

Метод Isomap

Isomap (*Isometric feature mapping*) относится к нелинейным методам уменьшения размерности исходного описания исследуемого объекта, здесь предполагается, что данные расположены на нелинейной поверхности меньшей размерности, чем исходное пространство. *Isomap* используется для вычисления квазиизометрического низкоразмерного вложения набора многомерных точек данных. Алгоритм обеспечивает простой метод оценки внутренней геометрии многообразия данных, основанный на грубой оценке соседей каждой точки данных на многообразии. *Isomap* очень эффективен и обычно применим к широкому спектру источников данных и размерностей.

Введение

В целях преодоления ограниченности линейных подходов был разработан ряд техник для применения к данным с различными характеристиками структурами. Данные техники неконтролируемого уменьшения размерности используют обучение на основе многообразий. Обучение на основе многообразий стремится обучать многообразия, представленные гладкими, изогнутыми поверхностями (*manifold*) внутри многомерного пространства. После чего можно извлечь "локальные линейные вставки" которые касательны к многообразию. Как правило, для точного представления данного многообразия этих вставок вполне достаточно. Поскольку подобные многообразия моделируются не какой-либо одной математической функцией, а несколькими небольшими линейными участками (патчами), такие линейные соседи способны моделировать любое многообразие.

Фундаментальные аспекты алгоритмов обучения на основе многообразий:

- В данных существуют нелинейные отношения, которые можно моделировать при помощи многообразий ?? гладких и не особо изогнутых (не очень сложных) поверхностей, охватывающих несколько измерений. Многообразия непрерывны.

- Вместо многомерной формы данных их проекции по определенным направлениям для сохранения общей формы, можно выполнить более сложные манипуляции, наподобие разворачивания скрученной полосы или выворачивания сферы наружу.

- Наилучший метод моделирования многообразий ?—? это представление изогнутого пространства как составленного из нескольких соседних частей. Если каждая точка данных сможет сохранить расстояние не

до всех остальных точек, а только до ближайших к ней, то геометрические отношения в таких данных можно удержать.

Одним из первых исследований в области обучения на основе многообразий был алгоритм *Isomap*, чье название расшифровывается как изометрическое отображение. Он расширяет метрическое многомерное масштабирование (*MDS*) за счет включения геодезических расстояний, налагаемых взвешенным графом. Чтобы быть конкретным, классическое масштабирование метрической *MDS* выполняет низкоразмерное встраивание на основе попарного расстояния между точками данных, которое обычно измеряется с использованием прямого евклидова расстояния. *Isomap* отличается использованием геодезического расстояния, индуцированного графом окрестностей, вложенным в классическое масштабирование. Геодезическое расстояние — это обобщение расстояния для изогнутых поверхностей. Это сделано для включения структуры коллектора в результирующее вложение. *Isomap* определяет геодезическое расстояние как сумму весов ребер на кратчайшем пути между двумя узлами (вычисляется с использованием алгоритма Дейкстры, например). Таким образом, *IsoMap* оптимизирует расстояния вдоль обнаруживаемого многообразия, стремясь сохранить геодезическое расстояние или расстояние, измеряемое не в Евклидовом пространстве, а на изогнутой поверхности многообразия. При этом, Верхние n собственных векторов матрицы геодезических расстояний представляют координаты в новом n - мерном евклидовом пространстве. Другими словами в исходное многомерное пространства данных размерности N встраивается коллектор (manifold) размерности n ($n < N$). Коллектор обычно определяется как пространство, которое локально является гомеоморфным к евклидовому пространству (наряду с некоторыми другими техническими условиями), коллектор не включает в себя его границу, когда он встраивается в большее пространство.

Общее описание алгоритма Isomap .

1. Определить соседей каждой точки. Все точки в некотором фиксированном радиусе. K ближайших соседей.
2. Построить граф окрестностей. Каждая точка соединена с другой, если это K ближайшего соседа. Длина ребра равна евклидову расстоянию.
3. Вычислить кратчайший путь между двумя узлами. Используя алгоритм Дейкстры (Флойда – Уоршола)
4. Вычислить низкоразмерное вложение, используя метод многомер-

ного масштабирования (*MDS*).

Расширения *Isomap*

- Landmark *ISOMAP* (*LISOMAP*) : *Landmark – Isomap* - это вариант *Isomap*, который работает быстрее, чем *Isomap*. Однако точность коллектора снижается из-за незначительного фактора. В этом алгоритме используется $n \ll N$ контрольных точек из общего количества N точек данных, и вычисляется матрица $n \times N$ геодезического расстояния между каждой точкой данных и контрольными точками. Затем к матрице применяется Landmark-*MDS* (*LMDS*), чтобы найти евклидово вложение всех точек данных.

