

# Разработка программного обеспечения, ориентированного на пользователя, для проведения кластер-анализа по критерию наименьших квадратов

---

Выполнил:

Еремейкин Пётр Александрович  
студент группы мНод16-ТМСС  
eremeykin@gmail.com

Руководитель:

Миркин Борис Григорьевич  
д.т.н., профессор

# Постановка задачи кластеризации

Пусть имеется  $N$  объектов и у каждого объекта определены значения  $V$  признаков. Множество всех объектов  $Y$  можно представить в виде таблицы:

$$Y = \begin{pmatrix} y_1 \\ \dots \\ y_N \end{pmatrix} = \begin{pmatrix} y_{11} & \dots & y_{1V} \\ \dots & \dots & \dots \\ y_{N1} & \dots & y_{NV} \end{pmatrix}$$

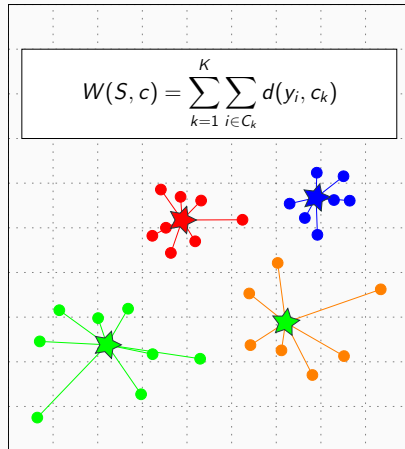
Требуется получить разбиение  $S = \{C_1, \dots, C_K\}$ , состоящее из  $K$  кластеров, которые не пересекаются и покрывают всё множество объектов  $Y$ . Чёткой формулировки относительно того, что должно быть включено в кластеры не существует. Общая идея состоит в том чтобы сходные объекты были включены в один кластер, а несходные не принадлежали одному кластеру.

# Традиционное решение ( $k$ -means)

Самый популярный алгоритм кластеризации —  $k$ -means. Этот метод основан на поочерёдной минимизации квадратичного критерия по двум группам переменных: центрам кластеров и принадлежности объектов кластерам.

Недостатки метода:

- требует задания числа кластеров
- сильно зависит от инициализации
- плохо работает для зашумлённых данных

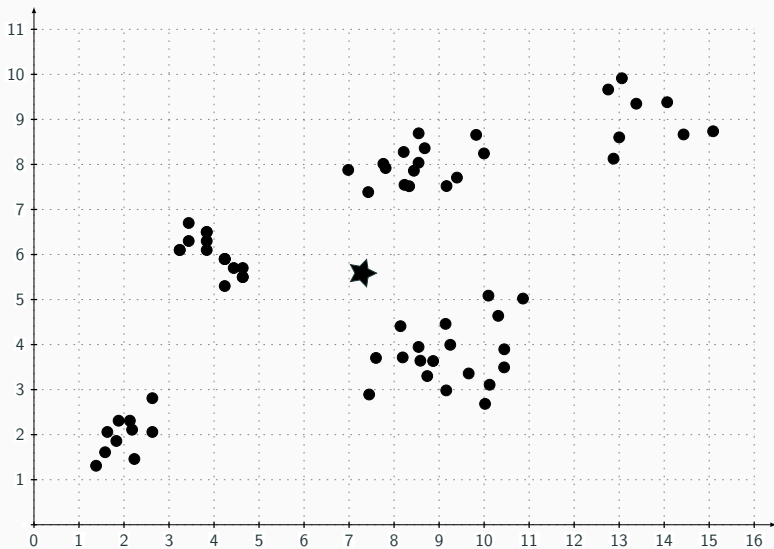


За последнее время появилось большое число новых и эффективных алгоритмов кластеризации, многие из них еще не реализованы в популярных библиотеках, таких как `scipy` для языка Python или `Clustering Toolbox` для MATLAB. Предлагается разработать программу, в которую входили бы следующие алгоритмы:

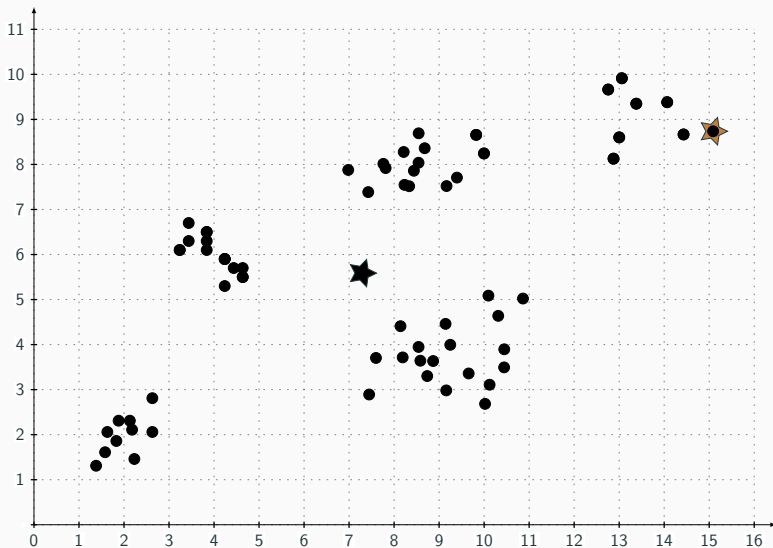
- *ik*-means
- A-Ward
- $A\text{-Ward}_{p\beta}$
- dePDDP
- BiKM-R

