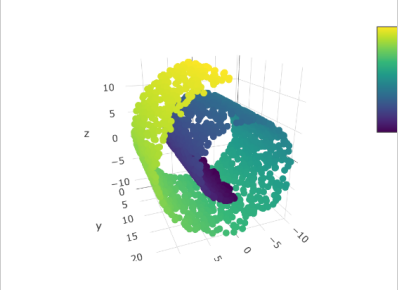
$$x \to \Phi_t(x) = \left(\lambda_1^{t/2}\phi_1(x), \lambda_2^{t/2}\phi_2(x), \lambda_3^{t/2}\phi_3(x), ...\right)^\top$$

και έτσι με την βοήθεια της απεικόνισης διάχυσης (Diffusion map) η απόσταση διάχυσης (Diffusion distance) είναι

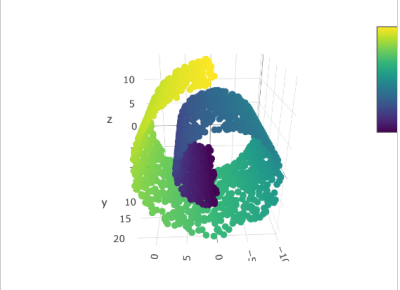
$$D_t(x,y) = \|\Phi_t(x) - \Phi_t(y)\|.$$

8 Πειράματα

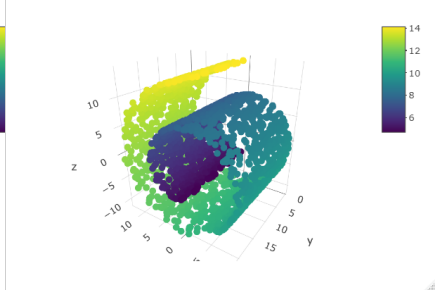
8.1 Swiss Roll



Σχήμα 19: Swissroll Data 1



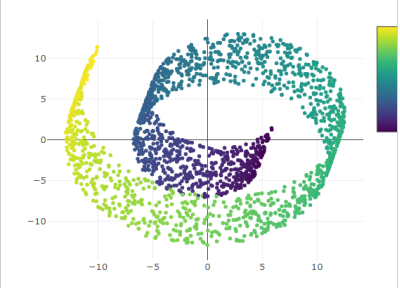
Σχήμα 20: Swissroll Data 2



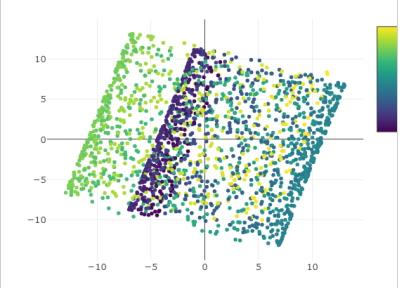
Σχήμα 21: Swissroll Data 3

Για τα παραπάνω δεδομένα θα χρησιμοποιήσουμε τους παραπάνω αλγόριθμους (PCA, ISOMAP, LLE, Laplacian Eigenmaps, Diffusion maps) για να κάνουμε dimensionality reduction από τις 3 διαστάσεις στις 2.

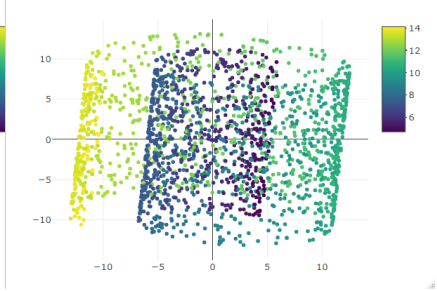
Principal Component Analysis(P.C.A)



Σχήμα 22: Swissroll plot του 1ου και του 2ου component



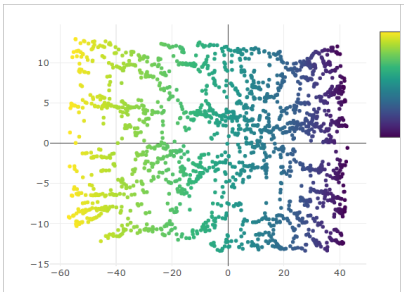
Σχήμα 23: Swissroll plot του 2ου και του 3ου component



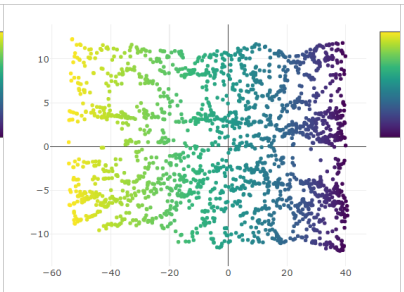
Σχήμα 24: Swissroll plot του 1ου και του 3ου component

Παρατηρούμε ότι ο αλγόριθμος PCA δεν κατάφερε να ξεδιπλώσει το Swiss roll manifold. Αυτό συμβαίνει διότι το Swiss roll βλέπουμε ότι είναι ένα μη γραμμικό manifold.

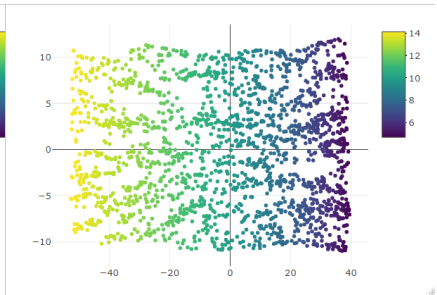
Isometric Mapping(ISOMAP)



Σχήμα 25: ISOMAP αλγόριθμος με 5-κοντινότερους γείτονες



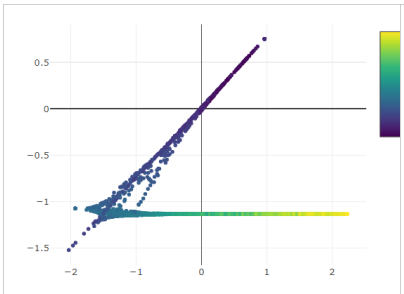
Σχήμα 26: ISOMAP αλγόριθμος με 7-κοντινότερους γείτονες



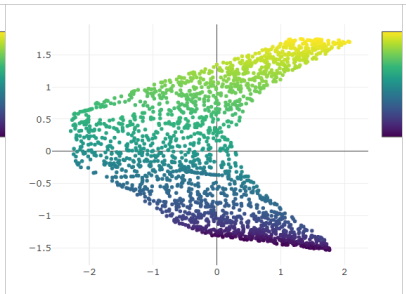
Σχήμα 27: ISOMAP αλγόριθμος με 10-κοντινότερους γείτονες

Βλέπουμε ότι το Isomap έκανε μία αρκετά καλή δουλειά και κατάφερει να ξεδιπλώση το Swiss roll. Όμως γενικά αυτός ο αλγόριθμος είναι υπολογιστικά αργός.

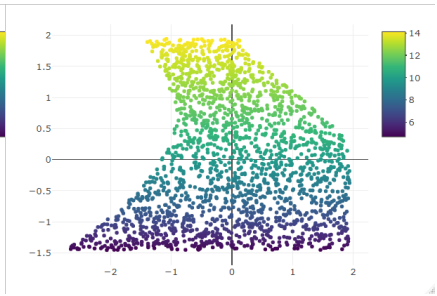
Locally Linear Embedding(LLE)



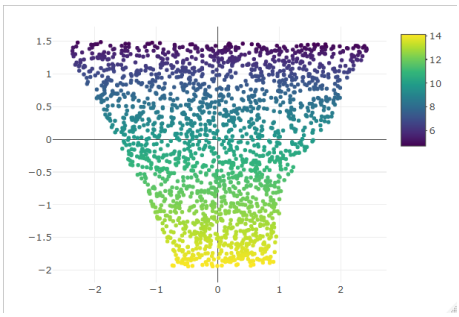
Σχήμα 28: LLE αλγόριθμος με 5-κοντινότερους γείτονες



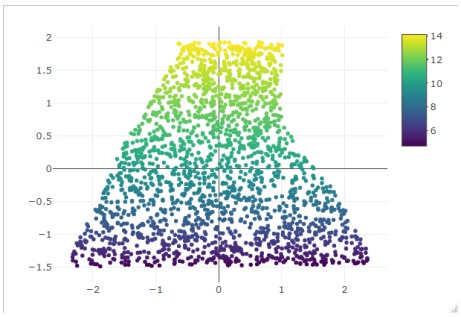
Σχήμα 29: LLE αλγόριθμος με 7-κοντινότερους γείτονες



Σχήμα 30: LLE αλγόριθμος με 10-κοντινότερους γείτονες



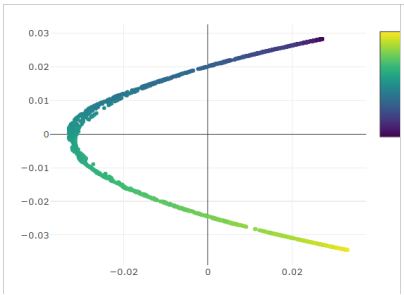
(α΄) LLE αλγόριθμος με 15-κοντινότερους γείτονες



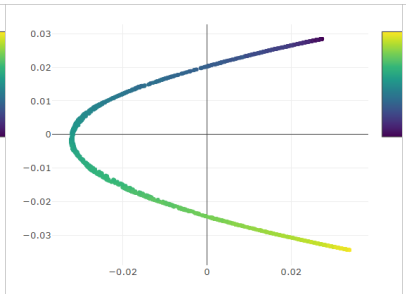
(β΄) LLE αλγόριθμος με 18-κοντινότερους γείτονες

Το LLE παρατηρούμε ότι δεν μας δίνει καλά αποτελέσματα έχοντας χαμηλό αριθμό γειτόνων. Για 15 και 18 γείτονες βλέπουμε ότι έχει δώσει ένα ικανοποιητικό αποτέλεσμα αλλά ο αλγόριθμος Isomap δίνει καλύτερο αποτέλεσμα. Σε σχέση με τον αλγόριθμο Isomap ο LLE είναι αρκετά πιο γρήγορος υπολογιστικά.

