Introduction to Artificial Intelligence

English practicals 10: Machine learning

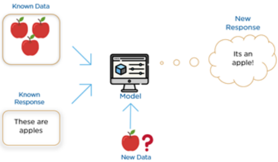
Marika Ivanová

Department of Theoretical Computer Science and Mathematical Logic (KTIML)
Faculty of Mathematics and Physics

May 13th 2021

Machine learning

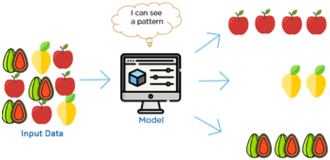
Supervised
(task driven)



source: towardsdatascience.com

- Ad popularity
- Spam classification
- Face recognition

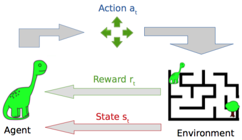
Unsupervised
(data driven)



source: towardsdatascience.com

- Recommendation systems
- Buying habits
- Grouping user logs

Reinforcement
(learn from mistakes)



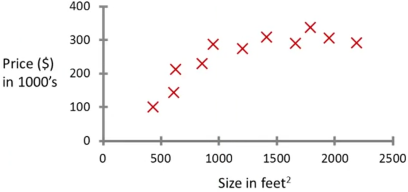
source: towardsdatascience.com

- Video games
- Industrial simulations
- Robot navigation

- We already touched upon reinforcement learning when last time we talked about the robot moving in the maze

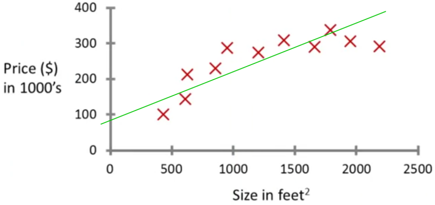
Supervised learning: regression vs classification

Regression (linear, polynomial)



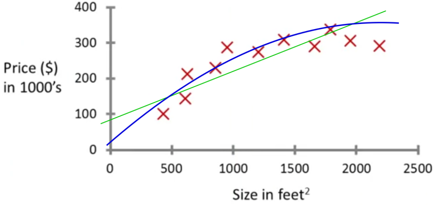
Supervised learning: regression vs classification

Regression (linear, polynomial)



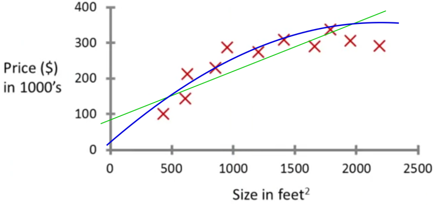
Supervised learning: regression vs classification

Regression (linear, polynomial)

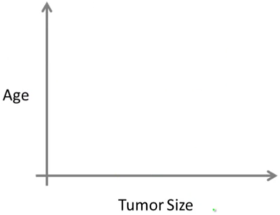


Supervised learning: regression vs classification

Regression (linear, polynomial)

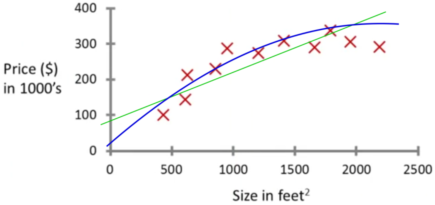


Classification

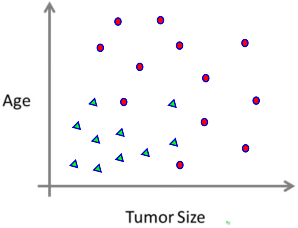


Supervised learning: regression vs classification

Regression (linear, polynomial)

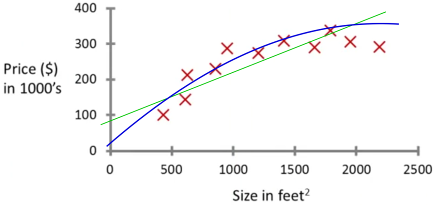


Classification

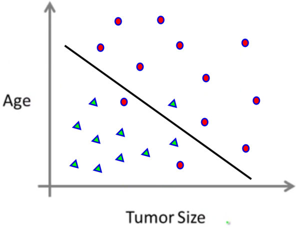


Supervised learning: regression vs classification

Regression (linear, polynomial)



Classification



Exercise: regression vs classification

Decide what type of problems are the following tasks:

1. You have an inventory of identical items. You want to predict how many of these items will sell over the next 3 months.
2. You'd like to examine individual customer accounts, and for each account decide if it has been hacked.

a) Treat both as classification problems
b) Treat problem (1) as classification, problem (2) as regression
c) Treat problem (1) as regression, problem (2) as classification
d) Treat both as regerssion problems

Exercise: regression vs classification

Decide what type of problems are the following tasks:

1. You have an inventory of identical items. You want to predict how many of these items will sell over the next 3 months.
2. You'd like to examine individual customer accounts, and for each account decide if it has been hacked.

a) Treat both as classification problems

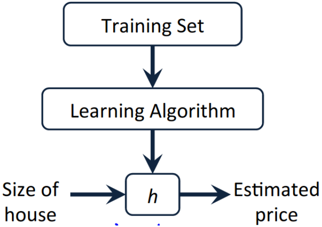
b) Treat problem (1) as classification, problem (2) as regression

c) Treat problem (1) as regression, problem (2) as classification

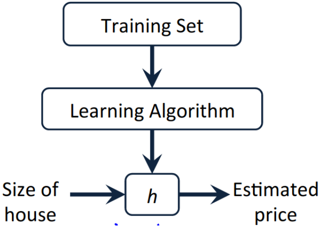
d) Treat both as regerssion problems

C is correct

Linear regression

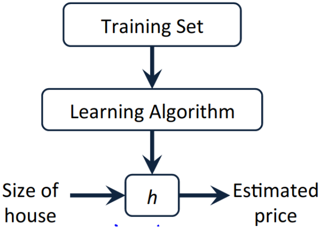


Linear regression

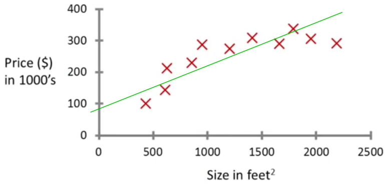


Size in feet^2 (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
...	...