## Алгоритм *ik-means*

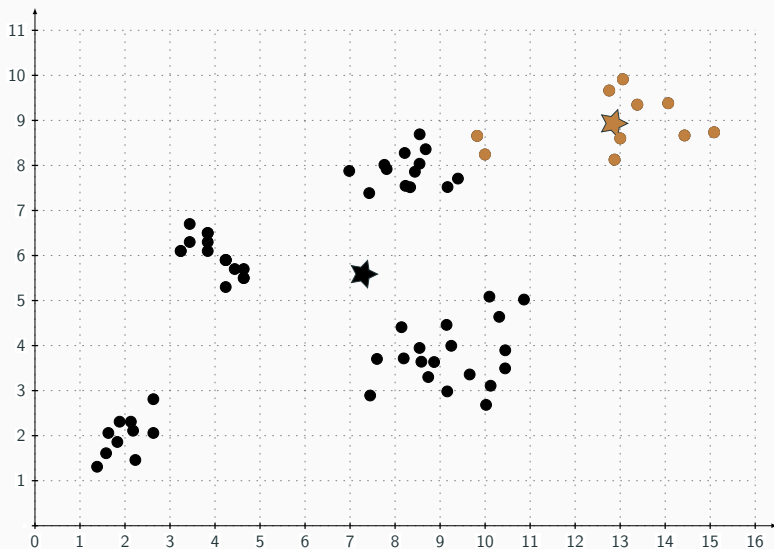
---



# Алгоритм *ik*-means

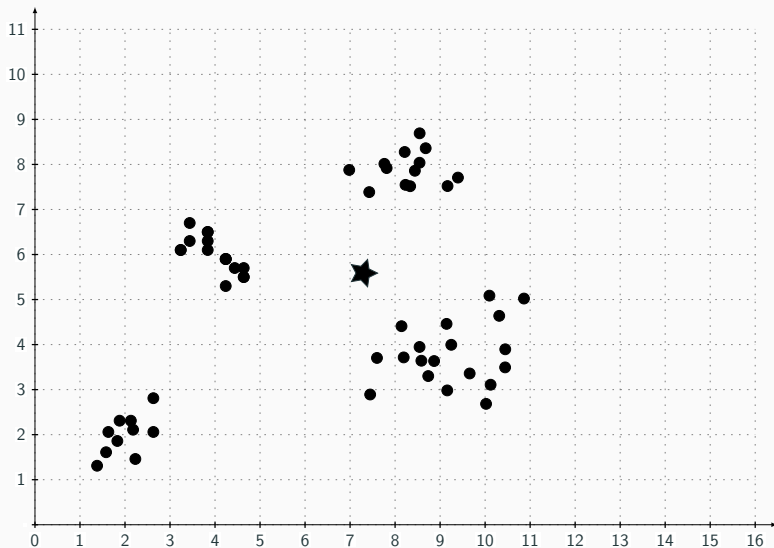


# Алгоритм *ik*-means

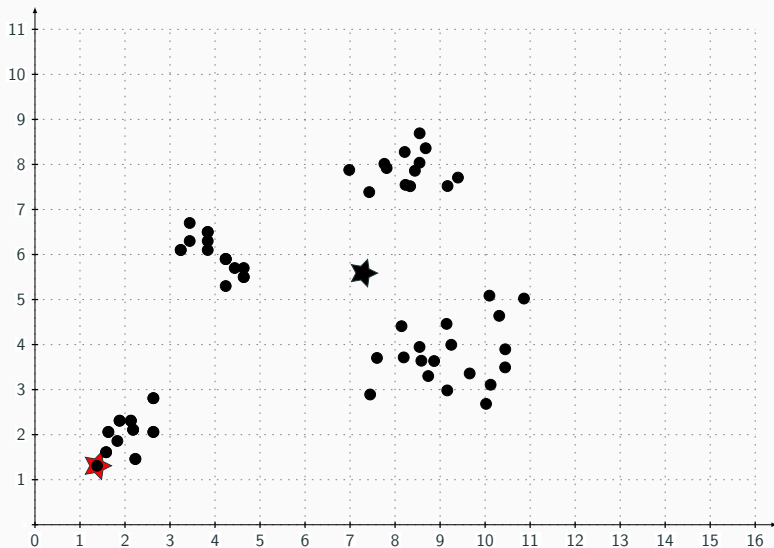




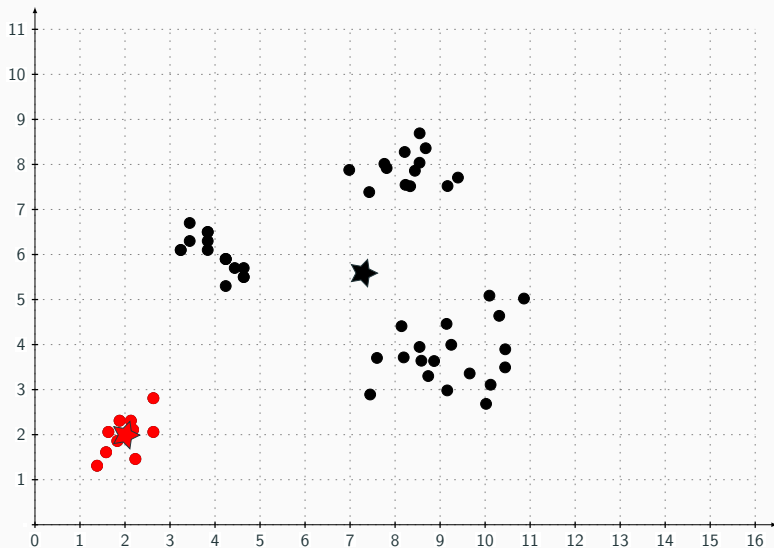
# Алгоритм *ik*-means



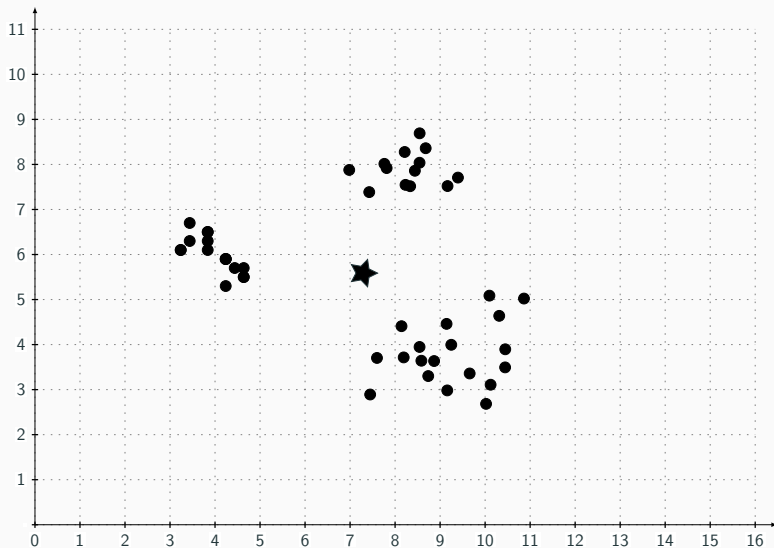