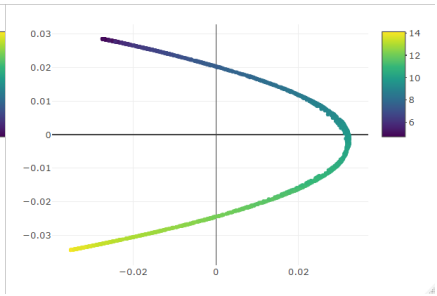
Laplacian Eigenmaps



Σχήμα 32: Laplacian Eigenmaps αλγόριθμος με 5-κοντινότερους γείτονες



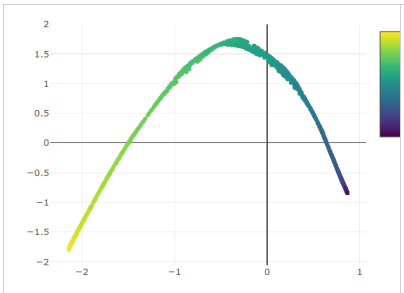
Σχήμα 33: Laplacian Eigenmaps αλγόριθμος με 7-κοντινότερους γείτονες



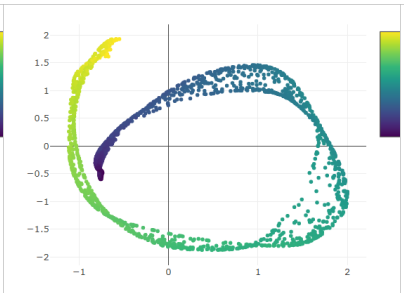
Σχήμα 34: Laplacian Eigenmaps αλγόριθμος με 10-κοντινότερους γείτονες

Το Laplacian Eigenmap δεν είναι ο κατάλληλος αλγόριθμος για να ξεδιπλώσουμε το manifold.

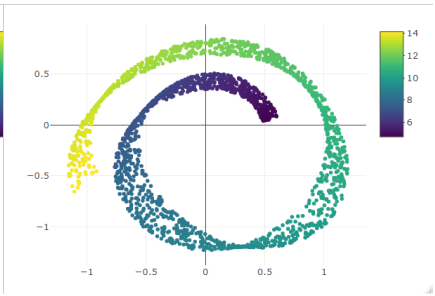
Diffusion Maps



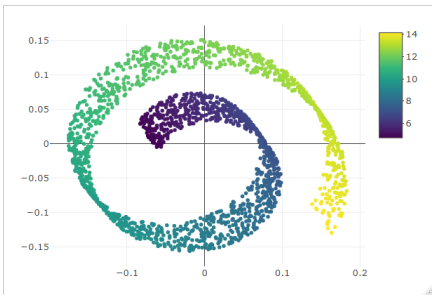
Σχήμα 35: Diffusion Maps αλγόριθμος με bandwidth = 5



Σχήμα 36: Diffusion Maps αλγόριθμος με bandwidth = 10



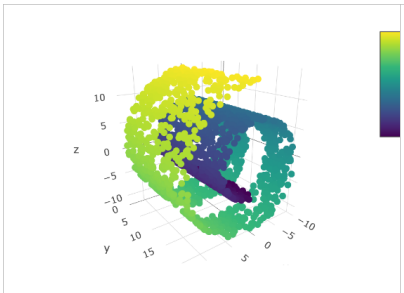
Σχήμα 37: Diffusion Maps αλγόριθμος με bandwidth = 100



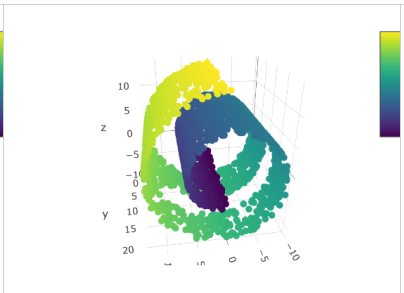
Σχήμα 38: Diffusion Maps αλγόριθμος με bandwidth = 1000

Βλέπουμε ότι ούτε ο αλγόριθμος Diffusion Maps μπορεί να ξεδιπλώσει το Swiss roll manifold.

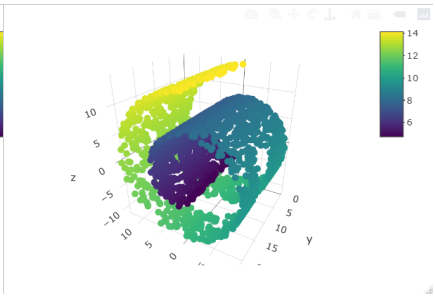
8.2 Swiss Roll με τρύπα



Σχήμα 39: Swissroll hole Data 1



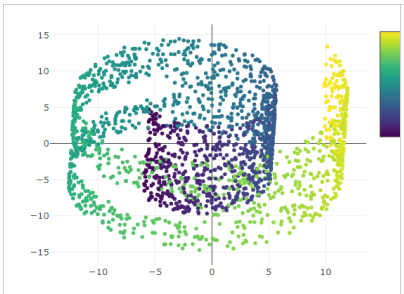
Σχήμα 40: Swissroll hole Data 2



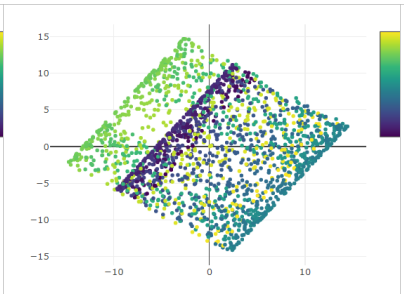
Σχήμα 41: Swissroll hole Data 3

Το παραπάνω Swiss roll παρατηρούμε ότι έχει μία τρύπα στα δεδομένα. Αυτό που θέλουμε να δούμε είναι αν αυτή η τρύπα επηρεάσει τα αποτελέσματα των αλγόριθμων. Άρα και εδώ θα χρησιμοποιήσουμε τους παραπάνω αλγόριθμους (PCA, ISOMAP, LLE, Laplacian Eigenmaps, Diffusion maps) για να κάνουμε dimensionality reduction στο παραπάνω manifold από τις 3 διαστάσεις στις 2.

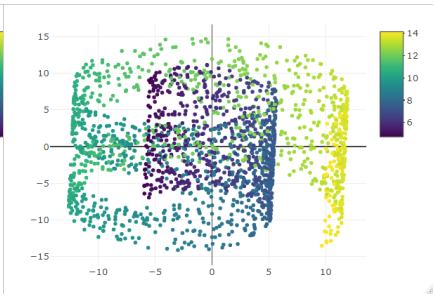
Principal Component Analysis(P.C.A)



Σχήμα 42: Swissroll hole plot του 1ου και του 2ου component



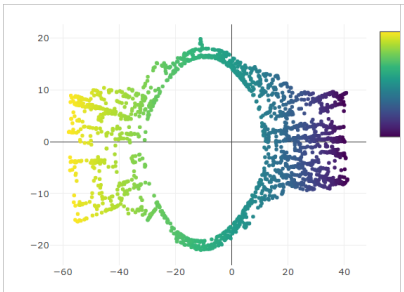
Σχήμα 43: Swissroll hole plot του 2ου και του 3ου component



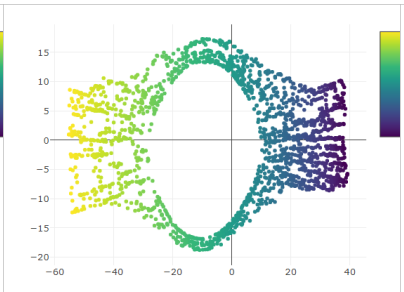
Σχήμα 44: Swissroll hole plot του 1ου και του 3ου component

Παρατηρούμε ότι ο αλγόριθμος PCA δεν μπόρεσε να ξεδιπλώσει το Swiss roll με τρύπα πράγμα το οποίο περιμέναμε.