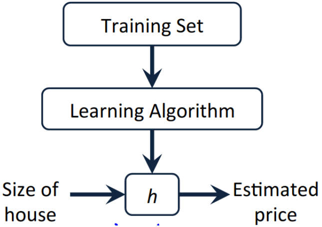
Linear regression



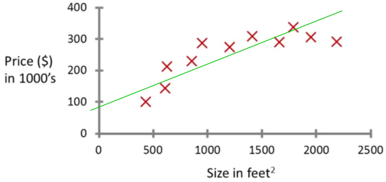
Size in feet² (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
...	...



Linear regression



Size in feet² (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…



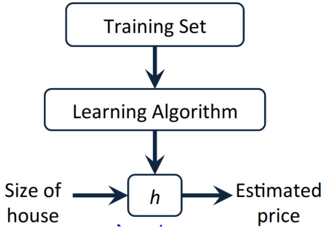
Hypothesis:

$$h_w(x) = w_0 + w_1(x)$$

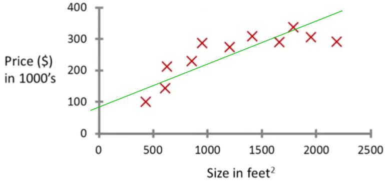
$w_0, w_1 \ldots$ parameters

how to choose w?

Linear regression



Size in feet^2 (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…



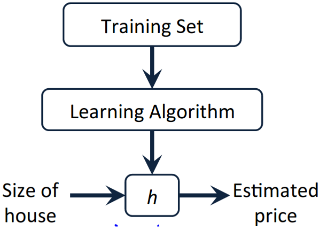
Hypothesis:

$$h_w(x) = w_0 + w_1(x)$$

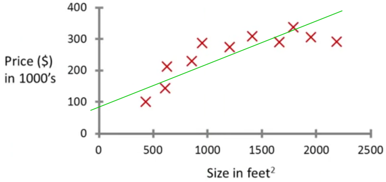
$w_0, w_1 \ldots$ parameters
how to choose w?

Choose w_0, w_1 so that $h_w(x)$ is close to y for training examples (x, y)

Linear regression



Size in feet² (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…



Hypothesis:

$$h_w(x) = w_0 + w_1(x)$$

$w_0, w_1 \dots$ parameters
how to choose w?

Choose w_0, w_1 so that $h_w(x)$ is close to y for training examples (x, y)

$$\min_{w_0, w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)} \right)^2$$

Linear regression

- $\min_{w_0, w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$

Linear regression

- $\min_{w_0, w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0, w_1} J(w_0, w_1)$

Linear regression

- $\min_{w_0, w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0, w_1} J(w_0, w_1)$

Linear regression

- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

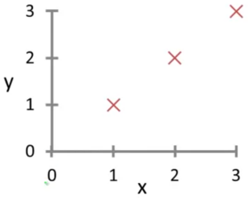
Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$

Linear regression

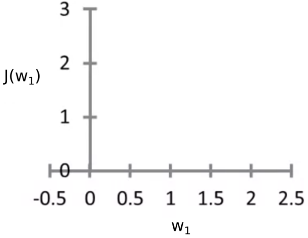
- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$

$h_w(x)$



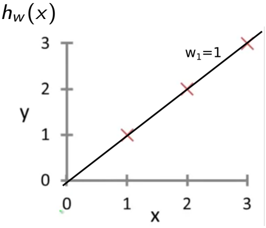
$J(w_1)$

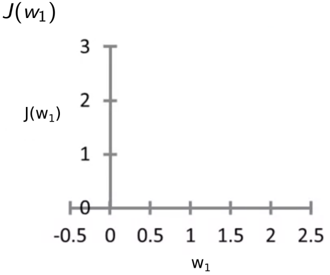


Linear regression

- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$



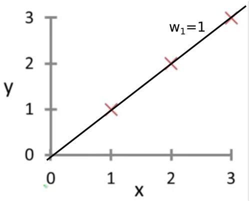


Linear regression

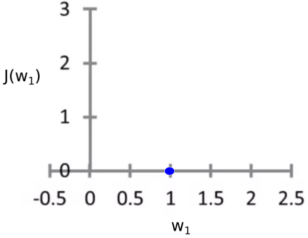
- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$

$h_w(x)$



$J(w_1)$

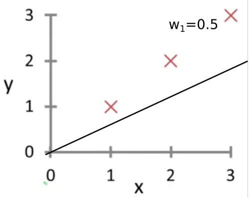


Linear regression

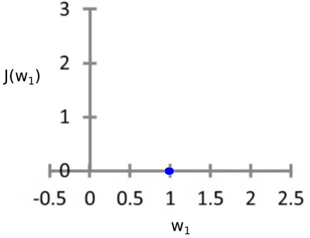
- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$

$h_w(x)$



$J(w_1)$

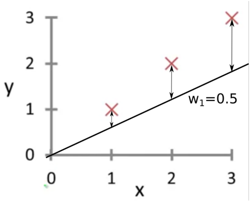


Linear regression

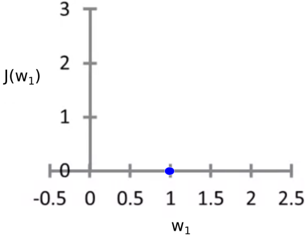
- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$

