Московский государственный университет имени М.В. Ломоносова

Факультет вычислительной математики и кибернетики

Кафедра математических методов прогнозирования

ГРАФИК В ПАРАЛЛЕЛЬНЫХ ОСЯХ И ЕГО ПРИЛОЖЕНИЯ

Семестровый отчет студента 3 курса
Тыцкого Владислава

Научный руководитель
к.ф.-м.н. доцент Майсурадзе А.И.

Москва 2020

Аннотация

В данном отчете представлен обзор графика в параллельных осях(parallel coordinates) и его модификаций, описание собственной библиотеки для построения данных графиков и итоги работы над ней.

Введение

В анализе данных важной частью любого исследования является представление данных[1] в наглядной для человека форме. Это необходимо не только для самого исследователя, но и для тех, кто читает исследование.

Для представления данных низкой размерности (до 3-ей) существует множество вариантов визуализации. Однако далеко не все эти методы подходят для высокоразмерных данных. Это является фундаментальной проблемой представления информации на экране компьютера. Её обходят по-разному: через представление координат (одной и более) как параметры и рисование нескольких диаграмм для низкоразмерных данных, через рисование проекций на подпространства или через агрегирование выборки. Но среди всех способов визуализации можно выделить так называемый график в параллельных осях (parallel coordinates).

[1] Часто будут использоваться синонимы выборка, датасет

График в параллельных осях

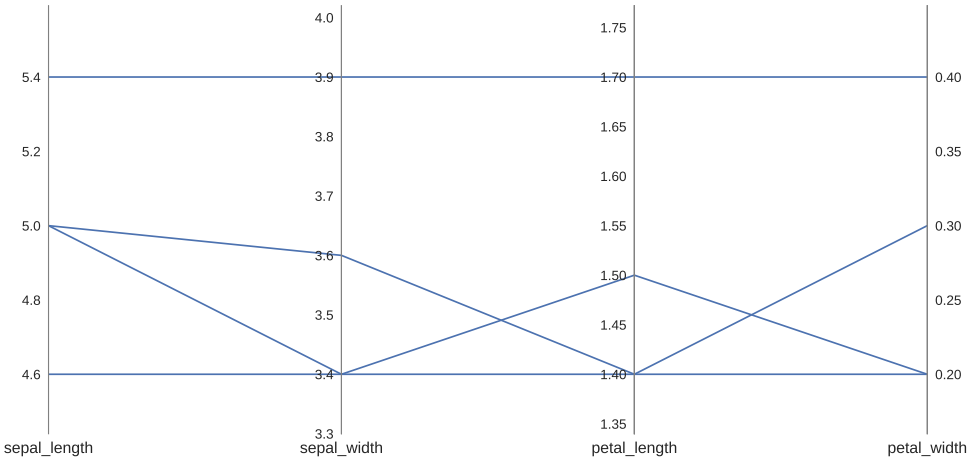

Рис. 1: Классический график в параллельных осях

График в параллельных осях — метод визуализации многомерных данных. Для отображения векторов в n-мерном пространстве рисуется n параллельных линий (осей) на одинаковом расстоянии друг от друга. Вектор представляется в виде ломаной кривой, с вершинами на параллельных осях. Точка пересечения кривой с i-ой осью соответствует i-ой координате объекта.[1]
Возникают естественные вопросы:

- В каком порядке расположить оси?
- В какую сторону направлять ось?
- Какой масштаб выбрать для каждой оси?

Модификации

1. Добавление кластеров

К сожалению, классический график в параллельных осях становится практически нечитаемым при увеличении количества объектов и осей, поэтому чаще всего его применяют в модифицированном виде — каждому объекту из выборки ставят в соответствие некоторую категориальную метку по какому-то правилу[2], а далее линия, соответствующая объекту, окрашивается в некоторый цвет однозначный метке объекта. Таким образом на графике можно проследить как ведут себя "похожие" объекты.

