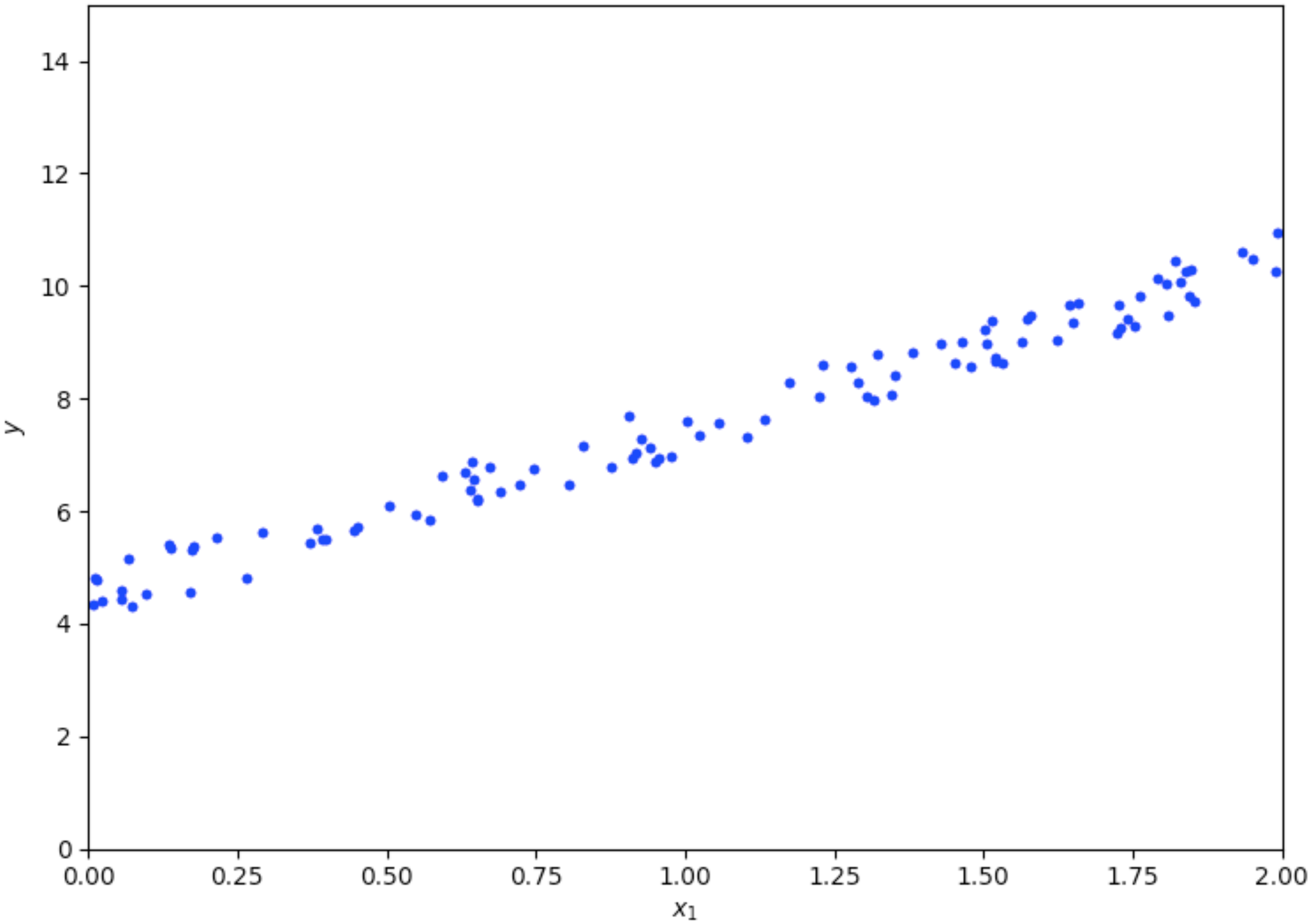