$h_w(x)$



$J(w_1)$

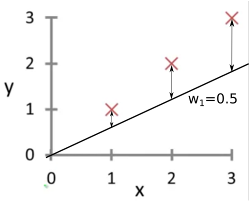


Linear regression

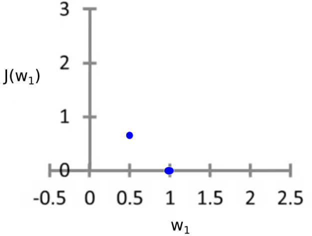
- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$

$h_w(x)$



$J(w_1)$

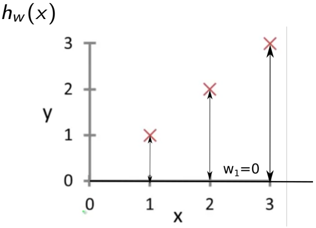


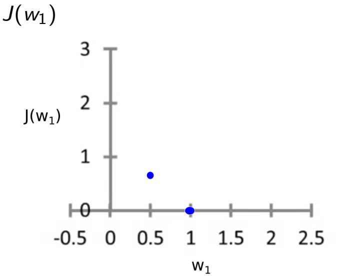
$J(0.5) = 1/6 * [(0.5-1)^2 + (1-2)^2 + (1.5-3)^2] = 3.5/6 \approx 0.58$

Linear regression

- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$





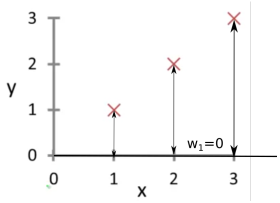
$J(0.5) = 1/6 * [(0.5-1)^2 + (1-2)^2 + (1.5-3)^2] = 3.5/6 \approx 0.58$

Linear regression

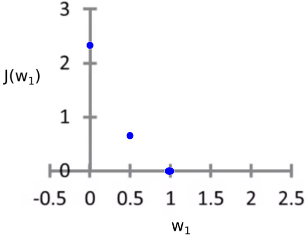
- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$

$h_w(x)$



$J(w_1)$

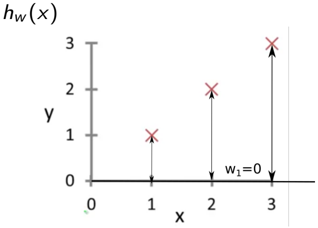


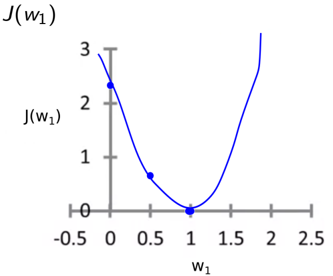
$J(0.5) = 1/6 * [(0.5-1)^2 + (1-2)^2 + (1.5-3)^2] = 3.5/6 \approx 0.58$
$J(0) = 1/6 * [(1^2 + 2^2 + 3^2)] \approx 2.3$

Linear regression

- $\min_{w_0,w_1} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$
- cost function: $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$, thus $\min_{w_0,w_1} J(w_0, w_1)$

Intuition: simplify the hypothesis: $h_w(x) = w_1 x_1$, we want $\min_{w_1} J(w_1)$

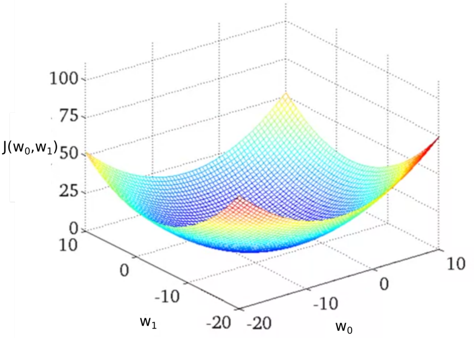




$J(0.5) = 1/6 * [(0.5-1)^2 + (1-2)^2 + (1.5-3)^2] = 3.5/6 \approx 0.58$
$J(0) = 1/6 * [(1^2 + 2^2 + 3^2)] \approx 2.3$

Linear regression

Cost function when we do not simplify the hypothesis, i. e,
$h_w(x) = w_0 + w_1 x$



Linear regression: gradient descent

Linear regression model:

- $h_w(x) = w_0 + w_1 x_1$
- $J(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2$

Gradient descent:

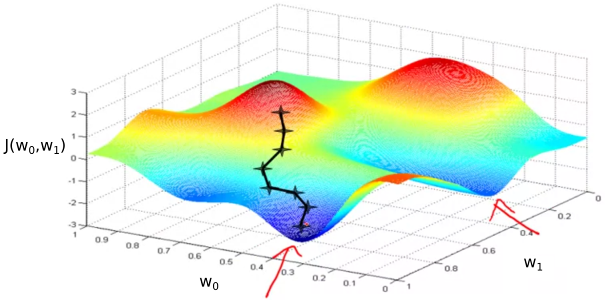
repeat until convergence {
$$w_j \leftarrow w_j - \alpha \frac{\partial}{\partial w_j} J(w_0, w_1)$$
for $j = 0$ and $j = 1$
}

$$\frac{\partial}{\partial w_j} J(w_0, w_1) = \frac{\partial}{\partial w_j} \frac{1}{2m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)^2 = \frac{\partial}{\partial w_j} \frac{1}{2m} \sum_{i=1}^{m} \left(w_0 + w_1 x^{(i)} - y^{(i)}\right)^2$$