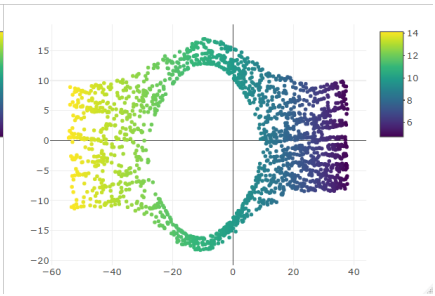
Isometric Mapping(ISOMAP)



Σχήμα 45: ISOMAP αλγόριθμος με 5-κοντινότερους γείτονες



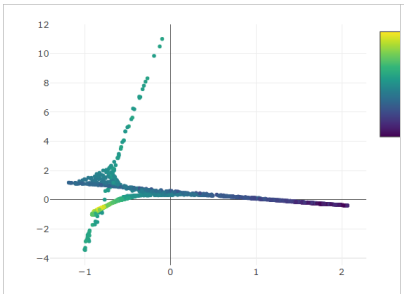
Σχήμα 46: ISOMAP αλγόριθμος με 7-κοντινότερους γείτονες



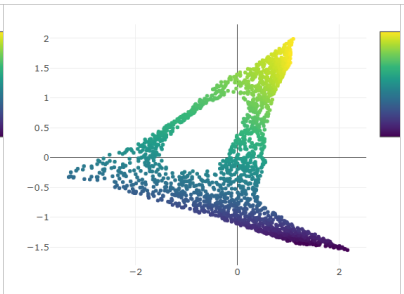
Σχήμα 47: ISOMAP αλγόριθμος με 10-κοντινότερους γείτονες

Βλέπουμε ότι για το Swiss roll με την τρύπα μας είναι πιο δύσκολο να το ξεδιπλώσουμε χρησιμοποιόντας τον αλγόριθμο Isomap. Επομένως η τρύπα επηρέασε το αποτέλεσμα του αλγόριθμο ISOMAP(παραμόρφωσε το αποτέλεσμα).

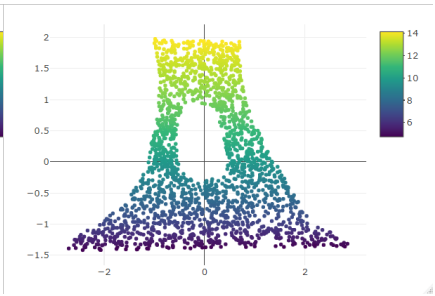
Locally Linear Embedding(LLE)



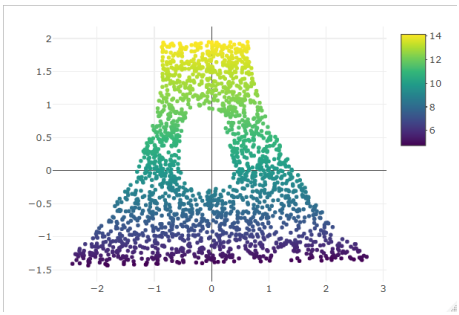
Σχήμα 48: LLE αλγόριθμος με 5-κοντινότερους γείτονες



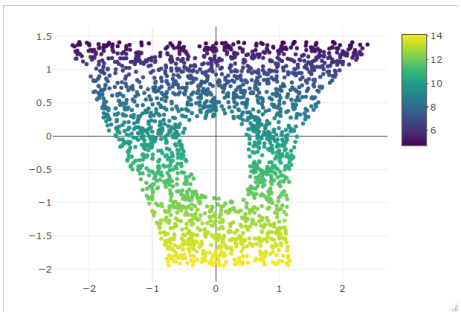
Σχήμα 49: LLE αλγόριθμος με 7-κοντινότερους γείτονες



Σχήμα 50: LLE αλγόριθμος με 10-κοντινότερους γείτονες



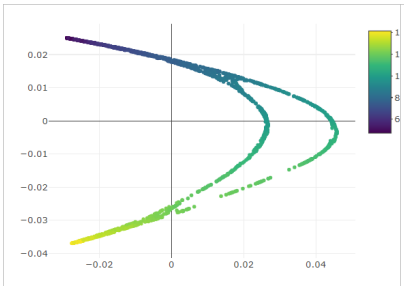
(α′) LLE αλγόριθμος με 15-κοντινότερους γείτονες



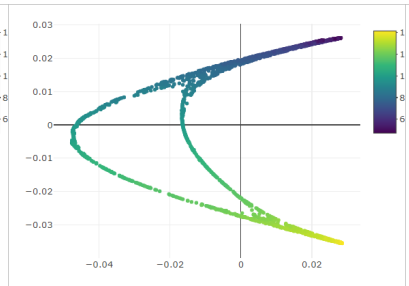
(β′) LLE αλγόριθμος με 18-κοντινότερους γείτονες

Και εδώ το LLE δίνει καλύτερα αποτελέσματα όσο μεγαλώνουμε το k. Όσο αναφορά την τρύπα που είχε το συγκεκριμένο manifold, βλέπουμε ότι δεν επηρέασε τόσο πολύ την λύση που μας έδινε ο αλγόριθμος LLE(Locally Linear Embedding). Το μόνο που έκανε είναι να παραμορφώσει και εδώ το σχήμα

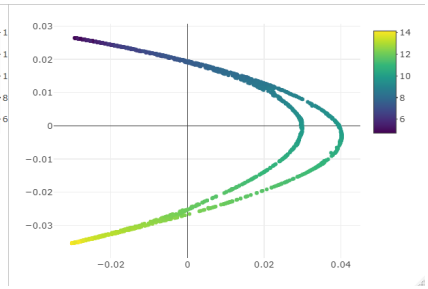
Laplacian Eigenmaps



Σχήμα 52: Laplacian Eigenmaps αλγόριθμος με 5-κοντινότερους γείτονες



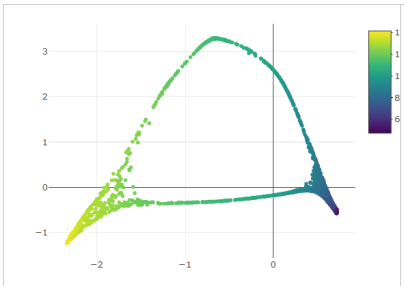
Σχήμα 53: Laplacian Eigenmaps αλγόριθμος με 7-κοντινότερους γείτονες



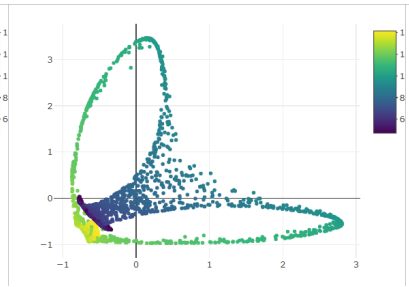
Σχήμα 54: Laplacian Eigenmaps αλγόριθμος με 10-κοντινότερους γείτονες

Όπως είπαμε και παραπάνω, ο αλγόριθμος Laplacian Eigenmaps δεν μπορεί να ξεδιπλώσει ούτε το Swiss roll αλλά ούτε και το Swiss roll με την τρύπα.

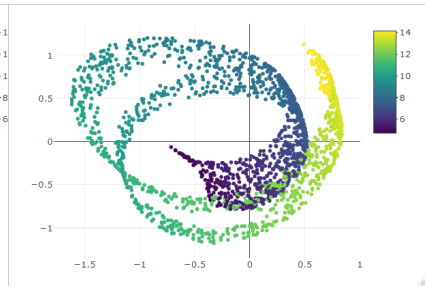
Diffusion Maps



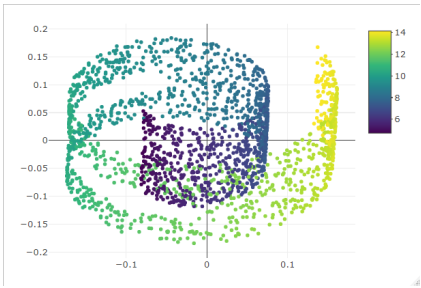
Σχήμα 55: Diffusion Maps αλγόριθμος με bandwidth = 5



Σχήμα 56: Diffusion Maps αλγόριθμος με bandwidth = 10



Σχήμα 57: Diffusion Maps αλγόριθμος με bandwidth = 100

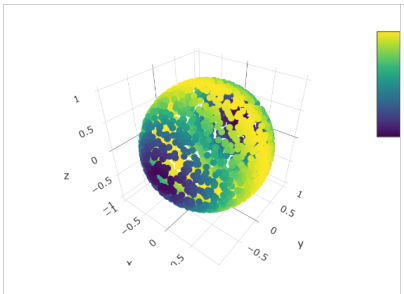


Σχήμα 58: Diffusion Maps αλγόριθμος με bandwidth = 1000

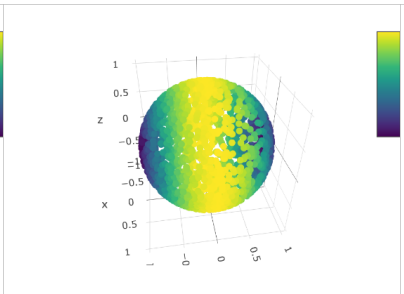
Και εδώ ο αλγόριθμος Diffusion Maps δεν μπορεί να ξεδιπλώσει το manifold.

Ένας αλγόριθμος που θα μπορούσε να ξεδιπλώσει σωστά αυτό το manifold είναι ο αλγόριθμος Hessian LLE. Όμως δεν θα τον εφαρμόσουμε διότι δεν τον έχουμε αναφέρει στα προηγούμενα κεφάλαια.