# Алгоритм *ik*-means



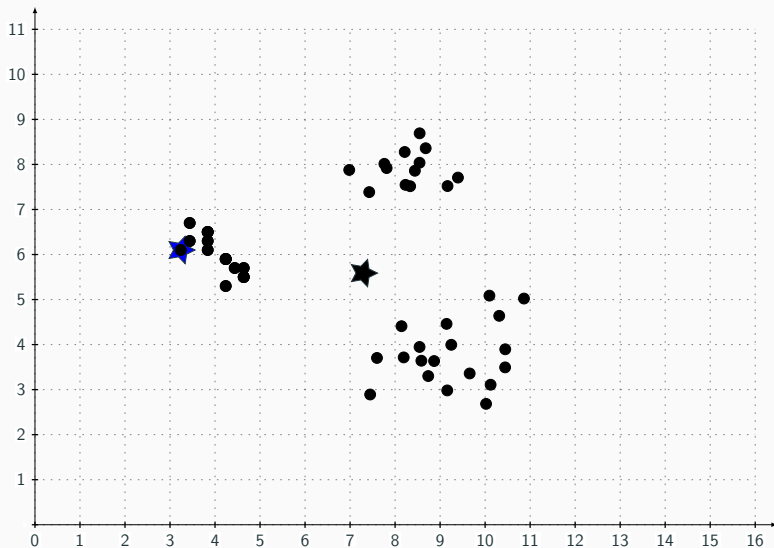
# Алгоритм *ik*-means



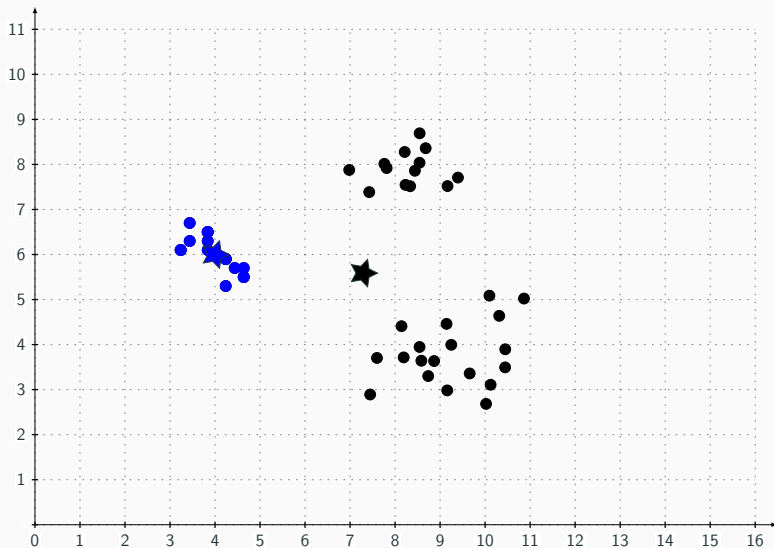
# Алгоритм *ik*-means



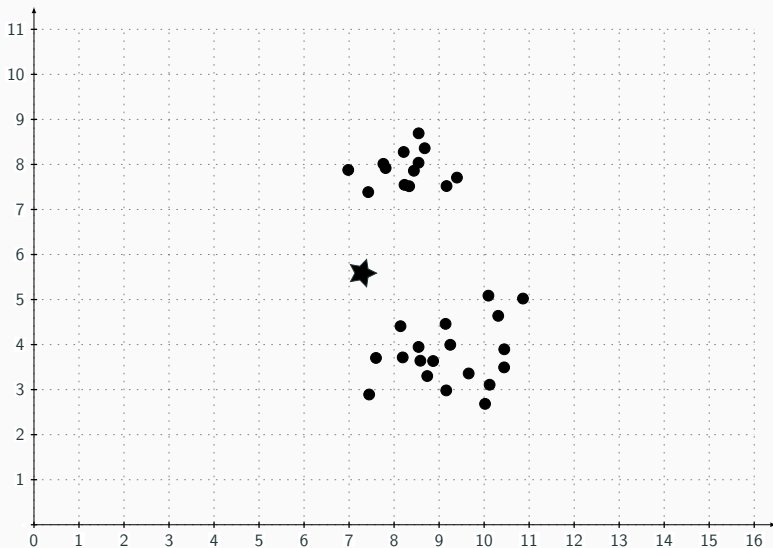
# Алгоритм *ik*-means



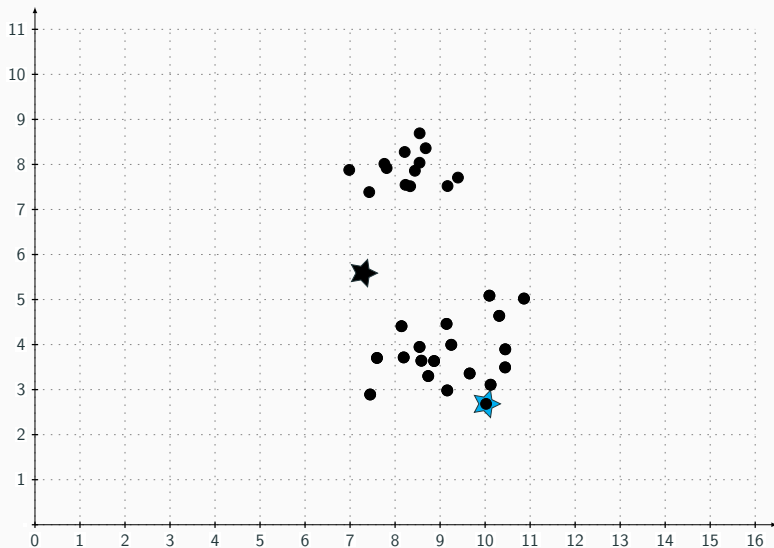
# Алгоритм *ik*-means



# Алгоритм *ik*-means

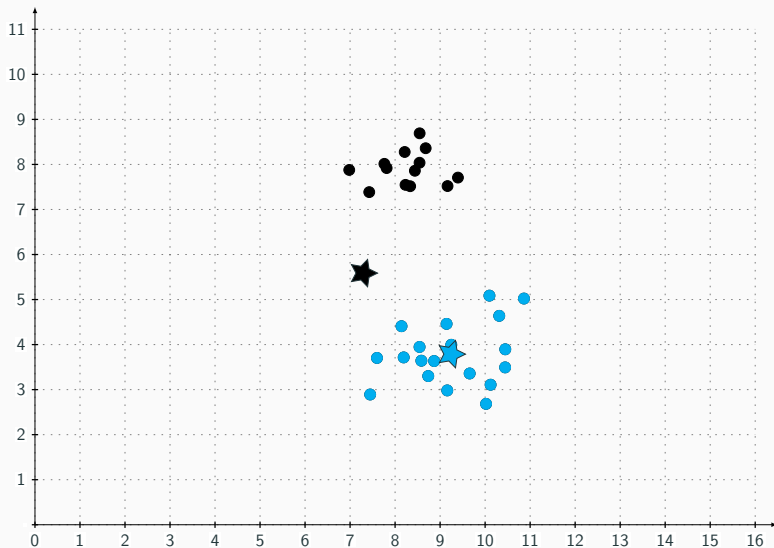