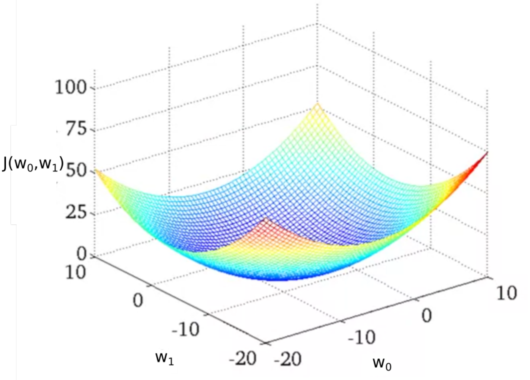
$$\frac{\partial}{\partial w_0} J(w_0, w_1) = \frac{\partial}{\partial w_0} \frac{1}{m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right)$$

$$\frac{\partial}{\partial w_1} J(w_0, w_1) = \frac{\partial}{\partial w_1} \frac{1}{m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)}\right) x^{(i)}$$

Linear regression: gradient descent



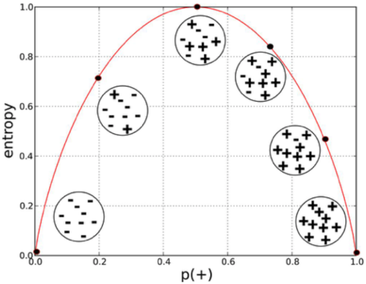
Linear regression: gradient descent



Decision trees

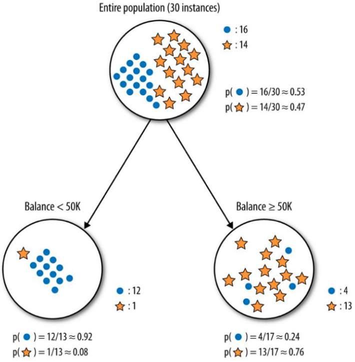
1. Select the most important attribute based on entropy $H(V)$
2. Divide samples based on this attribute
3. Repeat 1) and 2) until all samples belong to the same category

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$



Exercise: Decision tree (1/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

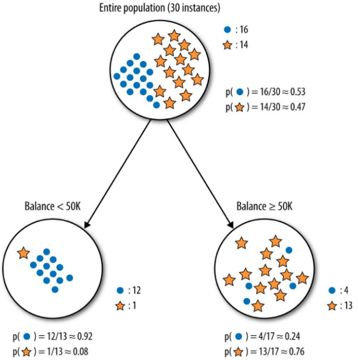


Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(parent) =$
- $H(balance < 50) =$
- $H(balance > 50) =$

Exercise: Decision tree (1/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

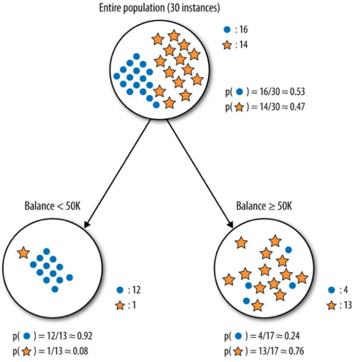


Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(parent) = -\frac{16}{30}\log\frac{16}{30} - \frac{14}{30}\log\frac{14}{30} \approx 0.99$
- $H(balance < 50) =$
- $H(balance > 50) =$

Exercise: Decision tree (1/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

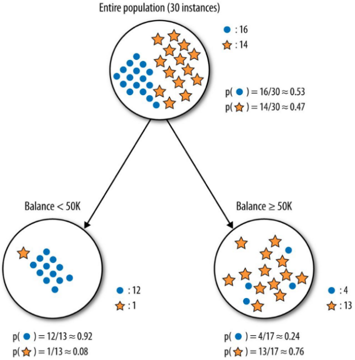


Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(parent) = -\frac{16}{30}\log\frac{16}{30} - \frac{14}{30}\log\frac{14}{30} \approx 0.99$
- $H(balance < 50) = -\frac{12}{13}\log\frac{12}{13} - \frac{1}{13}\log\frac{1}{13} \approx 0.39$
- $H(balance > 50) =$

Exercise: Decision tree (1/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

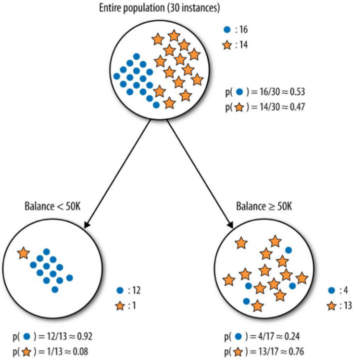


Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(parent) = -\frac{16}{30}\log\frac{16}{30} - \frac{14}{30}\log\frac{14}{30} \approx 0.99$
- $H(balance < 50) = -\frac{12}{13}\log\frac{12}{13} - \frac{1}{13}\log\frac{1}{13} \approx 0.39$
- $H(balance > 50) = -\frac{4}{17}\log\frac{4}{17} - \frac{13}{17}\log\frac{13}{17} \approx 0.79$

Exercise: Decision tree (1/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$



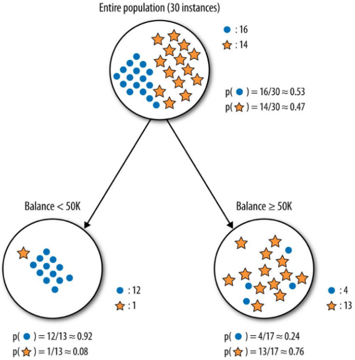
Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(parent) = -\frac{16}{30}\log\frac{16}{30} - \frac{14}{30}\log\frac{14}{30} \approx 0.99$
- $H(balance < 50) = -\frac{12}{13}\log\frac{12}{13} - \frac{1}{13}\log\frac{1}{13} \approx 0.39$
- $H(balance > 50) = -\frac{4}{17}\log\frac{4}{17} - \frac{13}{17}\log\frac{13}{17} \approx 0.79$