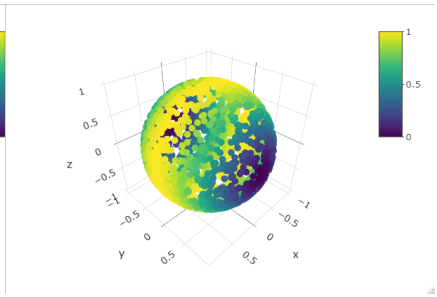
8.3 Σφαίρα



Σχήμα 59: Sphere Data 1



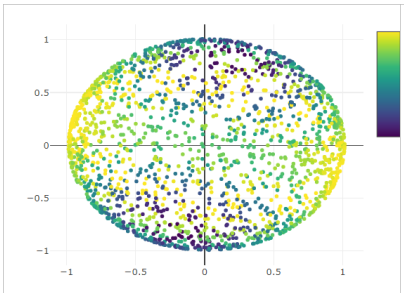
Σχήμα 60: Sphere Data 2



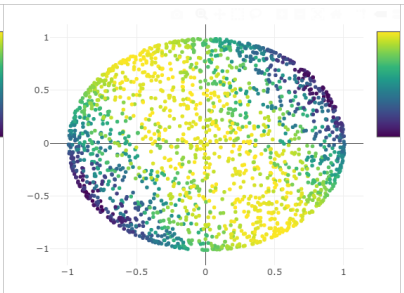
Σχήμα 61: Sphere Data 3

Εδώ θα προσπαθήσουμε να κάνουμε Dimensionality Reduction στην σφαίρα χρησιμοποιώντας τους ίδιους αλγόριθμος που έχουμε χρησιμοποιήσει στο Swiss roll(PCA, ISOMAP, LLE, Laplacian Eigenmaps, Diffusion maps).

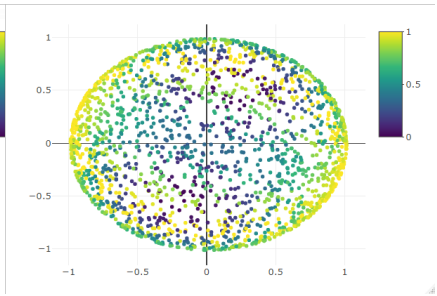
Principal Component Analysis(P.C.A)



Σχήμα 62: Sphere plot του 1ου και του 2ου component

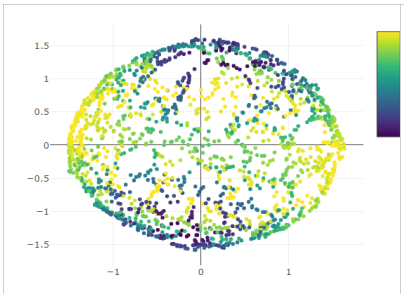


Σχήμα 63: Sphere plot του 2ου και του 3ου component

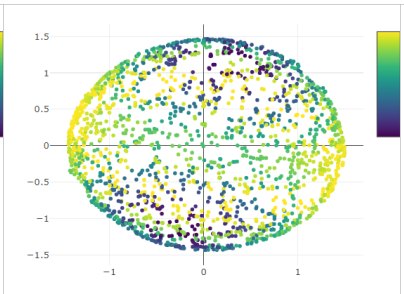


Σχήμα 64: Sphere plot του 1ου και του 3ου component

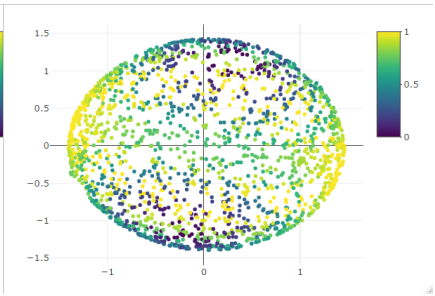
Isometric Mapping(ISOMAP)



Σχήμα 65: ISOMAP αλγόριθμος με 5-κοντινότερους γείτονες

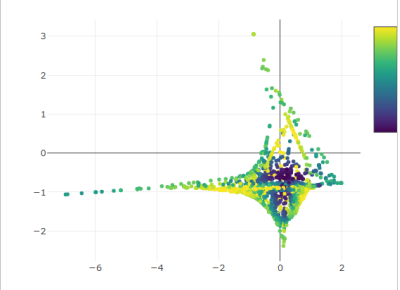


Σχήμα 66: ISOMAP αλγόριθμος με 7-κοντινότερους γείτονες

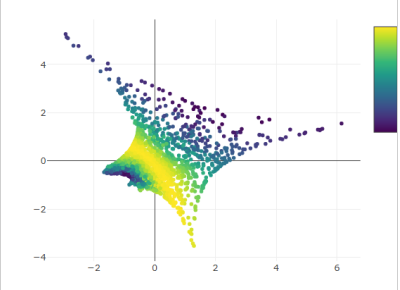


Σχήμα 67: ISOMAP αλγόριθμος με 10-κοντινότερους γείτονες

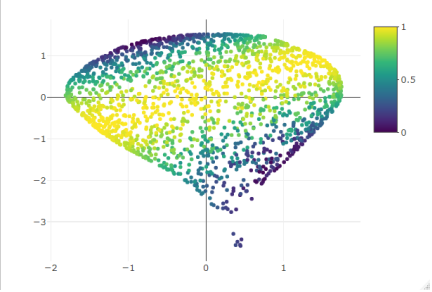
Locally Linear Embedding(LLE)



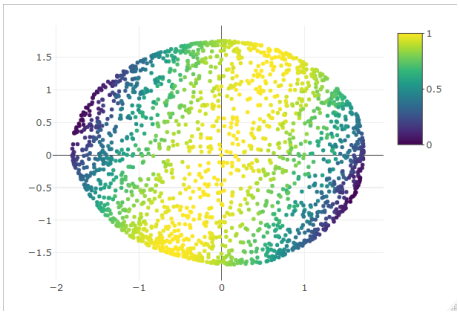
Σχήμα 68: LLE αλγόριθμος με 5-κοντινότερους γείτονες



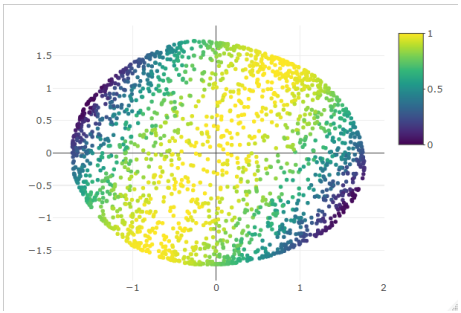
Σχήμα 69: LLE αλγόριθμος με 7-κοντινότερους γείτονες



Σχήμα 70: LLE αλγόριθμος με 10-κοντινότερους γείτονες

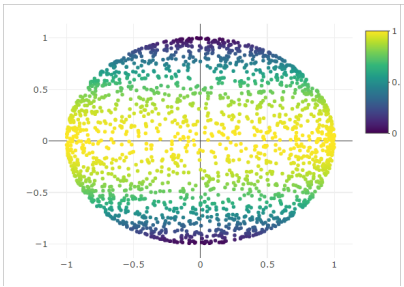


(α′) LLE αλγόριθμος με 15-κοντινότερους γείτονες

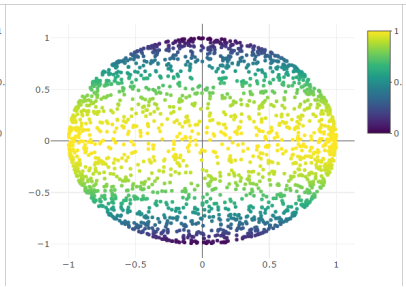


(β′) LLE αλγόριθμος με 18-κοντινότερους γείτονες

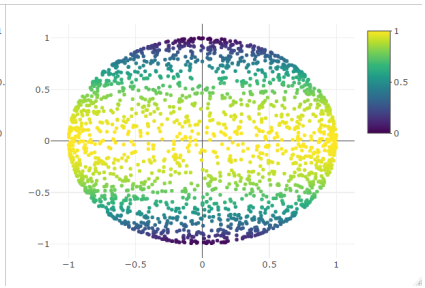
Laplacian Eigenmaps



Σχήμα 72: Laplacian Eigenmaps αλγόριθμος με 5-κοντινότερους γείτονες

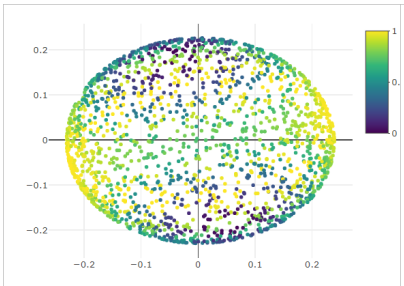


Σχήμα 73: Laplacian Eigenmaps αλγόριθμος με 7-κοντινότερους γείτονες

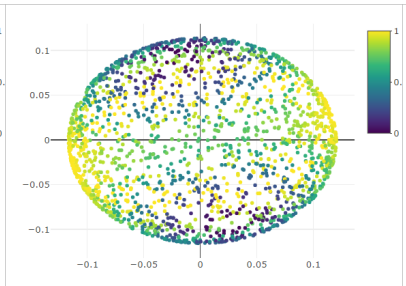


Σχήμα 74: Laplacian Eigenmaps αλγόριθμος με 10-κοντινότερους γείτονες

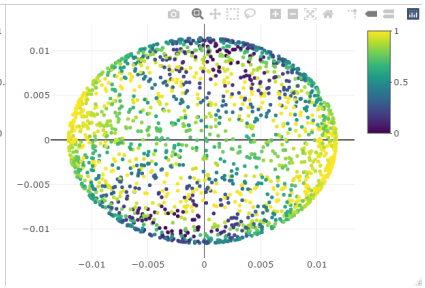
Diffusion Maps



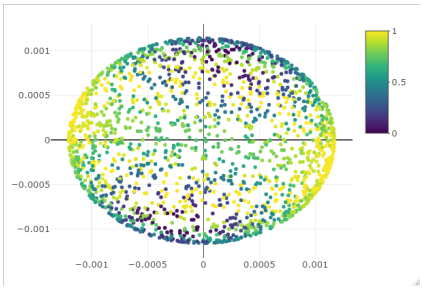
Σχήμα 75: Diffusion Maps αλγόριθμος με bandwidth = 5