[2]Обычно правило выбирают так, чтобы объекты с одной меткой были "похожими" в некотором смысле – это называют кластеризацией. Иногда правило может естественным образом вытекать из самих данных, например, различные виды растений.

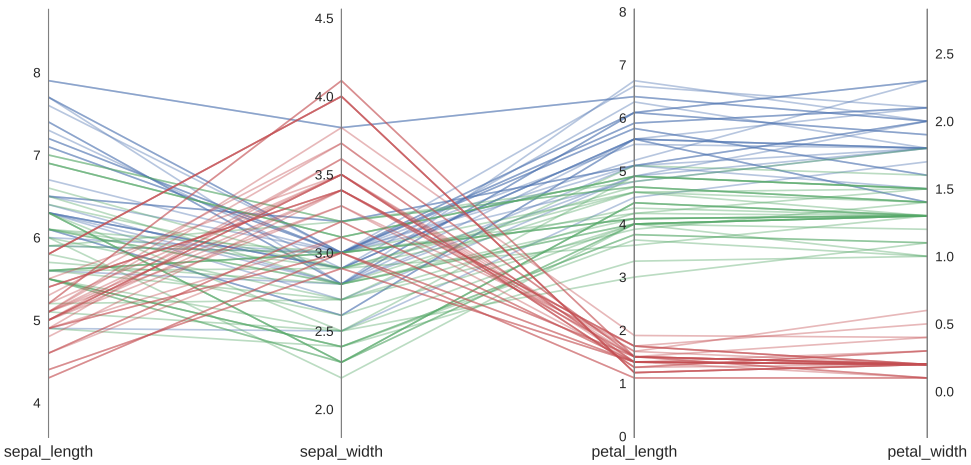

Рис. 2: График в параллельных осях с кластерами

Но многие проблемы от этого не исчезают:

- По прежнему теряется читаемость при увеличении количества объектов и признаков.
- Ломанные линии воспринимаются человеком сложнее.
- Какое правило выбрать для разметки объектов?

2. Гладкость линий

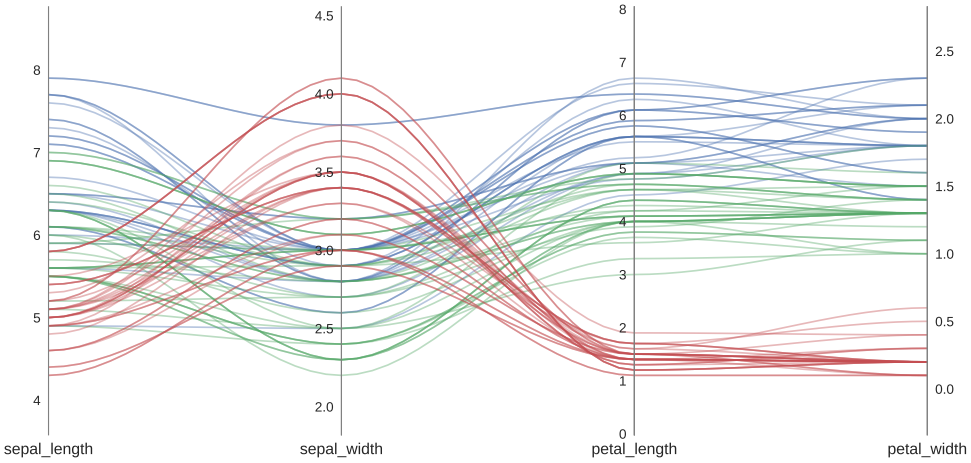

Рис. 3: График в параллельных осях с гладкими линиями

Нет никакой необходимости рисовать именно ломанные линии, можно рисовать гладкие кривые, которые ”входят” под некоторым углом к оси (чаще всего перпендикулярно). Человеку легче воспринимать гладкие линии, поэтому читаемость графика существенно возрастает.[2]