Weighted average of entropy for each node:

Exercise: Decision tree (1/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$



Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

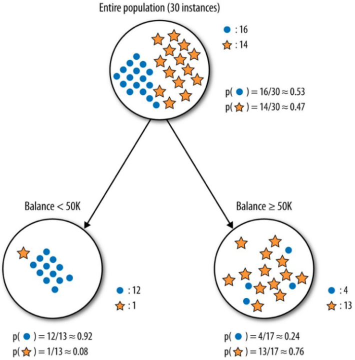
- $H(parent) = -\frac{16}{30}\log\frac{16}{30} - \frac{14}{30}\log\frac{14}{30} \approx 0.99$
- $H(balance < 50) = -\frac{12}{13}\log\frac{12}{13} - \frac{1}{13}\log\frac{1}{13} \approx 0.39$
- $H(balance > 50) = -\frac{4}{17}\log\frac{4}{17} - \frac{13}{17}\log\frac{13}{17} \approx 0.79$

Weighted average of entropy for each node:

- $H(balance) = 13/30*0.39 + 17/30*0.79 \approx 0.62$

Exercise: Decision tree (1/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$



Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(parent) = -\frac{16}{30}\log\frac{16}{30} - \frac{14}{30}\log\frac{14}{30} \approx 0.99$
- $H(balance < 50) = -\frac{12}{13}\log\frac{12}{13} - \frac{1}{13}\log\frac{1}{13} \approx 0.39$
- $H(balance > 50) = -\frac{4}{17}\log\frac{4}{17} - \frac{13}{17}\log\frac{13}{17} \approx 0.79$

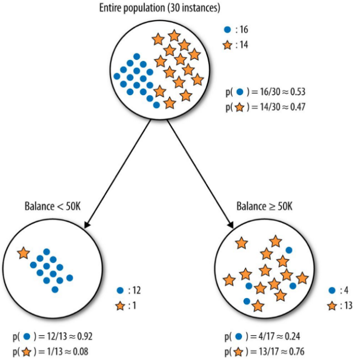
Weighted average of entropy for each node:

- $H(balance) = 13/30 * 0.39 + 17/30 * 0.79 \approx 0.62$

Information gain:

Exercise: Decision tree (1/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$



Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(parent) = -\frac{16}{30}\log\frac{16}{30} - \frac{14}{30}\log\frac{14}{30} \approx 0.99$
- $H(balance < 50) = -\frac{12}{13}\log\frac{12}{13} - \frac{1}{13}\log\frac{1}{13} \approx 0.39$
- $H(balance > 50) = -\frac{4}{17}\log\frac{4}{17} - \frac{13}{17}\log\frac{13}{17} \approx 0.79$

Weighted average of entropy for each node:

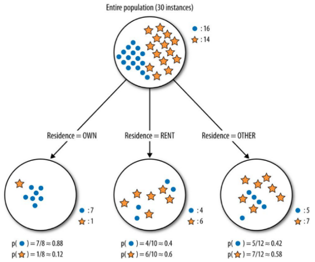
- $H(balance) = 13/30 * 0.39 + 17/30 * 0.79 \approx 0.62$

Information gain:

- $Gain(parent, balance) = H(parent) - H(balance) = 0.99 - 0.62 = 0.37$

Exercise: Decision tree (2/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

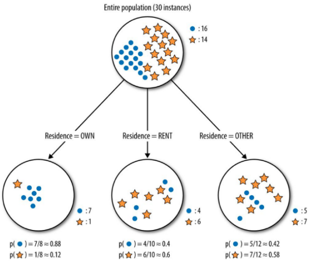


Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

Exercise: Decision tree (2/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

- $H(residence = OWN) = -\frac{7}{8}\log\frac{7}{8} - \frac{1}{8}\log\frac{1}{8} \approx 0.54$

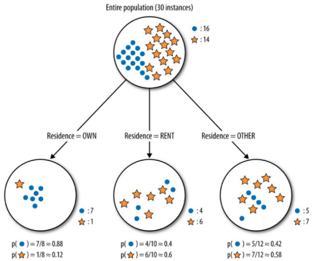


Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

Exercise: Decision tree (2/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

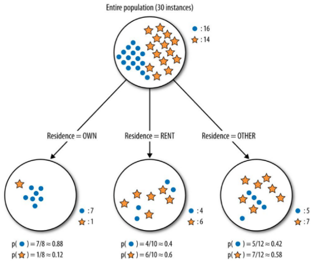
- $H(residence = OWN) = -\frac{7}{8}\log\frac{7}{8} - \frac{1}{8}\log\frac{1}{8} \approx 0.54$
- $H(residence = RENT) = -\frac{4}{10}\log\frac{4}{10} - \frac{6}{10}\log\frac{6}{10} \approx 0.97$



Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

Exercise: Decision tree (2/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

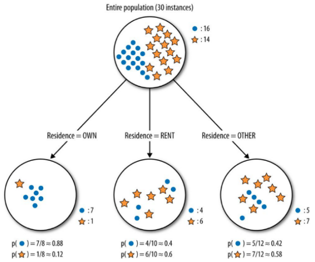


Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(residence = OWN) = -\frac{7}{8}\log\frac{7}{8} - \frac{1}{8}\log\frac{1}{8} \approx 0.54$
- $H(residence = RENT) = -\frac{4}{10}\log\frac{4}{10} - \frac{6}{10}\log\frac{6}{10} \approx 0.97$
- $H(residence = OTHER) = -\frac{5}{12}\log\frac{5}{12} - \frac{7}{12}\log\frac{7}{12} \approx 0.98$