## Алгоритм *ik*-means

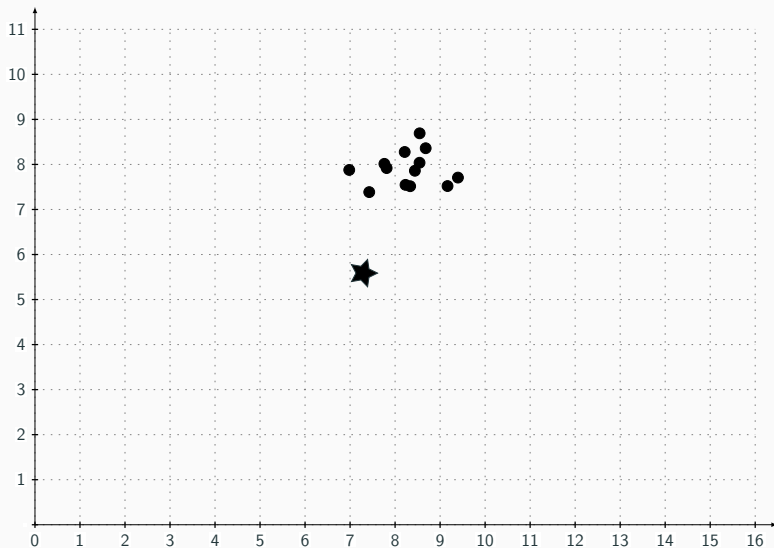




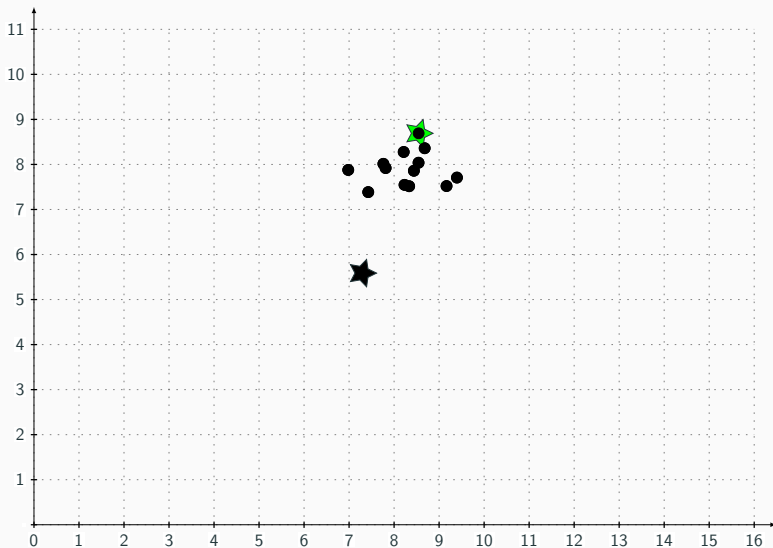
# Алгоритм *ik*-means



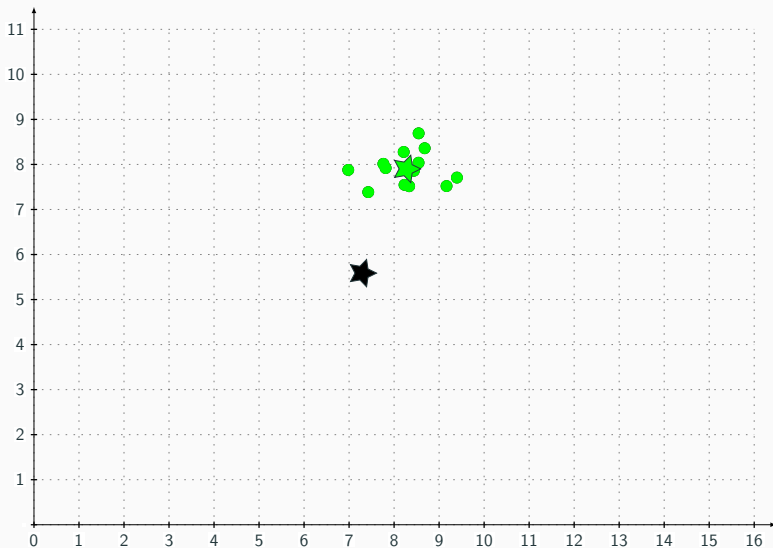
## Алгоритм *ik*-means



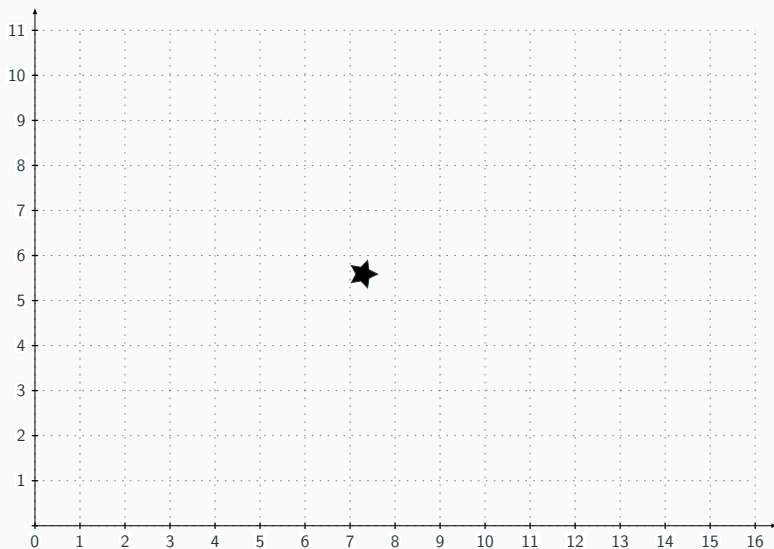
## Алгоритм *ik*-means



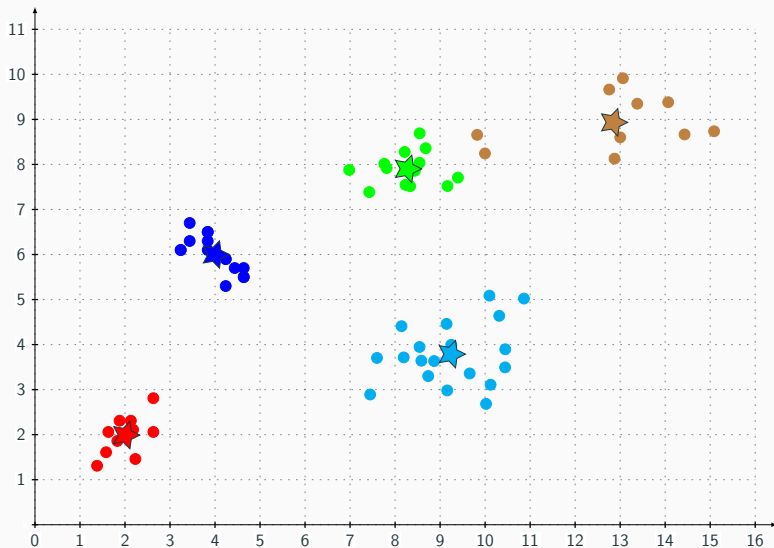