Σχήμα 76: Diffusion Maps αλγόριθμος με bandwidth = 10

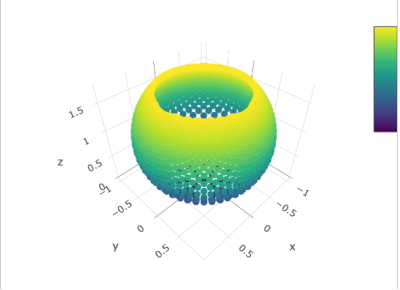


Σχήμα 77: Diffusion Maps αλγόριθμος με bandwidth = 100

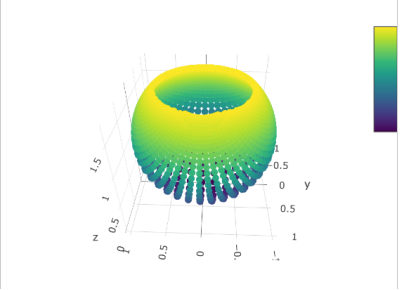


Σχήμα 78: Diffusion Maps αλγόριθμος με bandwidth = 1000

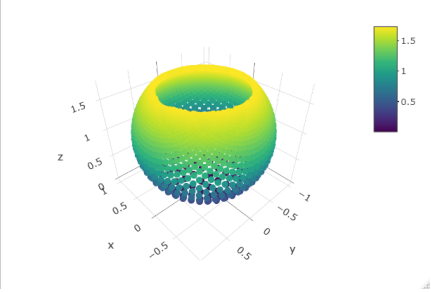
8.4 Σφαίρα με τρύπα



Σχήμα 79: Punctured Sphere Data 1

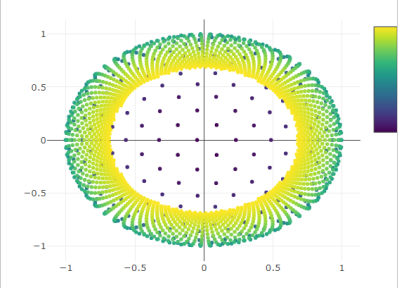


Σχήμα 80: Punctured Sphere Data 2

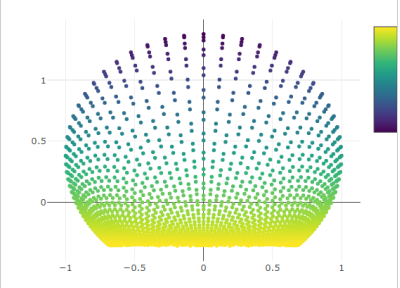


Σχήμα 81: Punctured Sphere Data 3

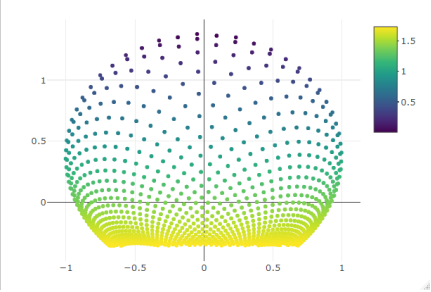
Principal Component Analysis(P.C.A)



Σχήμα 82: Punctured Sphere plot του 1ου και του 2ου component



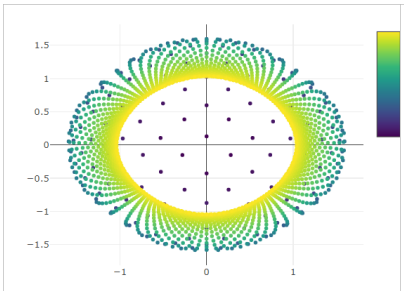
Σχήμα 83: Punctured Sphere plot του 2ου και του 3ου component



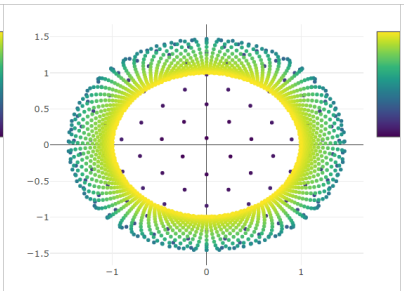
Σχήμα 84: Punctured Sphere plot του 1ου και του 3ου component

Το PCA βλέπουμε ότι δεν κατάφερε να ξεδιπλώσει το manifold.

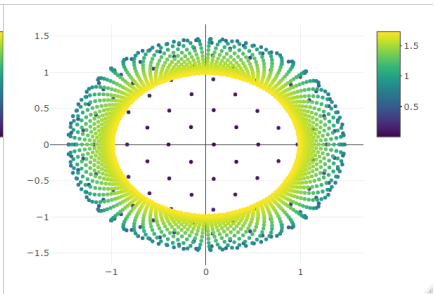
Isometric Mapping(ISOMAP)



Σχήμα 85: ISOMAP αλγόριθμος με 5-κοντινότερους γείτονες

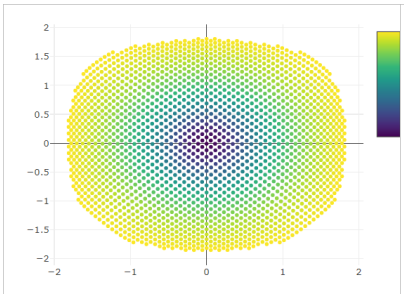


Σχήμα 86: ISOMAP αλγόριθμος με 7-κοντινότερους γείτονες

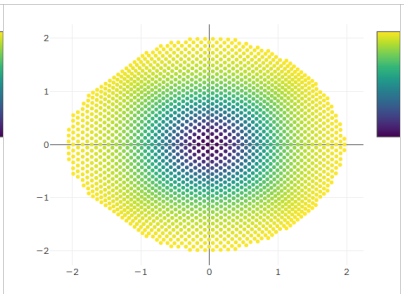


Σχήμα 87: ISOMAP αλγόριθμος με 10-κοντινότερους γείτονες

Locally Linear Embedding(LLE)

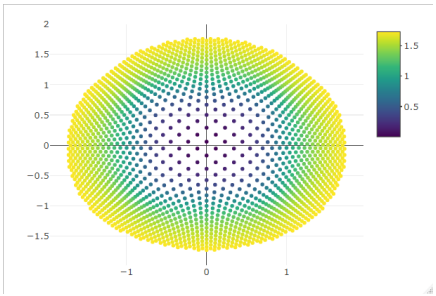


Σχήμα 88: LLE αλγόριθμος με 5-κοντινότερους γείτονες

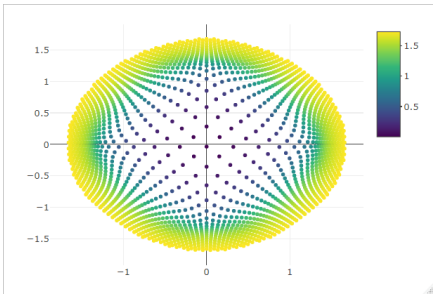


Σχήμα 89: LLE αλγόριθμος με 7-κοντινότερους γείτονες

Σχήμα 90: LLE αλγόριθμος με 10-κοντινότερους γείτονες



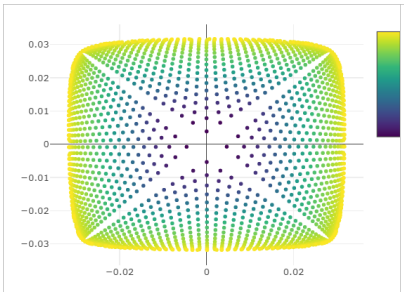
(α′) LLE αλγόριθμος με 15-κοντινότερους γείτονες



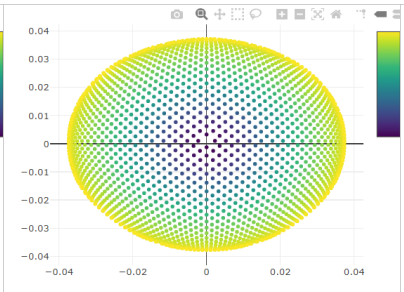
(β′) LLE αλγόριθμος με 18-κοντινότερους γείτονες

Ο αλγόριθμος Locally Linear Embedding(LLE) δίνει ένα αρκετά καλό αποτέλεσμα.

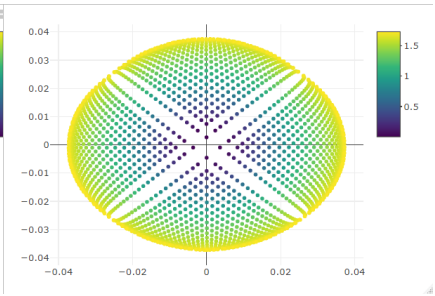
Laplacian Eigenmaps



Σχήμα 92: Laplacian Eigenmaps αλγόριθμος με 5-κοντινότερους γείτονες



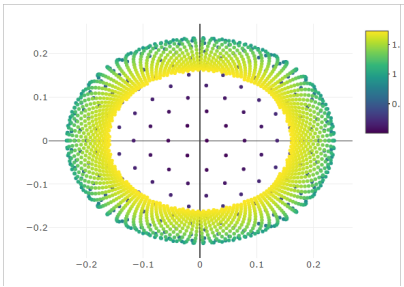
Σχήμα 93: Laplacian Eigenmaps αλγόριθμος με 7-κοντινότερους γείτονες



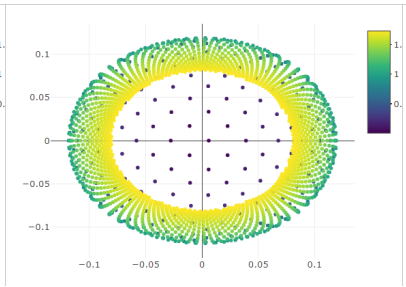
Σχήμα 94: Laplacian Eigenmaps αλγόριθμος με 10-κοντινότερους γείτονες

Ο αλγόριθμος Laplacian Eigenmaps δίνει και αυτός καλά αποτελέσματα.