Exercise: Decision tree (2/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$



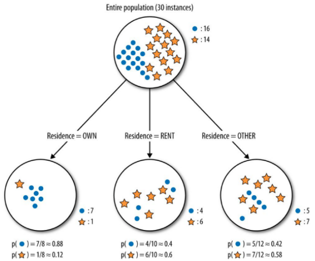
Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(residence = OWN) = -\frac{7}{8}\log\frac{7}{8} - \frac{1}{8}\log\frac{1}{8} \approx 0.54$
- $H(residence = RENT) = -\frac{4}{10}\log\frac{4}{10} - \frac{6}{10}\log\frac{6}{10} \approx 0.97$
- $H(residence = OTHER) = -\frac{5}{12}\log\frac{5}{12} - \frac{7}{12}\log\frac{7}{12} \approx 0.98$

Weighted average of entropy for each node:

Exercise: Decision tree (2/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$



Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

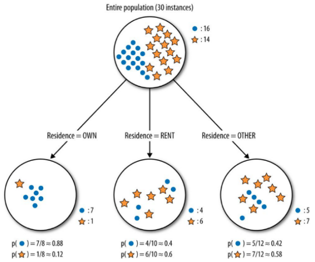
- $H(residence = OWN) = -\frac{7}{8}\log\frac{7}{8} - \frac{1}{8}\log\frac{1}{8} \approx 0.54$
- $H(residence = RENT) = -\frac{4}{10}\log\frac{4}{10} - \frac{6}{10}\log\frac{6}{10} \approx 0.97$
- $H(residence = OTHER) = -\frac{5}{12}\log\frac{5}{12} - \frac{7}{12}\log\frac{7}{12} \approx 0.98$

Weighted average of entropy for each node:

- $H(residence) = 8/30 * 0.54 + 10/30 * 0.97 + 12/30 * 0.98 \approx 0.86$

Exercise: Decision tree (2/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$



Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(residence = OWN) = -\frac{7}{8}\log\frac{7}{8} - \frac{1}{8}\log\frac{1}{8} \approx 0.54$
- $H(residence = RENT) = -\frac{4}{10}\log\frac{4}{10} - \frac{6}{10}\log\frac{6}{10} \approx 0.97$
- $H(residence = OTHER) = -\frac{5}{12}\log\frac{5}{12} - \frac{7}{12}\log\frac{7}{12} \approx 0.98$

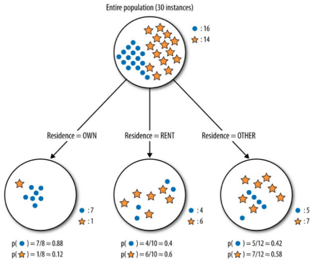
Weighted average of entropy for each node:

- $H(residence) = 8/30 * 0.54 + 10/30 * 0.97 + 12/30 * 0.98 \approx 0.86$

Information gain:

Exercise: Decision tree (2/2)

$$H(V) = -\sum_{i=1}^{c} p_i \log_2 p_i$$



Provost, Foster; Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking

- $H(residence = OWN) = -\frac{7}{8}\log\frac{7}{8} - \frac{1}{8}\log\frac{1}{8} \approx 0.54$
- $H(residence = RENT) = -\frac{4}{10}\log\frac{4}{10} - \frac{6}{10}\log\frac{6}{10} \approx 0.97$
- $H(residence = OTHER) = -\frac{5}{12}\log\frac{5}{12} - \frac{7}{12}\log\frac{7}{12} \approx 0.98$

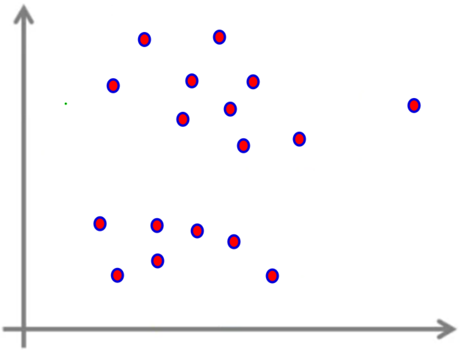
Weighted average of entropy for each node:

- $H(residence) = 8/30 * 0.54 + 10/30 * 0.97 + 12/30 * 0.98 \approx 0.86$

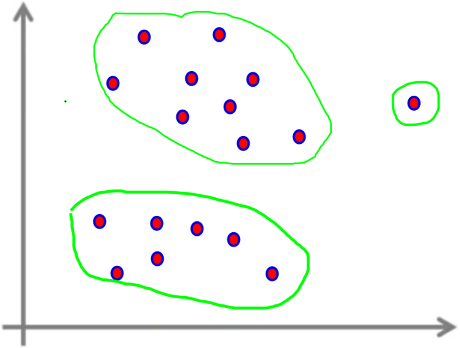
Information gain:

- $Gain(parent, residence) = H(parent) - H(residence) = 0.99 - 0.86 = 0.13$

Unsupervised learning: clustering

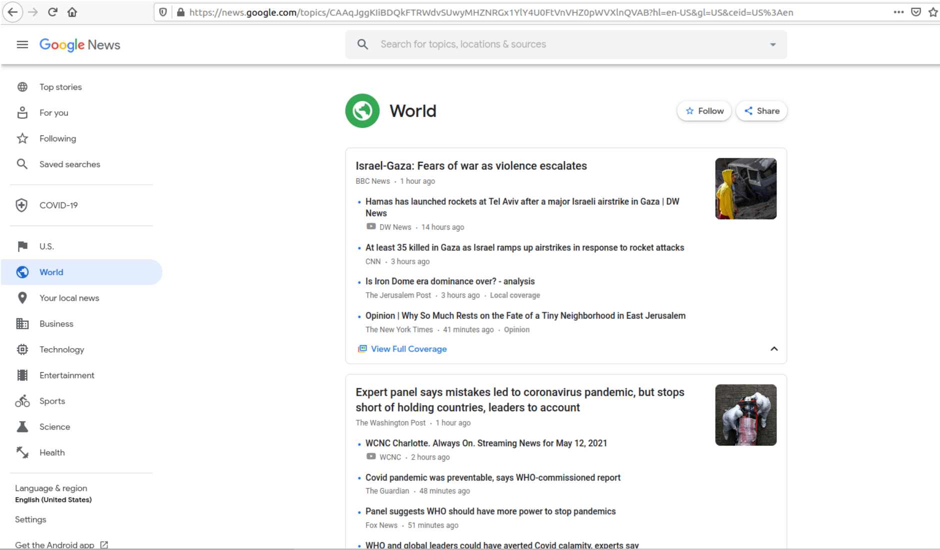


Unsupervised learning: clustering



Unsupervised learning: clustering example

Google news



k-means

Simple, widely used clustering algorithm
The number of clusters must be given

Input: k, set of points $a_1, \ldots, a_n$

Place centroids $c_1, \ldots, c_k$ at random locations

Repeat until convergence:

- for each point a_i:

  find nearest centroid: $c_j = \arg\min_j d_{ij}$
  assign the point a_i to cluster C_j

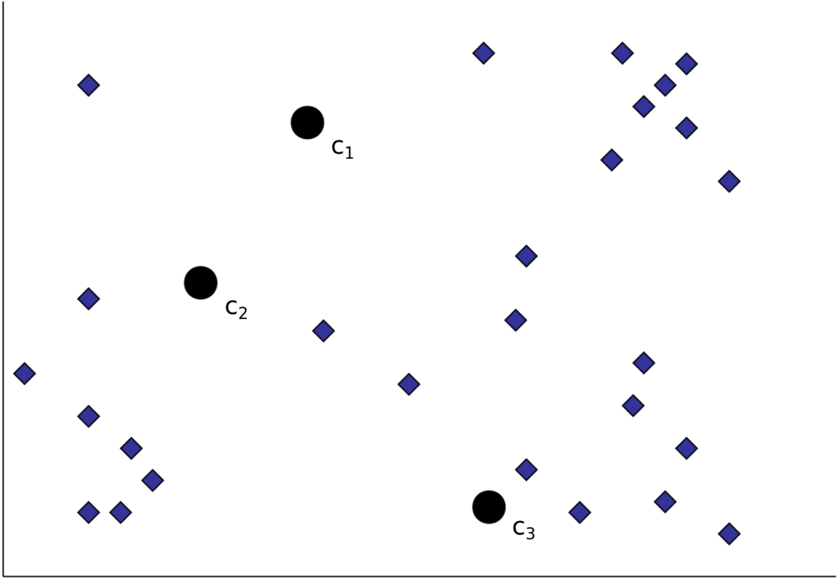
- for each cluster C_j, $j = 1, \ldots, k$:

  new centroid c_j = mean of all points a_i assigned to cluster C_j in the previous step

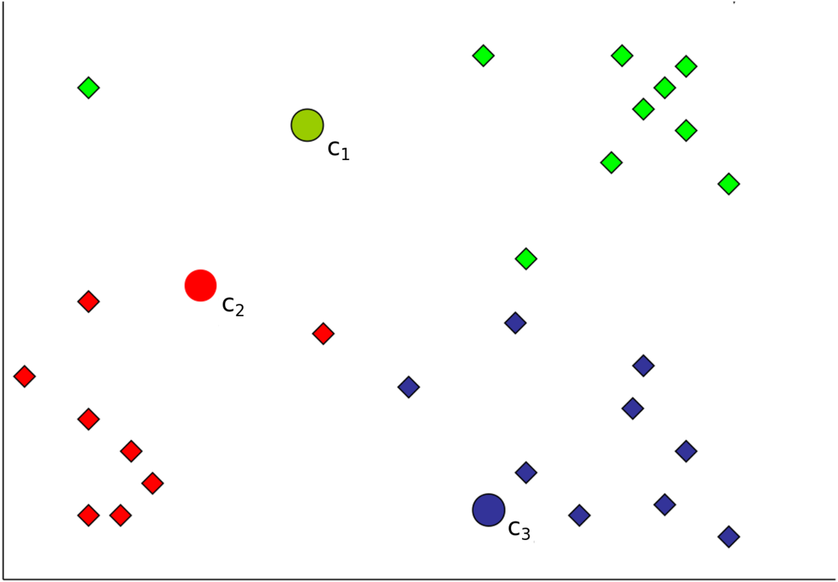
$$c_j(\ell) = \frac{1}{n} \sum_{a_i \in C_j} a_i(\ell) \text{ for } \ell = 1, \ldots, m$$