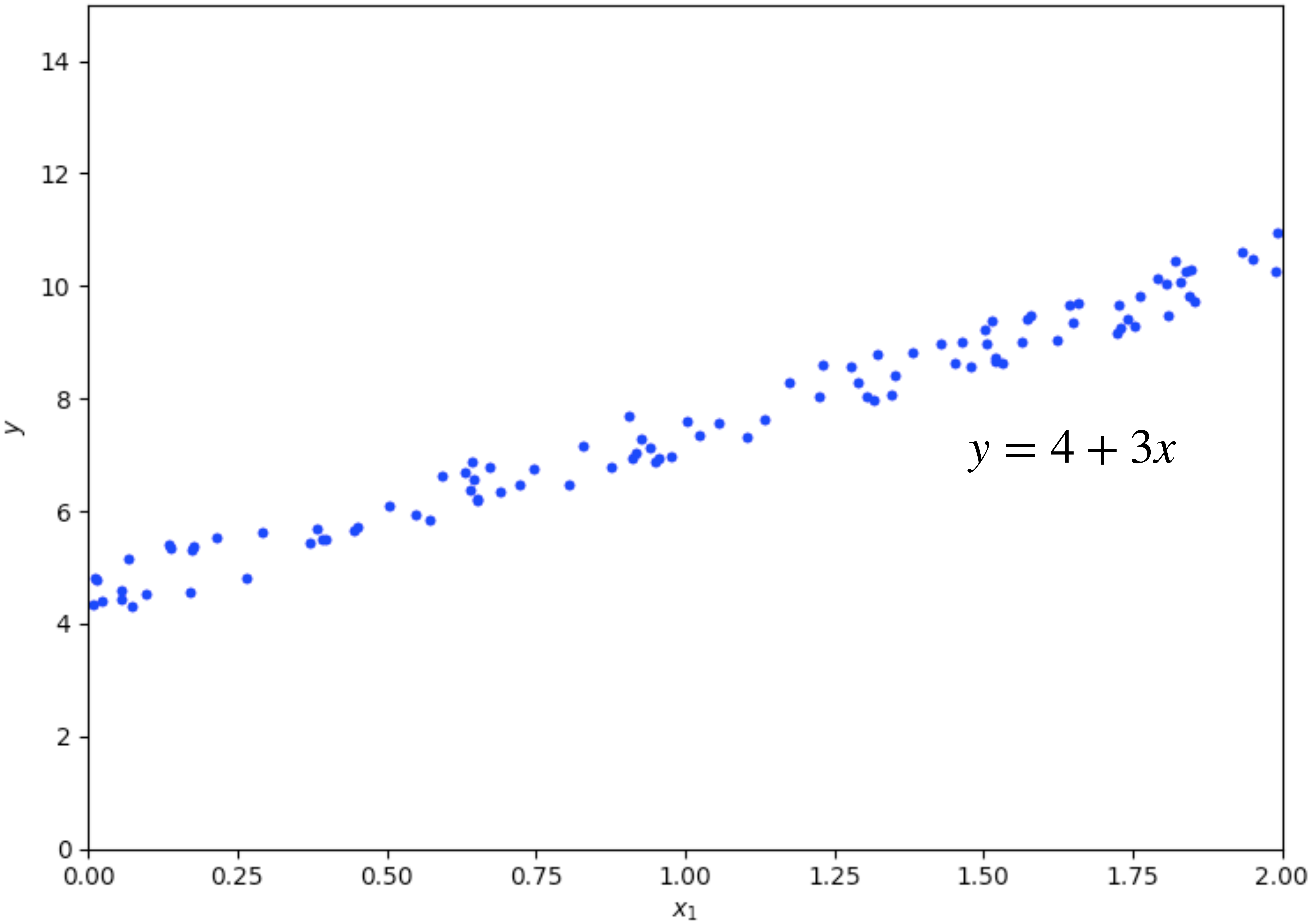
Machine Learning