- *CIsoMap* включает увеличение областей с высокой плотностью и сжатие областей с низкой плотностью точек данных в коллекторе. Веса краев, которые максимизированы в многомерном масштабировании (*MDS*), изменяются, а все остальное остается неизменным.

- Развертывание параллельного переноса: заменяет оценки геодезических расстояний на основе путей Дейкстры на приближения на основе параллельного переноса, повышая устойчивость к нерегулярности и пустотам в выборке.

Возможные проблемы

Связность каждой точки данных в графе соседства определяется как ее ближайшие k евклидовых соседей в многомерном пространстве. Этот шаг уязвим для «ошибок короткого замыкания», если k слишком велико по отношению к структуре коллектора или если шум в данных немного сдвигает точки за пределы коллектора. Даже одна ошибка короткого замыкания может изменить многие элементы в матрице геодезических расстояний, что, в свою очередь, может привести к совершенно иному (и неправильному) низкоразмерному встраиванию. Наоборот, если k слишком мал, граф окрестности может стать слишком разреженным для точной аппроксимации геодезических путей. Но в этот алгоритм были внесены улучшения, чтобы он лучше работал с разреженными и зашумленными наборами данных.

[originalsource](#)

Пример реализации алгоритма *Isomap* на *python* и *R*.

Для расчетов использовались данные о Российских регионах. В *python* реализации исходным было 3D пространство, регионы размечены на три класса. Для *R* реализации исходное пространство 4D, а регионы размечены на четыре класса.

Код *python* :

```

'''python
from mpl_toolkits.mplot3d import Axes3D
import numpy as np
import scipy.stats as st
import matplotlib.pyplot as plt
from pandas import Series, DataFrame
import pandas as pd
#import umap
#import umap.plot
from sklearn.manifold import Isomap
result =pd.read_table('dan04.txt',sep='\s+')
print(result)
X =DataFrame(result, columns=['P10','P11','P14'])
X=np.array (X)
SUMX=X[:,0] +X[:,1] + X[:,2]
X[:,0] =X[:,0] /SUMX
X[:,1] =X[:,1] /SUMX
X[:,2] =X[:,2] /SUMX

YT=result['P18']
y=np.array(YT)
XX1=result.query("P18 in [0]")
XX2=result.query("P18 in [1]")
XX3=result.query("P18 in [2]")
XX1 =DataFrame(XX1, columns=['P10','P11','P14'])
XX1=np.array (XX1)
XX2 =DataFrame(XX2, columns=['P10','P11','P14'])
XX2=np.array (XX2)
XX3 =DataFrame(XX3, columns=['P10','P11','P14'])
XX3=np.array (XX3)

SUMX1=XX1[:,0] +XX1[:,1] + XX1[:,2]
XX1[:,0] =XX1[:,0] /SUMX1
XX1[:,1] =XX1[:,1] /SUMX1
XX1[:,2] =XX1[:,2] /SUMX1
SUMX2=XX2[:,0] +XX2[:,1] + XX2[:,2]
XX2[:,0] =XX2[:,0] /SUMX2
XX2[:,1] =XX2[:,1] /SUMX2

```

```

XX2[:,2] =XX2[:,2] /SUMX2
SUMX3=XX3[:,0] +XX3[:,1] + XX3[:,2]
XX3[:,0] =XX3[:,0] /SUMX3
XX3[:,1] =XX3[:,1] /SUMX3
XX3[:,2] =XX3[:,2] /SUMX3

# Преобразование данных, уменьшающее размерность до 2
#задаются:число ближайших соседей и размерность коллектора
mod_iso = Isomap(n_neighbors =5,n_components=2)
X_iso = mod_iso.fit_transform(X)
X_iso.shape

XI =DataFrame(X_iso, columns=['1','2'])
XI = XI.join(YT)

XI1=XI.query("P18 in [0]")
XI2=XI.query("P18 in [1]")
XI3=XI.query("P18 in [2]")

XI1 =DataFrame(XI1, columns=['1','2'])
XI1=np.array (XI1)
XI2 =DataFrame(XI2, columns=['1','2'])
XI2=np.array (XI2)
XI3 =DataFrame(XI3, columns=['1','2'])
XI3=np.array (XI3)

#построение исходного графика 3 D
fig = plt.figure(1, figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')

xs1 = XX1[:,0]
ys1 = XX1[:,1]
zs1 = XX1[:,2]

xs2 = XX2[:,0]
ys2 = XX2[:,1]
zs2 = XX2[:,2]