# Алгоритм *ik*-means



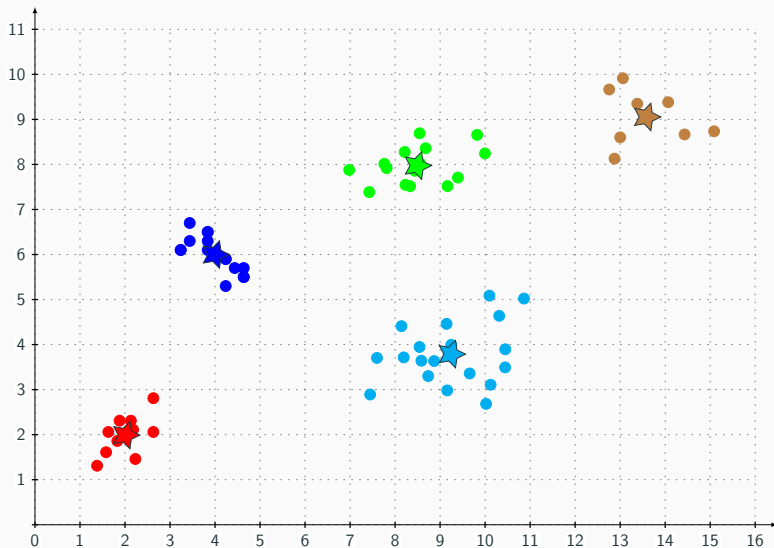
# Алгоритм *ik*-means



# Алгоритм *ik*-means



## Алгоритм *ik*-means

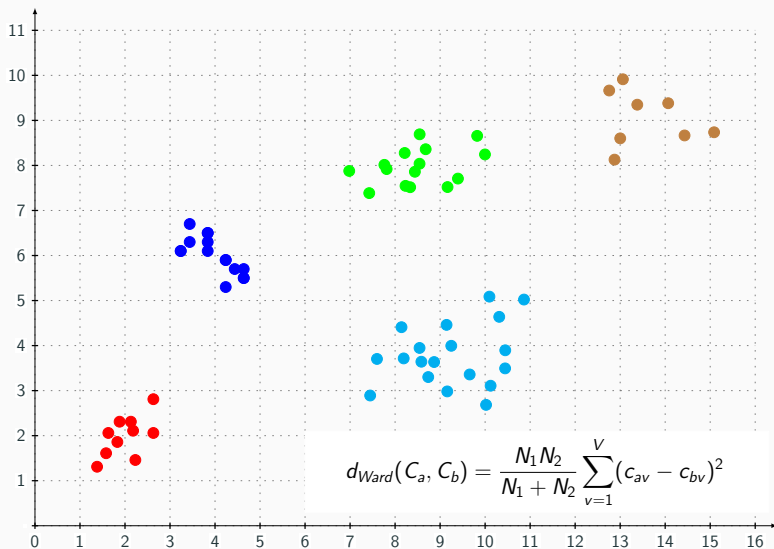


## Алгоритм A-Ward

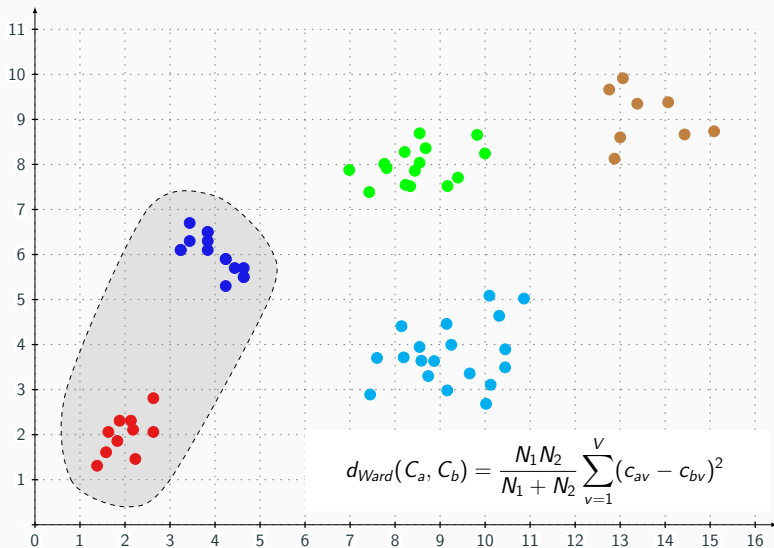
---



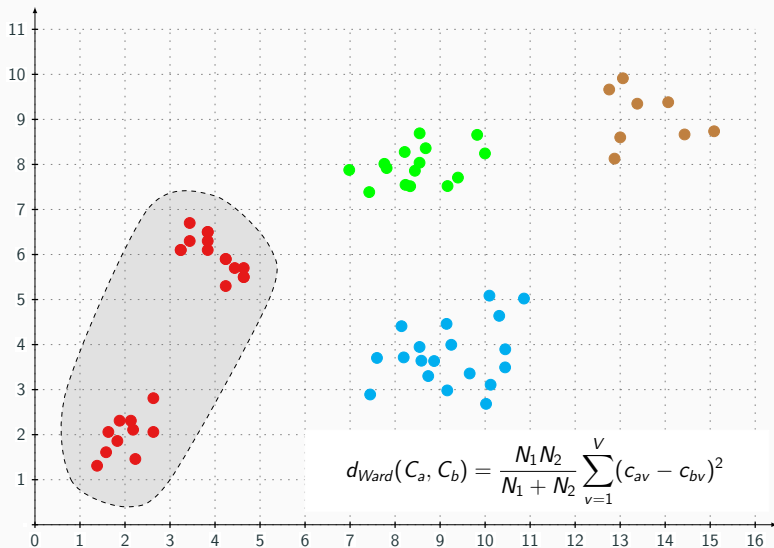
# Алгоритм A-Ward



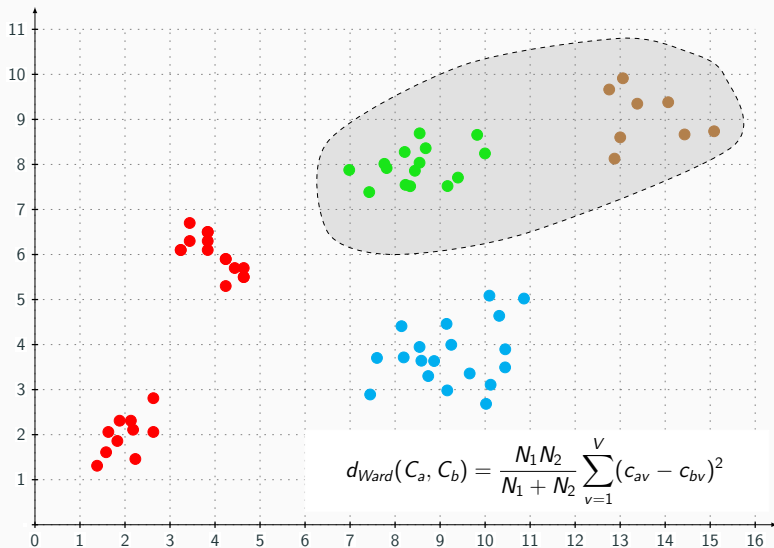