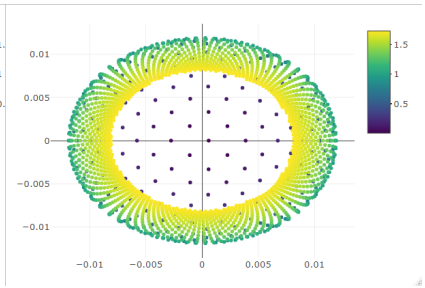
Diffusion Maps



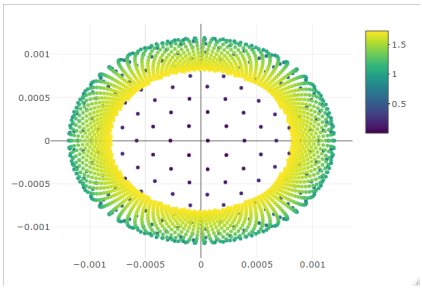
Σχήμα 95: Diffusion Maps αλγόριθμος με bandwidth = 5



Σχήμα 96: Diffusion Maps αλγόριθμος με bandwidth = 10



Σχήμα 97: Diffusion Maps αλγόριθμος με bandwidth = 100



Σχήμα 98: Diffusion Maps αλγόριθμος με bandwidth = 1000

Το Diffusion Maps δίνει την σωστή μορφή αλλά δίνει έμφαση στην sparse περιοχή στο κάτω μέρος της σφαίρας με την τρύπα.

Αναφορές

Robert A. Wijsman, 3.Differentiable Manifolds, Tangent Spaces, and Vector Fields, Institute of Mathematical Statistics Lecture Notes - Monograph Series, 1990: 38-52 (1990)

Yunqian Ma, Yun Fu, Manifold Learning Theory and Applications, CRC Press, Inc.Subs. of Times Mirror 2000 Corporate Blvd. NW Boca Raton, FLUnited States, December 2011

John Lee, Introduction to Smooth Manifolds, Springer, New York, NY (2012)

John Lee, Introduction to Topological Manifolds, Springer, New York, NY (2011)

Leonor Godinho and José Natário, An Introduction to Riemannian Geometry With Applications to Mechanics and Relativity, Lisbon 2004

Matthias Kreck,Differential Algebraic Topology: From Stratifolds to Exotic Spheres, American Mathematical Society (2010)

Steven Rosenberg, The Laplacian on a Riemannian Manifold. In The Laplacian on a Riemannian Manifold: An Introduction to Analysis on Manifolds, Cambridge University Press, (1997)

Yann Ollivier, A survey of Ricci curvature for metric spaces and Markov chains, Advanced Studies in Pure Mathematics 57, Probabilistic Approach to Geometry(2010)

Michael Spivak, A Comprehensive Introduction to Differential Geometry, Vol.1, 3rd Edition,Publish or Perish, Inc (1999)

Adams, R.A.(1975), Sobolev spaces. Academic Press, New York.

Ronald H.W. Hoppe, Finite Element Methods, Spring 2011

T. Cox, M. Cox. Multidimensional Scaling. Chapman Hall, Boca Raton, 2nd edition, 2001.

Joshua B. Tenenbaum, A Global Geometric Framework for Non-linear Dimensionality Reduction, January 2001

I. Jolliffe, Principal Component Analysis. Springer-Verlag, New York, 1986.

Bernhard Schölkopf and Alexander J. Smola, Learning with Kernels Support Vector Machines, Regularization, Optimization, and Beyond, M.I.T press, (2002).

S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. Science, 290:2323–2326, 2000.

Li Jing and Chao Shao, Selection of the Suitable Parameter Value for ISOMAP. JOURNAL OF SOFTWARE, VOL. 6, NO. 6, JUNE 2011.

Ulrike von Luxburg, A Tutorial on Spectral Clustering, August 2006

Hajime Urakawa, Spectral Geometry of the Laplacian, World Scientific , August 2017

Bernhard Schölkopf, Sebastian Mika, Alex Smola, Gunnar. Rätsch,Klaus-Robert Müller, Kernel PCA pattern reconstruction via approximate pre-images

Avner Halevy, Extensions of Laplacian Eigenmaps for Manifold Learning, 2011

Belkin, Mikhail and Niyogi, Partha. Laplacian eigenmaps and spectral techniques for embedding and clustering, 2001

Mikhail Belkin, Partha Niyogi, Laplacian Eigenmaps for Dimensionality Reduction and Data Representation, 2003

Jianbo Shi, Jitendra Malik, Normalized Cuts and Image Segmentation

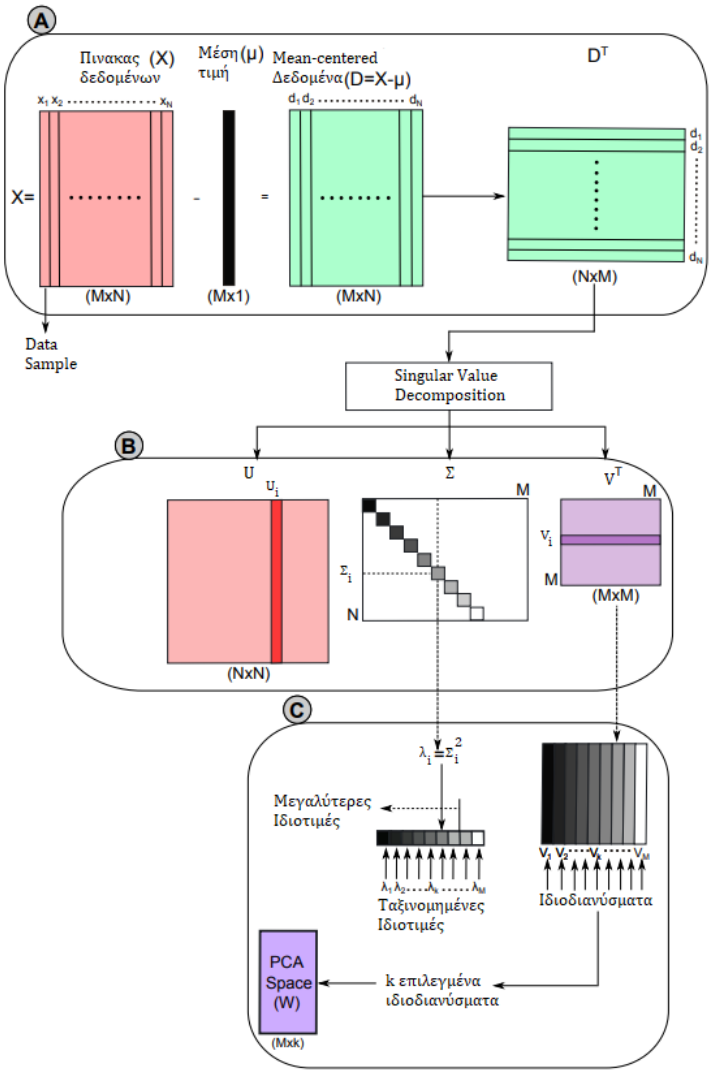
Y. S. Bhata, Gregory Arnoldb, Diffusion maps and radar data analysis, Institute for Mathematics and its Applications, University of Minnesota, Minneapolis Minnesota 55455, USA

Alaa Tharwat, Principal Component Analysis - A Tutorial, Electrical Department, Faculty of Engineering, Suez Canal University, Ismailia, Egypt

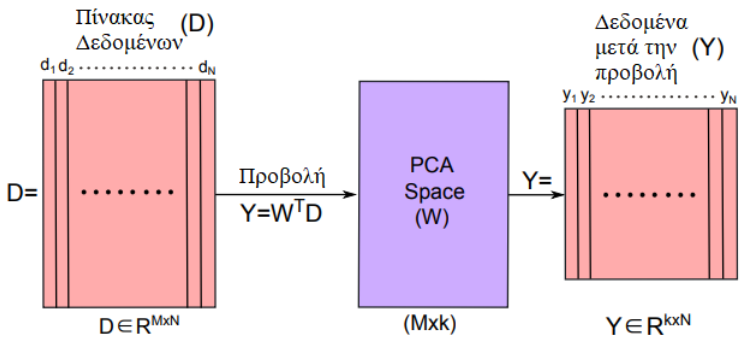
Shortest path problem(Dijkstra's algorithm),Champter 2 : `https://prodtest.pearsonschoolsandfecolleges.co.uk/secondary/Mathematics/16plus/AdvancingMathsForAQA2ndEdition/Samples/SampleMaterial/Chp-02%20023-043.pdf`

Σχήμα 7:`https://xuan-li.github.io/post/properties-of-manifolds/`

9 Παράρτημα



Σχήμα 99: http://faculty.salina.k-state.edu/tim/DAT/_downloads/Principal_Component_Analysis_Tutorial.pdf