Stop when none of the cluster assignments change

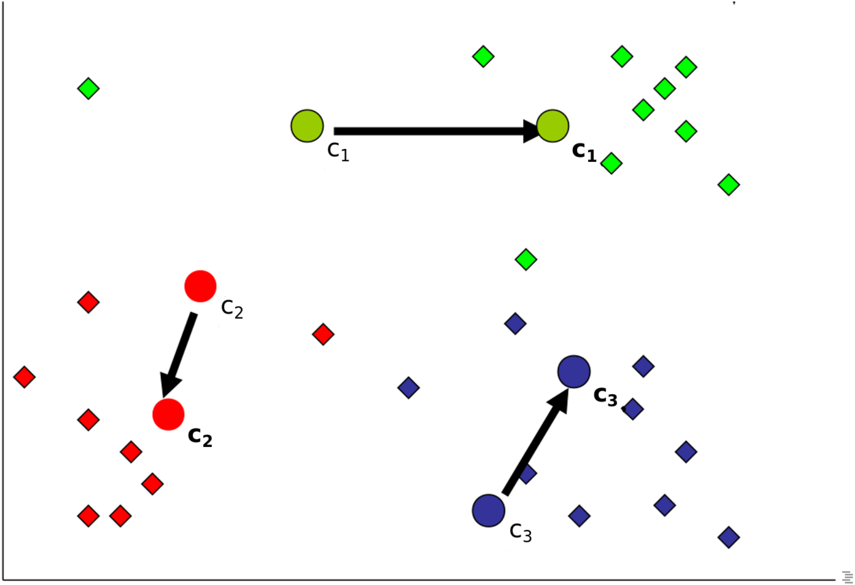
k-means



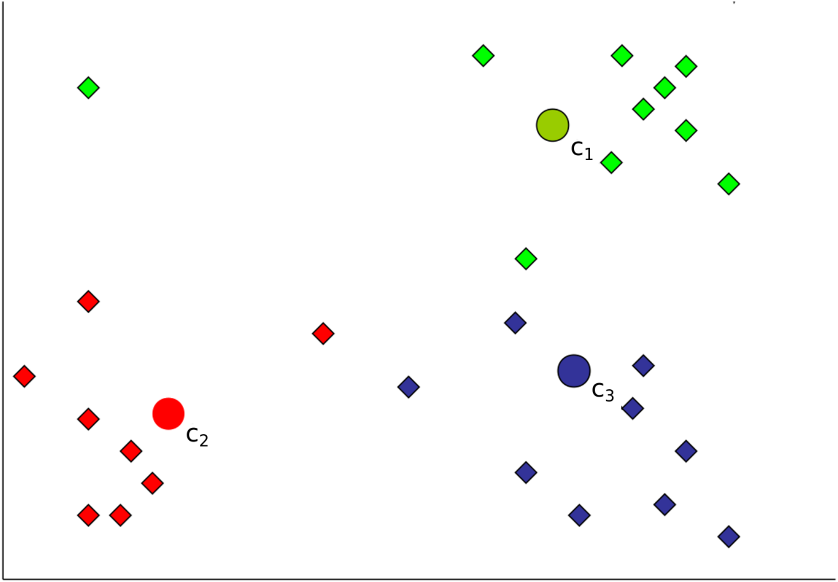
k-means



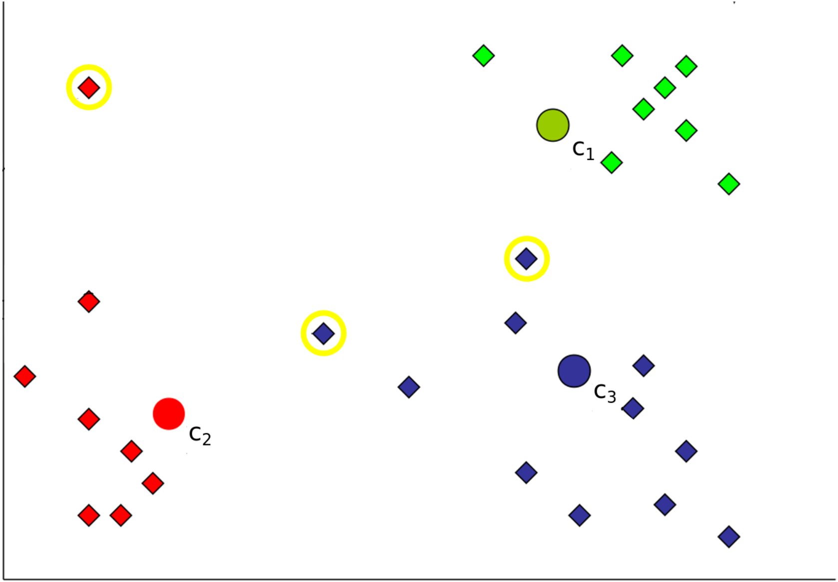
k-means



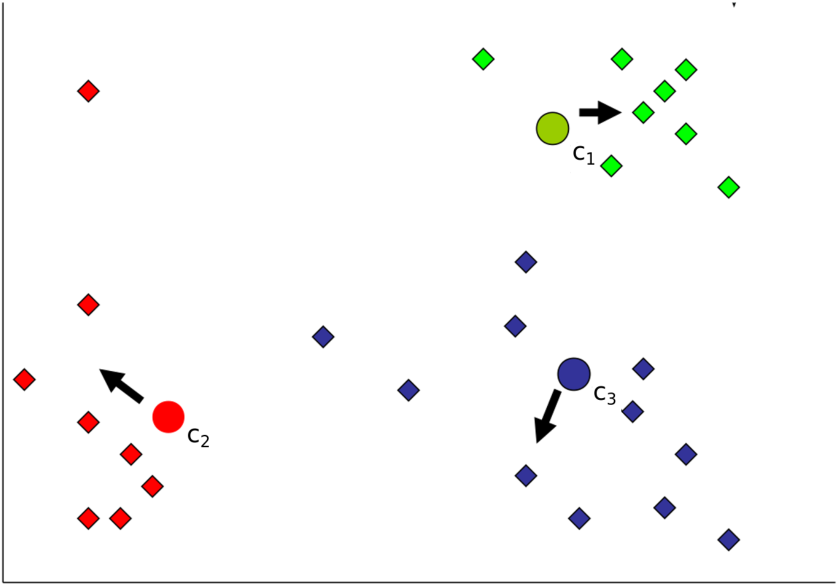
k-means



k-means



k-means



k-means