# Алгоритм A-Ward



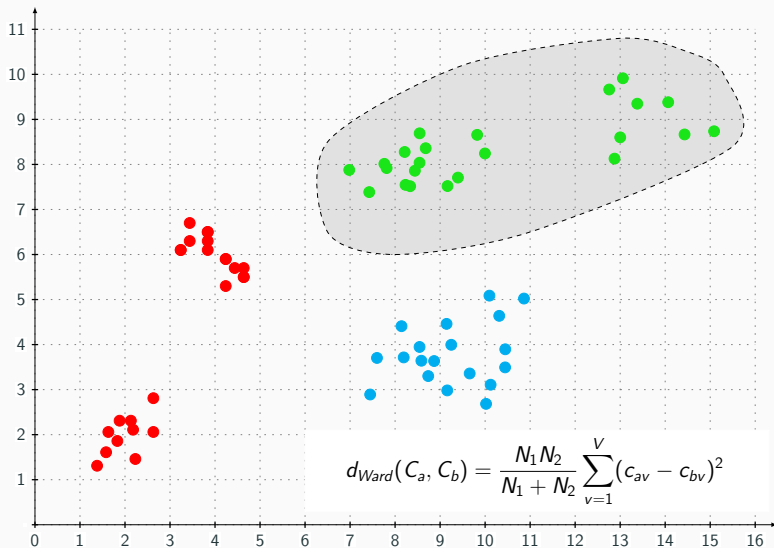
# Алгоритм A-Ward



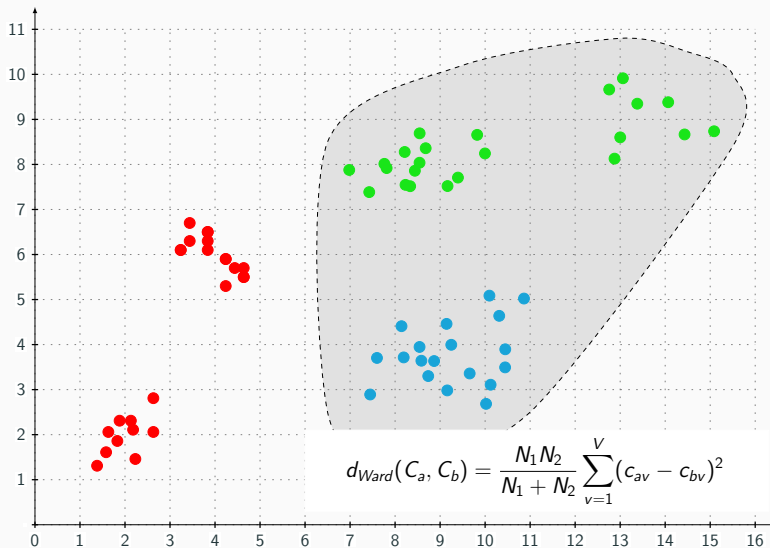
# Алгоритм A-Ward



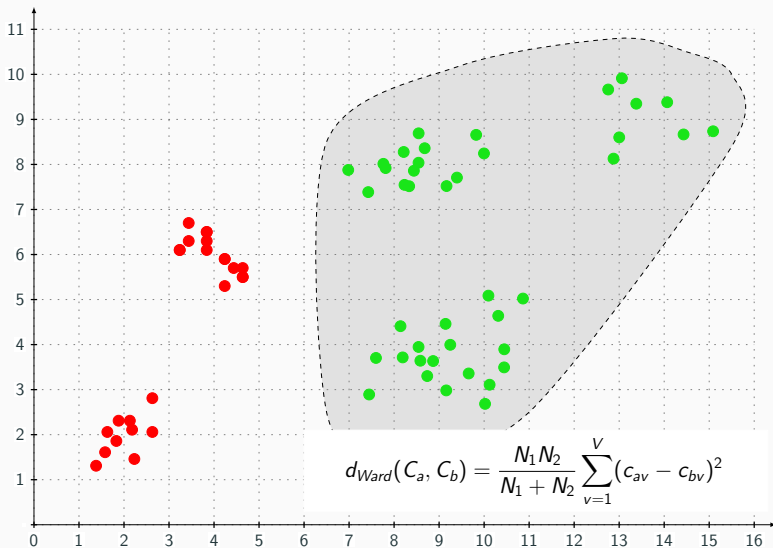
# Алгоритм A-Ward



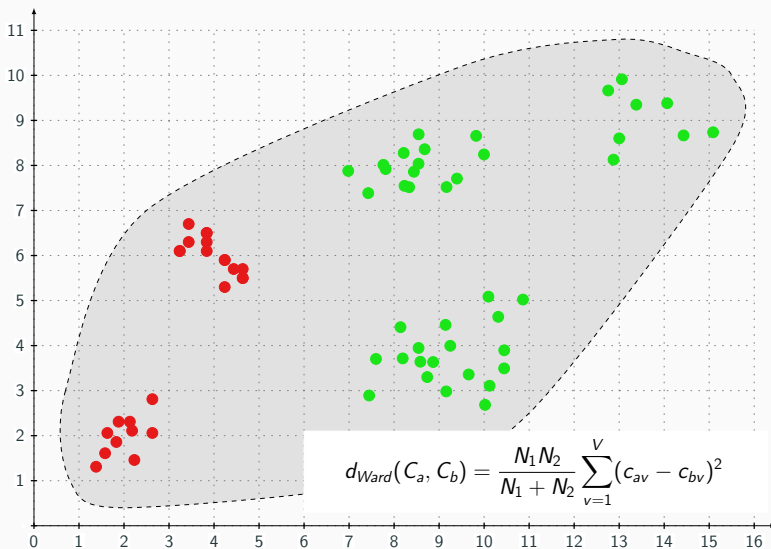
# Алгоритм A-Ward



# Алгоритм A-Ward

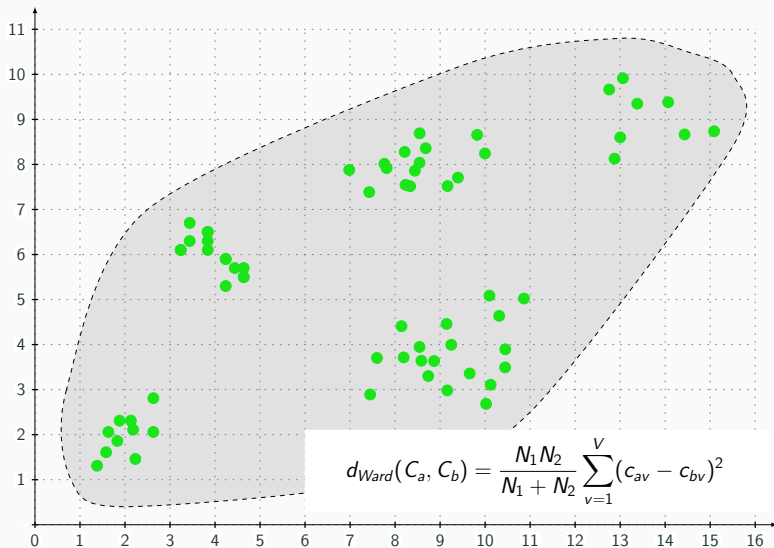