```

```

xs3 = XX3[:,0]
ys3 = XX3[:,1]
zs3 = XX3[:,2]

ax.scatter(xs1, ys1, zs1,c='g', cmap=plt.cm.nipy_spectral,
edgecolor='k',s=80,label='A')
ax.scatter(xs2, ys2, zs2,c='r', cmap=plt.cm.nipy_spectral,
edgecolor='k',s=80,label='P')
ax.scatter(xs3, ys3, zs3,c='b', cmap=plt.cm.nipy_spectral,
edgecolor='k',s=80,label='D')
ax.set_title("Исходное пространство предикторов",
fontsize=16, fontweight='bold')
ax.set_xlabel('P10',fontsize=14, fontweight='bold')
ax.set_ylabel('P11',fontsize=14, fontweight='bold')
ax.set_zlabel('P14',fontsize=14, fontweight='bold')
ax.legend(loc="best")

# Построение редуцированного пространства 2D
y = np.choose(y, [1, 2, 0]).astype(np.float)
fig = plt.figure(2,figsize=(10, 8))
plt.clf()
plt.cla()
plt.scatter(XI1[:, 0], XI1[:, 1], c='g', cmap=plt.cm.nipy_spectral,
edgecolor='k',s=100,label='A')
plt.scatter(XI2[:, 0], XI2[:, 1], c='r', cmap=plt.cm.nipy_spectral,
edgecolor='k',s=100,label='P')
plt.scatter(XI3[:, 0], XI3[:, 1], c='b', cmap=plt.cm.nipy_spectral,
edgecolor='k',s=100,label='D')
plt.title("Снижение размерности методом ISOMAP",
fontsize=16, fontweight='bold')
plt.xlabel("Isomap1",fontsize=14, fontweight='bold')
plt.ylabel("Isomap2",fontsize=14, fontweight='bold')
plt.legend(loc='best')
plt.show()

'''

```

Графические результаты:

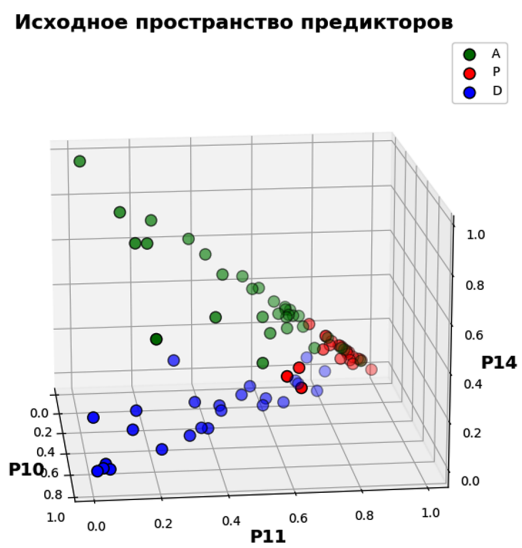


Рис. 1: Исходное 3D пространство описания регионов

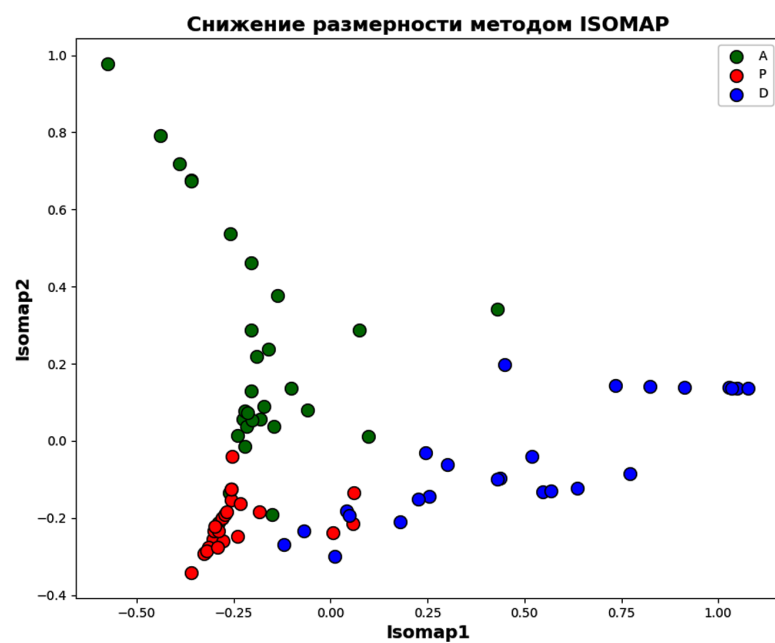


Рис. 2: Редуцированное 2D пространство ISOMAP

Код R :

```
““r
library(gmodels)
library(lattice)
library(vegan)
library(grDevices)
library(reshape2)
library(ggplot2)
library(ggthemes)
#library(fastICA)
library(psych)

load("xRegion.RData")

#ФОРМИРОВАНИЕ data.frame ОСНОВНЫЕ
#СОЦИАЛЬНО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ РЕГИОНОВ РФ
xreg <-data.frame(P1=x$Pok_001,P2=x$Pok_002,
P3=x$Pok_003,P4=x$Pok_004,
P5=x$Pok_005,P6=x$Pok_006,P7=x$Pok_007,
P8=x$Pok_008,P9=x$Pok_009,P10=x$Pok_010,
P11=x$Pok_011,P12=x$Pok_012,P13=x$Pok_013,
P14=x$Pok_014,P15=x$Pok_015,P16=x$Pok_016,
P17=x$Pok_017,P18=x$Pok_018)
#xreg
#Формирование фактора тип экономики региона
xreg_f <- as.factor(x$Pok_018)

#выбор исследуемого подмножества показателей регионов
w=c(10,11,14,16)
reg.ekon <-xreg[,w]
valreg <-rowSums(reg.ekon)
xreg_nor <-reg.ekon/valreg
Y <- xreg_nor
X=Y
X <- as.matrix(X)

#Метод isomap=====
```

```

dis <- vegdist(X)
iso <- isomap(dis, ndim=2, k=5)
y.iso <- data.frame(iso[1])
yx.iso <- data.frame( iso1=y.iso[,1], iso2=y.iso[,2], Econ=xreg_f)

#График регионов по классам в двухмерном пространстве isomap
x11()
g4 <- ggplot(data= yx.iso, aes(x=iso1, y=iso2, colour = Econ),
  colors = "Accent")
g4 <- g4 + geom_point(size=6)
g4 <- g4 + labs(title = "Регионы в пространстве двух компонент isomap",
  x="iso1", y="iso2", colour = "Классы")
#g4 <- g4 + theme_stata() + scale_colour_stata()
g4 <- g4 + theme_fivethirtyeight() + scale_colour_manual( values = c('purple',
  'blue', 'red', 'green'))
g4

'''

```

Графические результаты:

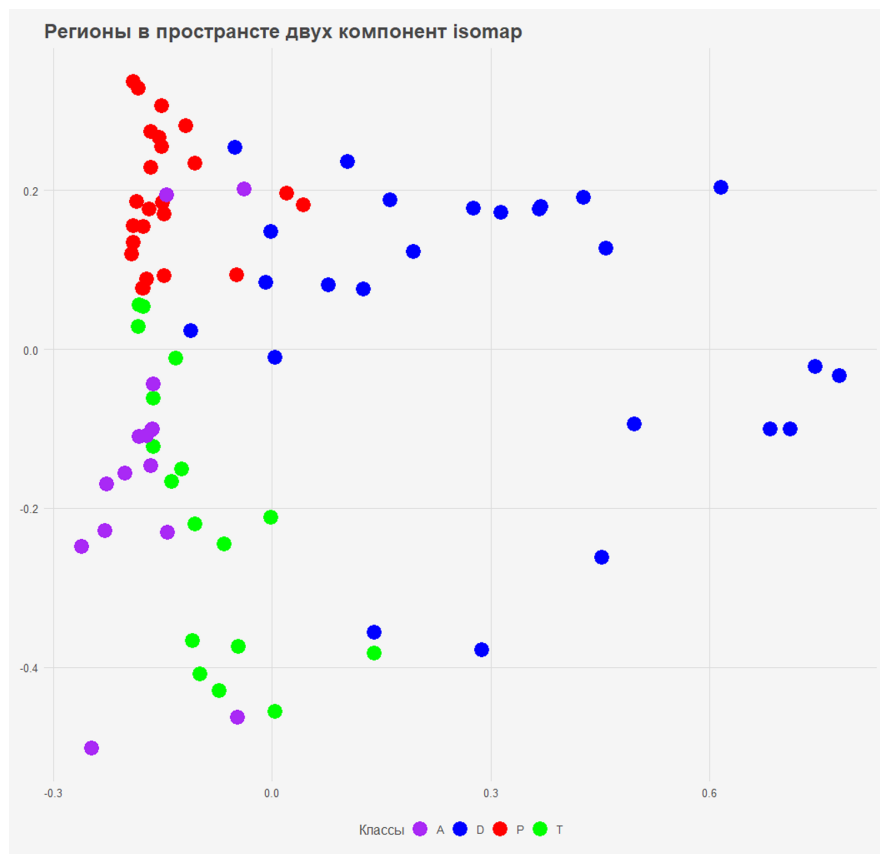


Рис. 3: Редуцированное 2D пространство ISOMAP