3. Связывание линий

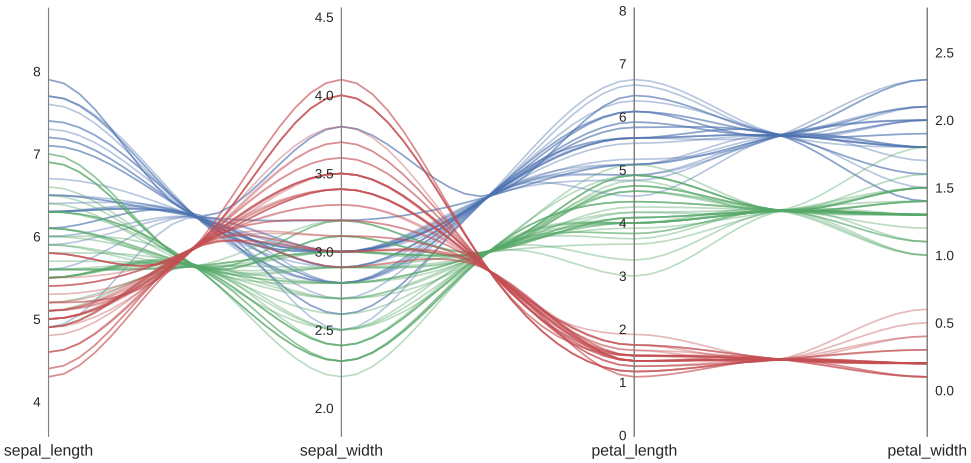

Рис. 4: График в параллельных осях со ”жгутами”

Линии с одинаковыми метками могут связываться в ”жгут” между парой осей, а далее распадаться к соответствующим точкам на оси. Степень связанности регулируема.[3]

Эту идею можно обобщить – пусть линии связываются не потому что принадлежат одному классу, а потому что имеют близкие значения координат i и i+1 при построении. Потенциально это избавляет от необходимости кластеризовать объекты перед построением.

4. Иерархические графики в параллельных осях

Иерархические графики в параллельных осях представляют собой метод визуализации не объектов, а иерархических кластерных структур — дендрограмм. Вместо визуализации конкретных объектов будем визуализировать сообщества похожих объектов. Чтобы визуализировать сообщества (кластеры) нужно выбрать некоторые статистики, например среднее или стандартное отклонение. Среднее нарисуем обычной линией, а стандартное отклонение отобразим полупрозрачным градиентом (Рис.5). Так график становится более читабельным, а детализация регулируется с помощью включения новых кластеров из дендрограммы.[4]

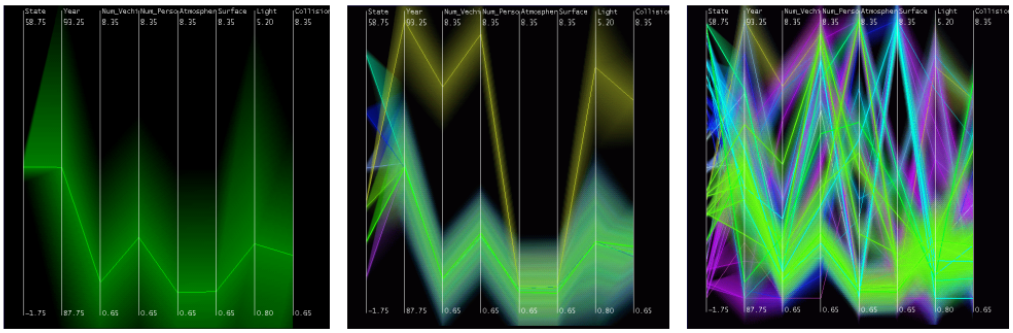

Рис. 5: Иерархические графики начиная с корня и заканчивая большим количеством кластеров

Библиотека визуализации

Обзор текущих прикладных средств

Несмотря на то, что написано большое количество работ о графиках в параллельных осях, существует лишь несколько программ, общедоступных для работы с ними. Например: ELKI, GGobi, Mondrian, Orange и ROOT. Отдельно выделяется D3.Parcoords.js – мощная библиотека на языке JavaScript, посвященная только графикам в параллельных осях. В python в библиотеке pandas есть лишь его базовая версия. В других же популярных python библиотеках нет даже этого.