# Алгоритм A-Ward

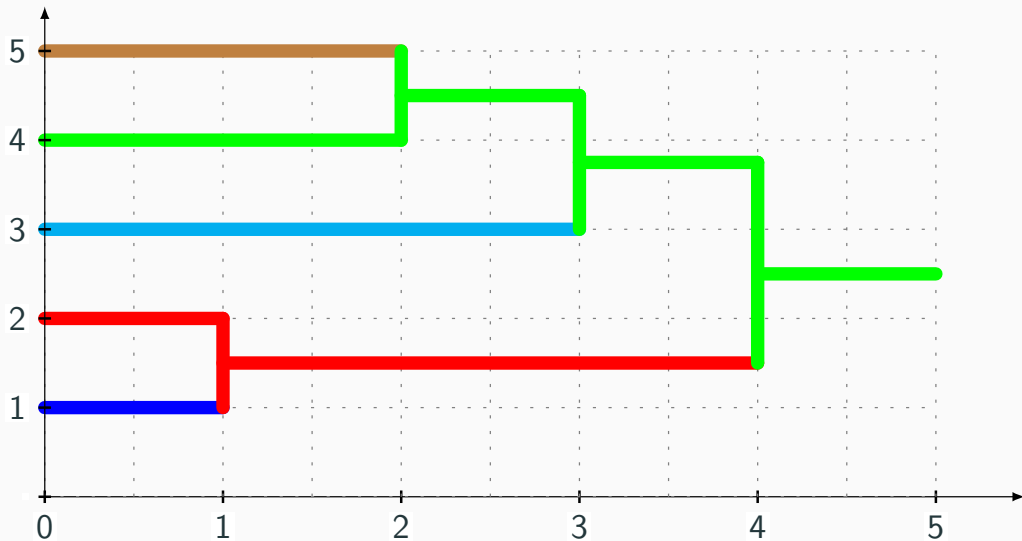




# Алгоритм A-Ward



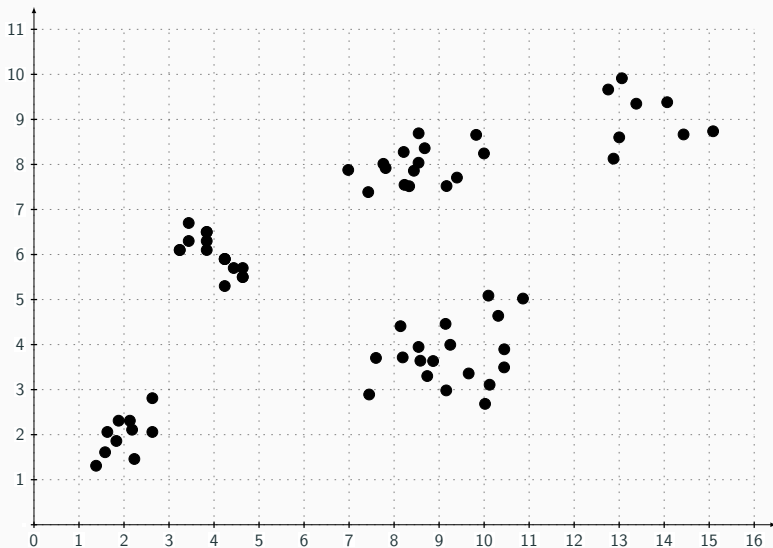
## Алгоритм A-Ward



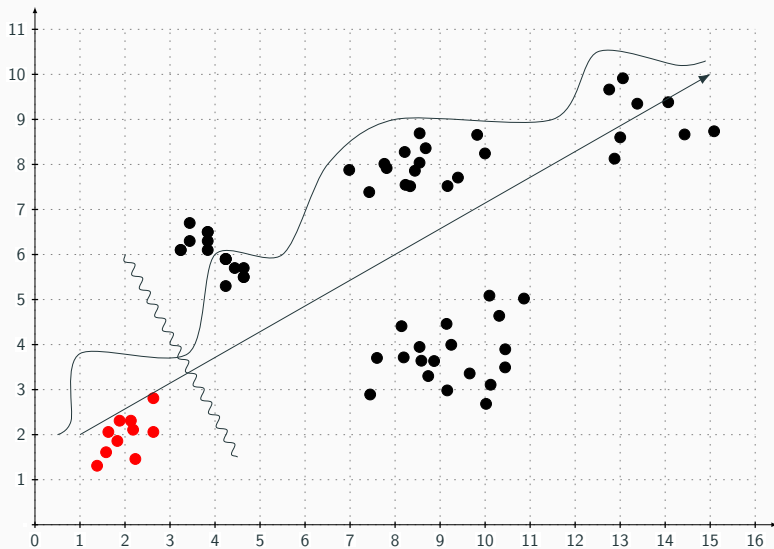
## Алгоритм dePDDP

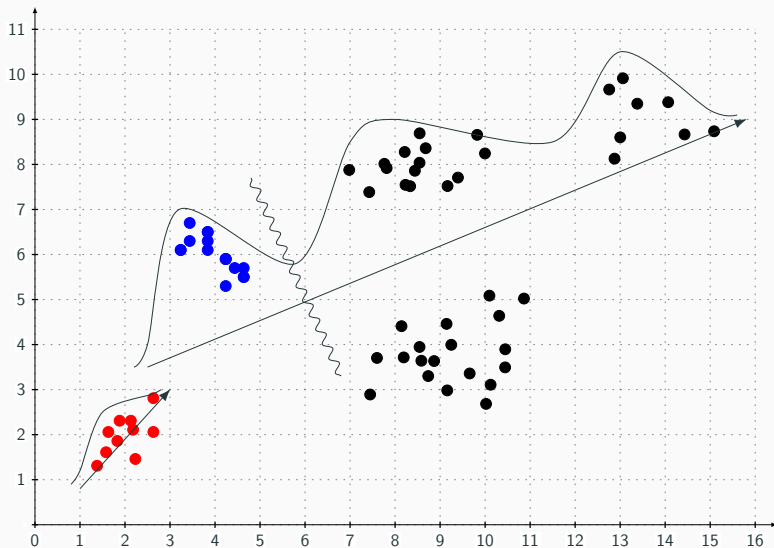
---

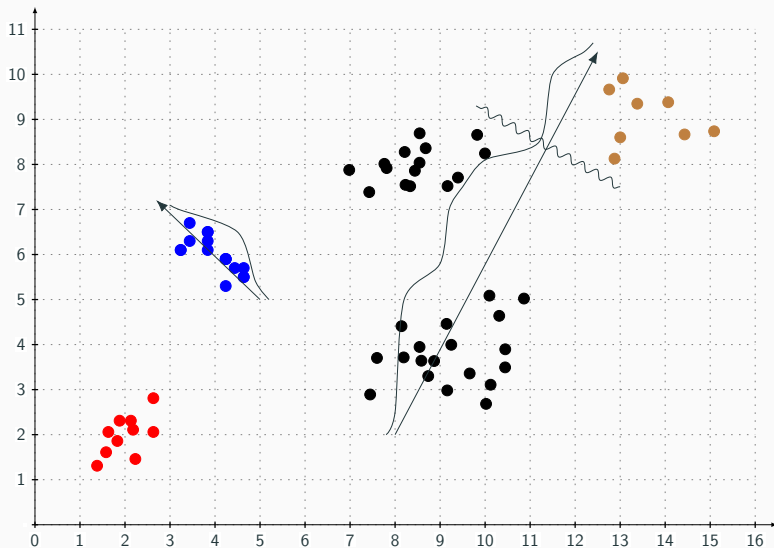