Σχήμα 100: http://faculty.salina.k-state.edu/tim/DAT/_downloads/Principal_Component_Analysis_Tutorial.pdf

Κώδικας στην R

```
library(plot3D)
library(dimRed)
library(vegan)
library("scatterplot3d")
library(lle)
library(loe)
library(RSpectra)
library(RANN)
library(Rdimtools)
library(rgl)
library(plotly)
#remotes::install_github("kcf-jackson/maniTools")



L2_distance <- function(a, b, df) {
  if (nrow(a) == 1) {
    a = rbind(a, numeric(ncol(a)))
    b = rbind(b, numeric(ncol(b)))
  }
  aa = apply(a^2, 2, sum)  #vector
  bb = apply(b^2, 2, sum)  #vector
  ab = t(a) %*% b          #matrix

  m1 = matrix(rep(aa, length(bb)), ncol = length(bb))
  m2 = matrix(rep(bb, length(aa)), nrow = length(aa), byrow = TRUE)
  d = m1 + m2 - 2 * ab
  d = sqrt(d * (d > 0))
  if (df==1) {
    mmat = 1 - diag( max(nrow(d), ncol(d)) )
    d = d * ( mmat[1:nrow(d), 1:ncol(d)] );
  }
  d
}


leigs <- function(data, type = 'nn', param, ne) {
  n = nrow(data)
  A = Matrix::Matrix(numeric(n*n), nrow = n)
  step = 100
  for (i1 in seq(1, n, step)) {
    i2 = i1 + step - 1
    if (i2 > n) {
      i2 = n
    }
    XX = data[i1:i2, ]
    dt = L2_distance(t(XX), t(data), 0)
    Z = t(apply(dt, 1, sort))
    I = t(apply(dt, 1, order))
    for (i in i1:i2) {
      ind = i - i1 + 1
      j = 2:(param + 1)
      A[i, I[ind, j]] = Z[ind, j]
      A[I[ind, j], i] = Z[ind, j]
    }
  }
```

```
  W = A
  W[W != 0] = 1
  D = apply(W, 1, sum)
  L = Matrix::Matrix(diag(D)) - W
  E = RSpectra::eigs_sym(L, ne, "SM") #default tolerence is 1e-10
  list(eigenvectors = E$vectors, eigenvalues = E$values)
}


Laplacian_Eigenmaps <- function(X, k, d) {
  proj = leigs(X, param = k, ne = d + 1)
  list(eigenvectors = proj$eigenvectors[, 1:d],
       eigenvalues = proj$eigenvalues[1:d])
}
swiss_roll <- function(N, height = 1) {
  p = (3 * pi/2) * (1 + 2 * runif(N, 0, 1))
  y = 21 * runif(N, 0, 1)
  ret_X = cbind(p * cos(p), y, height * p * sin(p))
  ret_ColorVector = p
  list(data = ret_X, colors = ret_ColorVector)
}



swiss_roll <- function(N, height = 1) {
  p = (3 * pi/2) * (1 + 2 * runif(N, 0, 1))
  y = 21 * runif(N, 0, 1)
  ret_X = cbind(p * cos(p), y, height * p * sin(p))
  ret_ColorVector = p
  list(data = ret_X, colors = ret_ColorVector)
}



plotly_2D <- function(proj_data, colors) {
  x <- proj_data[,1]
  y <- proj_data[,2]
  if (missing(colors))
    return(plotly::plot_ly(x = x, y = y, type = "scatter", mode = "markers"
    , marker = list(size = 5)))
  plotly::plot_ly(x = x, y = y, type = "scatter", mode = "markers", color = colors,
  marker = list(size = 5))
}


plotly_3D <- function(sim_data) {
  x <- sim_data$data[,1]
  y <- sim_data$data[,2]
  z <- sim_data$data[,3]
  scale <- sim_data$colors
  plotly::plot_ly(x = x, y = y, z = z, type = "scatter3d", mode = "markers",
                  color = scale, marker = list(size = 5))
}

plotly_3D1 <- function(sim_data,color) {
  x <- sim_data[,1]
  y <- sim_data[,2]
  z <- sim_data[,3]
```

```
  scale <- color
  if (missing(color))
    return(plotly::plot_ly(x = x, y = y, z = z, type = "scatter", mode = "markers",
    marker = list(size = 5)))
  plotly::plot_ly(x = x, y = y, z = z, type = "scatter3d", mode = "markers",
                  color = scale, marker = list(size = 5))
}

swiss_hole <- function(N, height = 1) {
  p = (3 * pi/2) * (1 + 2 * runif(2*N, 0, 1))
  y = 21 * runif(2*N, 0, 1)
  kl = rep(0, 2 * N)
  assign_one = (p > 9) & (p < 12) & (y > 9) & (y < 14)
  kl[assign_one] = 1
  kkz = which(!assign_one)
  p = p[kkz[1:N]]
  y = y[kkz[1:N]]
  ret_X = cbind(p * cos(p), y, height * p * sin(p))
  ret_ColorVector = p
  list(data = ret_X, colors = ret_ColorVector)
}

punctured_sphere <- function(N, z_scale = 1.0) {
  inc = 9 / sqrt(N);
  s = seq(-5, 5, by = inc)
  yy = matrix(rep(s, length(s)), nrow = length(s))
  xx = t(yy)
  rr2 = as.vector(xx)^2 + as.vector(yy)^2
  ii = order(rr2)
  Y = rbind(xx[ii[1:N]], yy[ii[1:N]])
  a = 4 / ( 4 + apply(Y^2, MARGIN = 2, sum) )
  X = cbind(a * Y[1,], a * Y[2,], z_scale * 2 * (1 - a))
  list(data = X, colors = X[,3])
}



#Swiss Roll
sim_data1 <- swiss_roll(N = 2000)

p1 <- plotly_3D(sim_data1); p1


#pca
dimred2 <-  prcomp(sim_data1$data)

plotly_2D(dimred2$x[,c(1,2)], color = sim_data1$colors)

plotly_2D(dimred2$x[,c(2,3)], color = sim_data1$colors)

plotly_2D(dimred2$x[,c(1,3)], color = sim_data1$colors)


#isomap k=5
i <-dist(sim_data1$data)
dimred =isomap(i,ndim=2,k=5)
```

```
plotly_2D(dimred$points, color = sim_data1$colors)

#isomap k=7
i <-dist(sim_data1$data)
dimred =isomap(i,ndim=2,k=7)

plotly_2D(dimred$points, color = sim_data1$colors)

#isomap k=10
i <-dist(sim_data1$data)
dimred =isomap(i,ndim=2,k=10)

plotly_2D(dimred$points, color = sim_data1$colors)


#lle k=5
dimred1 = lle(sim_data1$data,m=2,k=5)

plotly_2D(dimred1$Y, color = sim_data1$colors)


#lle k=7
dimred1 = lle(sim_data1$data,m=2,k=7)

plotly_2D(dimred1$Y, color = sim_data1$colors)


#lle k=10
dimred1 = lle(sim_data1$data,m=2,k=10)

plotly_2D(dimred1$Y, color = sim_data1$colors)


#lle k=15
dimred1 = lle(sim_data1$data,m=2,k=15)

plotly_2D(dimred1$Y, color = sim_data1$colors)


#lle k=18
dimred1 = lle(sim_data1$data,m=2,k=18)

plotly_2D(dimred1$Y, color = sim_data1$colors)



#Laplacian Eigenmaps k=5
LE_data <- Laplacian_Eigenmaps(sim_data1$data, k = 5, d = 2)
p2 <- plotly_2D(LE_data$eigenvectors, color = sim_data1$colors); p2


#Laplacian Eigenmaps k=7
LE_data <- Laplacian_Eigenmaps(sim_data1$data, k = 7, d = 2)
p2 <- plotly_2D(LE_data$eigenvectors, color = sim_data1$colors); p2


#Laplacian Eigenmaps k=10
LE_data <- Laplacian_Eigenmaps(sim_data1$data, k = 10, d = 2)
```

```
p2 <- plotly_2D(LE_data$eigenvectors, color = sim_data1$colors); p2


#Diffusion Maps bandwidth=5
dimred3 <-do.dm(sim_data1$data,ndim=2,bandwidth = 5)

plotly_2D(dimred3$Y, color = sim_data1$colors)


#Diffusion Maps bandwidth=10
dimred3 <-do.dm(sim_data1$data,ndim=2,bandwidth = 10)

plotly_2D(dimred3$Y, color = sim_data1$colors)


#Diffusion Maps bandwidth=100
dimred3 <-do.dm(sim_data1$data,ndim=2,bandwidth = 100)

plotly_2D(dimred3$Y, color = sim_data1$colors)

#Diffusion Maps bandwidth=1000
dimred3 <-do.dm(sim_data1$data,ndim=2,bandwidth = 1000)

plotly_2D(dimred3$Y, color = sim_data1$colors)





#swiss_hole


sim_data2 <- swiss_hole(N = 2000)



p2 <- plotly_3D(sim_data2); p2


#pca
dimred2 <-  prcomp(sim_data2$data)