Удивительно, что такое мощное средство визуализации обходят стороной разработчики библиотек. Возникает желание создать собственный продукт со всеми возможными подходами к рисованию графика в параллельных осях.

Цели и задачи библиотеки

В первую очередь необходимо заметить, что библиотека пишется на базе matplotlib. Это очень мощная низкоуровневая библиотека, умеющая рисовать всевозможные статические диаграммы. Статичность диаграммы можно считать как недостатком, так и достоинством[3]. С одной стороны интерактивность в случае графиков в параллельных осях существенно ускоряет построение эстетичного графика, но с другой это может излишне перегружать и усложнять взаимодействие пользователя с библиотекой, а также существенно уменьшить спектр возможностей. Библиотека на базе matplotilb позволит пользователю не только тончайшим образом настраивать вид графика, но и быстро получить красивый и информативный график "из коробки". [4]

[3]В сфере визуализации монополия на использование интерактивных графиков отдана JavaScript и его библиотекам. Статичные графики обычно рисуют с помощью matplotilb или библиотек, созданных на его базе.

[4]Примером может служить библиотека seaborn, написанная на базе matplotlib, но использующая высокоуровневые функции, позволяющие избегать утомляющей настройки.

Возможности библиотеки:

- Построение классических графиков в параллельных осях
  - Возможность рисовать гладкие линии. Должен быть непрерывный параметр, задаюший вид кривой.
  - Возможность "связывания" линий кластеров. Должен быть непрерывный параметр задающий степень связывания.
  - Возможность "связывания" линий на основе близости. Также должен быть непрерывный параметр задающий степень связывания.
- Построение иерархических графиков
  - Отрисовка полупрозрачного градиента.
  - Работа с иерархическими кластерами(дендрограммами).
  - Изображение распределения с помощью градиента (boxplot, histogram).
- Дополнительно
  - выделение подмножества линий в диапазоне значений одной из осей.
  - нахождение оптимального расположения осей.
  - создание иерархических кластеров на основе входящей выборки.

Технические особенности:

- Простой высокоуровневый интерфейс. Как и в библиотеке seaborn методы могут принимать pandas.DataFrame, обычные numpy массивы или списки – для всего единый интерфейс.
- Эстетичные графики "из коробки".

Итоги(после первого семестра)

По итогам семестра удалось реализовать большую часть возможностей библиотеки, касающихся классического графика в параллельных осях

- Возможность рисовать гладкие линии. **Пока что не добавлен параметр задающий вид кривой**.
- Возможность "связывания" линий кластеров. Добавлен непрерывный параметр задающий степень связывания.
- **Возможность связывания линий на основе близости не реализована**

Интерфейс для пользователя практически полностью повторяет реализацию seaborn. Пока что в качестве параметра доступен только pandas.DataFrame, но уже сейчас можно получать красивые графики "из коробки".[5]

[5]Большинство графиков в отчете нарисованы с помощью данной библиотеки.

Список литературы

[1] Inselberg A., Dimsdale B. Parallel coordinates: a tool for visualizing multi-dimensional geometry //Proceedings of the First IEEE Conference on Visualization: Visualization90. – IEEE, 1990. – С. 361-378.

[2] Heinrich J., Weiskopf D. State of the Art of Parallel Coordinates //Eurographics (STARs). – 2013. – С. 95-116.

[3] Heinrich J. et al. Evaluation of a bundling technique for parallel coordinates //arXiv preprint arXiv:1109.6073. – 2011.

[4] Fua Y. H., Ward M. O., Rundensteiner E. A. Hierarchical parallel coordinates for exploration of large datasets. – IEEE, 1999. – С. 43-508.