# Алгоритм dePDDP

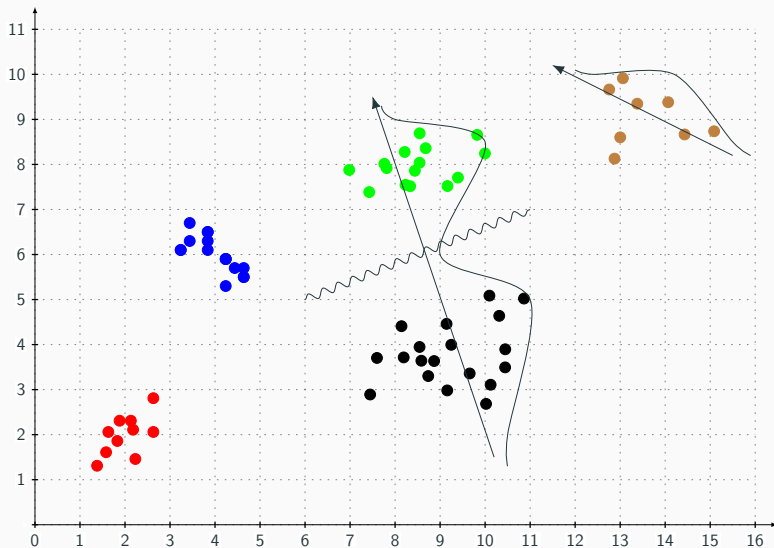


# Алгоритм dePDDP

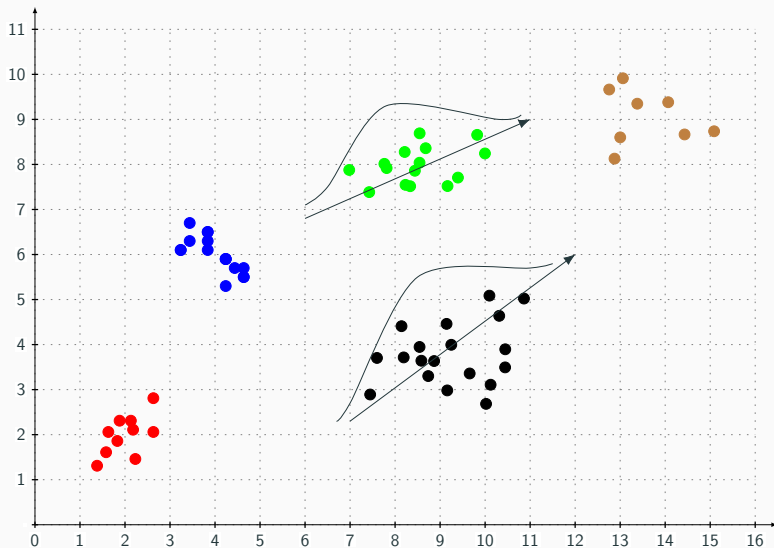




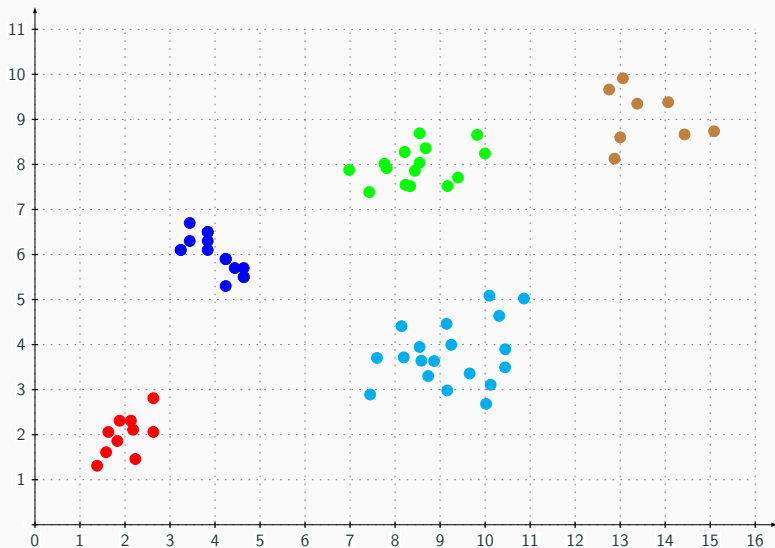








# Алгоритм dePDDP







## Выводы

Создана работоспособная программа

Созданную программу надо опробовать в реальных условиях

Спасибо за внимание!