plotly_2D(dimred2$x[,c(1,2)], color = sim_data2$colors)

plotly_2D(dimred2$x[,c(2,3)], color = sim_data2$colors)

plotly_2D(dimred2$x[,c(1,3)], color = sim_data2$colors)



#isomap k=5
i <-dist(sim_data2$data)
dimred =isomap(i,ndim=2,k=5)

plotly_2D(dimred$points, color = sim_data2$colors)

#isomap k=7
```

```
i <-dist(sim_data2$data)
dimred =isomap(i,ndim=2,k=7)

plotly_2D(dimred$points, color = sim_data2$colors)

#isomap k=10
i <-dist(sim_data2$data)
dimred =isomap(i,ndim=2,k=10)

plotly_2D(dimred$points, color = sim_data2$colors)


#lle k=5
dimred1 = lle(sim_data2$data,m=2,k=5)

plotly_2D(dimred1$Y, color = sim_data2$colors)


#lle k=7
dimred1 = lle(sim_data2$data,m=2,k=7)

plotly_2D(dimred1$Y, color = sim_data2$colors)


#lle k=10
dimred1 = lle(sim_data2$data,m=2,k=10)

plotly_2D(dimred1$Y, color = sim_data2$colors)


#lle k=15
dimred1 = lle(sim_data2$data,m=2,k=15)

plotly_2D(dimred1$Y, color = sim_data2$colors)

#lle k=18
dimred1 = lle(sim_data2$data,m=2,k=18)

plotly_2D(dimred1$Y, color = sim_data2$colors)



#Laplacian Eigenmaps k=5
LE_data <- Laplacian_Eigenmaps(sim_data2$data, k = 5, d = 2)
p2 <- plotly_2D(LE_data$eigenvectors, color = sim_data2$colors); p2


#Laplacian Eigenmaps k=7
LE_data <- Laplacian_Eigenmaps(sim_data2$data, k = 7, d = 2)
p2 <- plotly_2D(LE_data$eigenvectors, color = sim_data2$colors); p2

#Laplacian Eigenmaps k=10
LE_data <- Laplacian_Eigenmaps(sim_data2$data, k = 10, d = 2)
p2 <- plotly_2D(LE_data$eigenvectors, color = sim_data2$colors); p2


#Diffusion Maps bandwidth=5
dimred3 <-do.dm(sim_data2$data,ndim=2,bandwidth = 5)
```

```
plotly_2D(dimred3$Y, color = sim_data2$colors)

#Diffusion Maps bandwidth=10
dimred3 <-do.dm(sim_data2$data,ndim=2,bandwidth = 10)

plotly_2D(dimred3$Y, color = sim_data2$colors)

#Diffusion Maps bandwidth=100
dimred3 <-do.dm(sim_data2$data,ndim=2,bandwidth = 100)

plotly_2D(dimred3$Y, color = sim_data2$colors)

#Diffusion Maps bandwidth=1000
dimred3 <-do.dm(sim_data2$data,ndim=2,bandwidth = 1000)

plotly_2D(dimred3$Y, color = sim_data2$colors)









#punctured_sphere

sim_data3 <- punctured_sphere(N = 2000)



p2 <- plotly_3D(sim_data3); p2


#pca
dimred2 <-  prcomp(sim_data3$data)

plotly_2D(dimred2$x[,c(1,2)], color = sim_data3$colors)

plotly_2D(dimred2$x[,c(2,3)], color = sim_data3$colors)

plotly_2D(dimred2$x[,c(1,3)], color = sim_data3$colors)


#isomap k=5
i <-dist(sim_data3$data)
dimred =isomap(i,ndim=2,k=5)

plotly_2D(dimred$points, color = sim_data3$colors)

#isomap k=7
i <-dist(sim_data3$data)
dimred =isomap(i,ndim=2,k=10)

plotly_2D(dimred$points, color = sim_data3$colors)
```

```
#isomap k=10
i <-dist(sim_data3$data)
dimred =isomap(i,ndim=2,k=10)

plotly_2D(dimred$points, color = sim_data3$colors)


#lle k=5
dimred1 = lle(sim_data3$data,m=2,k=5)

plotly_2D(dimred1$Y, color = sim_data3$colors)

#lle k=7
dimred1 = lle(sim_data3$data,m=2,k=7)

plotly_2D(dimred1$Y, color = sim_data3$colors)

#lle k=10
dimred1 = lle(sim_data3$data,m=2,k=10)

plotly_2D(dimred1$Y, color = sim_data3$colors)

#lle k=15
dimred1 = lle(sim_data3$data,m=2,k=15)

plotly_2D(dimred1$Y, color = sim_data3$colors)

#lle k=18
dimred1 = lle(sim_data3$data,m=2,k=18)

plotly_2D(dimred1$Y, color = sim_data3$colors)


#Laplacian Eigenmaps k=5
LE_data <- Laplacian_Eigenmaps(sim_data3$data, k = 5, d = 2)
p2 <- plotly_2D(LE_data$eigenvectors, color = sim_data3$colors); p2

#Laplacian Eigenmaps k=7
LE_data <- Laplacian_Eigenmaps(sim_data3$data, k = 7, d = 2)
p2 <- plotly_2D(LE_data$eigenvectors, color = sim_data3$colors); p2

#Laplacian Eigenmaps k=10
LE_data <- Laplacian_Eigenmaps(sim_data3$data, k = 10, d = 2)
p2 <- plotly_2D(LE_data$eigenvectors, color = sim_data3$colors); p2



#Diffusion Maps  bandwidth=5
dimred3 <-do.dm(sim_data3$data,ndim=2,bandwidth = 5)

plotly_2D(dimred3$Y, color = sim_data3$colors)


#Diffusion Maps bandwidth=10
dimred3 <-do.dm(sim_data3$data,ndim=2,bandwidth = 10)

plotly_2D(dimred3$Y, color = sim_data3$colors)
```

```
#Diffusion Maps bandwidth=100
dimred3 <-do.dm(sim_data3$data,ndim=2,bandwidth = 100)

plotly_2D(dimred3$Y, color = sim_data3$colors)

#Diffusion Maps bandwidth=1000
dimred3 <-do.dm(sim_data3$data,ndim=2,bandwidth = 1000)

plotly_2D(dimred3$Y, color = sim_data3$colors)


#sphere
n <- 2000
theta <- runif(n,0,2*pi)
u <- runif(n,-1,1)
x <- sqrt(1-u^2)*cos(theta)
y <- sqrt(1-u^2)*sin(theta)
z <- u
uy = x^2+z^2 # squared Euclidean distance
uy = (uy-min(uy))/(max(uy)-min(uy)) # so will be between 0 and 1

p1 <- plotly_3D1(cbind(x,y,z), color = uy); p1



#pca
dimred2 <-  prcomp(cbind(x,y,z))

plotly_2D(dimred2$x[,c(1,2)], color = sim_data3$colors)

plotly_2D(dimred2$x[,c(2,3)], color = sim_data3$colors)

plotly_2D(dimred2$x[,c(1,3)], color = sim_data3$colors)


#isomap k=5
i <-dist(cbind(x,y,z))
dimred =isomap(i,ndim=2,k=5)

plotly_2D(dimred$points, color = uy)


#isomap k=7
i <-dist(cbind(x,y,z))
dimred =isomap(i,ndim=2,k=7)

plotly_2D(dimred$points, color = uy)

#isomap k=10
i <-dist(cbind(x,y,z))
dimred =isomap(i,ndim=2,k=10)

plotly_2D(dimred$points, color = uy)


#lle k=5
```

```
dimred1 = lle(cbind(x,y,z),m=2,k=5)


plotly_2D(dimred1$Y, color = uy)


#lle k=7
dimred1 = lle(cbind(x,y,z),m=2,k=7)


plotly_2D(dimred1$Y, color = uy)


#lle k=10
dimred1 = lle(cbind(x,y,z),m=2,k=10)


plotly_2D(dimred1$Y, color = uy)


#lle k=15
dimred1 = lle(cbind(x,y,z),m=2,k=15)


plotly_2D(dimred1$Y, color = uy)

#lle k=18
dimred1 = lle(cbind(x,y,z),m=2,k=18)


plotly_2D(dimred1$Y, color = uy)



#Laplacian Eigenmaps k=5
LE_data <- Laplacian_Eigenmaps(cbind(x,y,z), k = 5, d = 2)
p2 <- plotly_2D(cbind(x,y,z), color = uy); p2


#Laplacian Eigenmaps k=7
LE_data <- Laplacian_Eigenmaps(cbind(x,y,z), k = 7, d = 2)
p2 <- plotly_2D(cbind(x,y,z), color = uy); p2


#Laplacian Eigenmaps k=10
LE_data <- Laplacian_Eigenmaps(cbind(x,y,z), k = 10, d = 2)
p2 <- plotly_2D(cbind(x,y,z), color = uy); p2


#Diffusion maps bandwidth=5
dimred3 <-do.dm(cbind(x,y,z),ndim=2,bandwidth = 5)

plotly_2D(dimred3$Y, color = uy)

#Diffusion maps bandwidth=10
dimred3 <-do.dm(cbind(x,y,z),ndim=2,bandwidth = 10)

plotly_2D(dimred3$Y, color = uy)
```

```
#Diffusion maps bandwidth=100
dimred3 <-do.dm(cbind(x,y,z),ndim=2,bandwidth = 100)

plotly_2D(dimred3$Y, color = uy)

#Diffusion maps bandwidth=1000
dimred3 <-do.dm(cbind(x,y,z),ndim=2,bandwidth = 1000)

plotly_2D(dimred3$Y, color = uy)
```