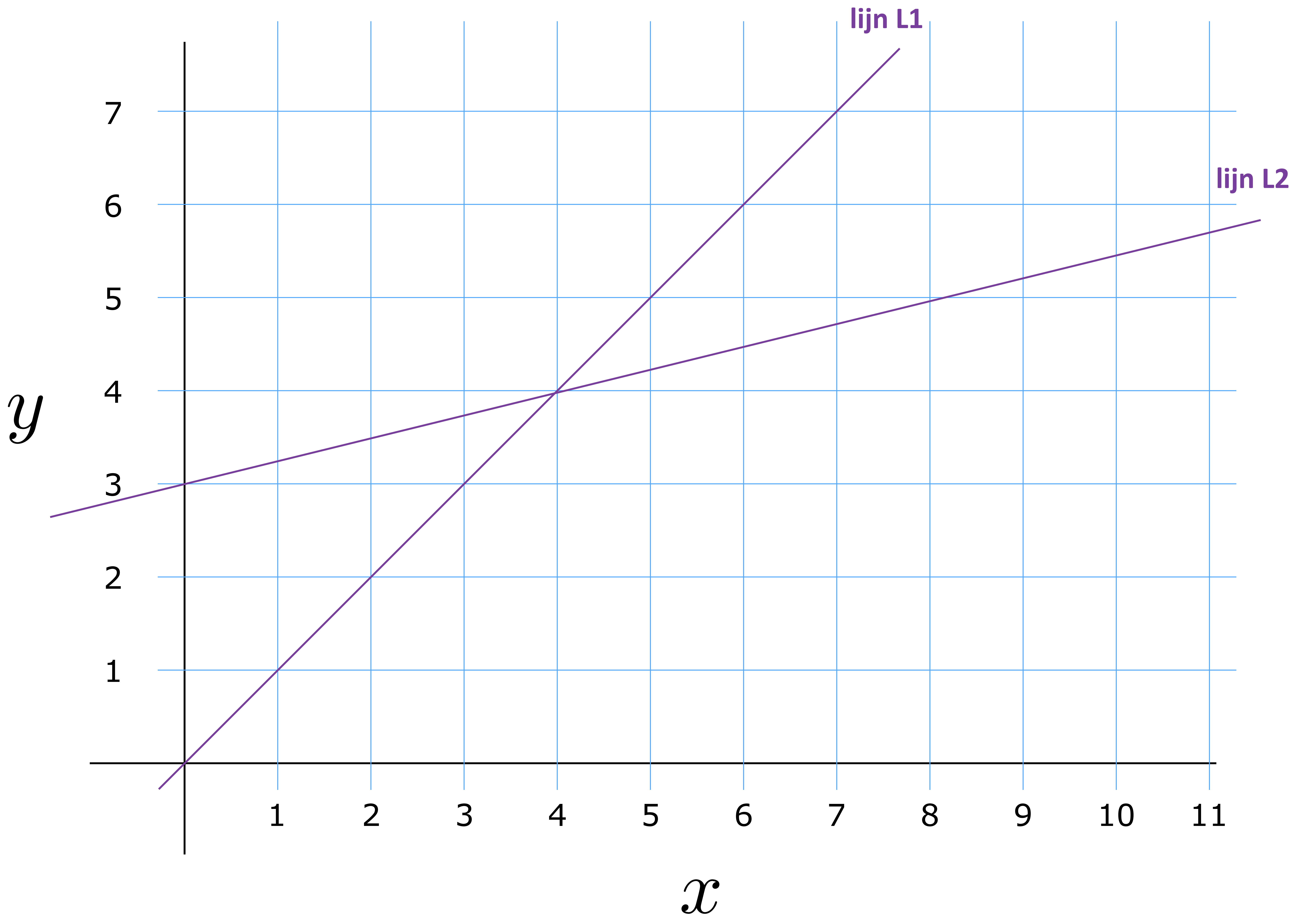
2. regressie en gradient descent

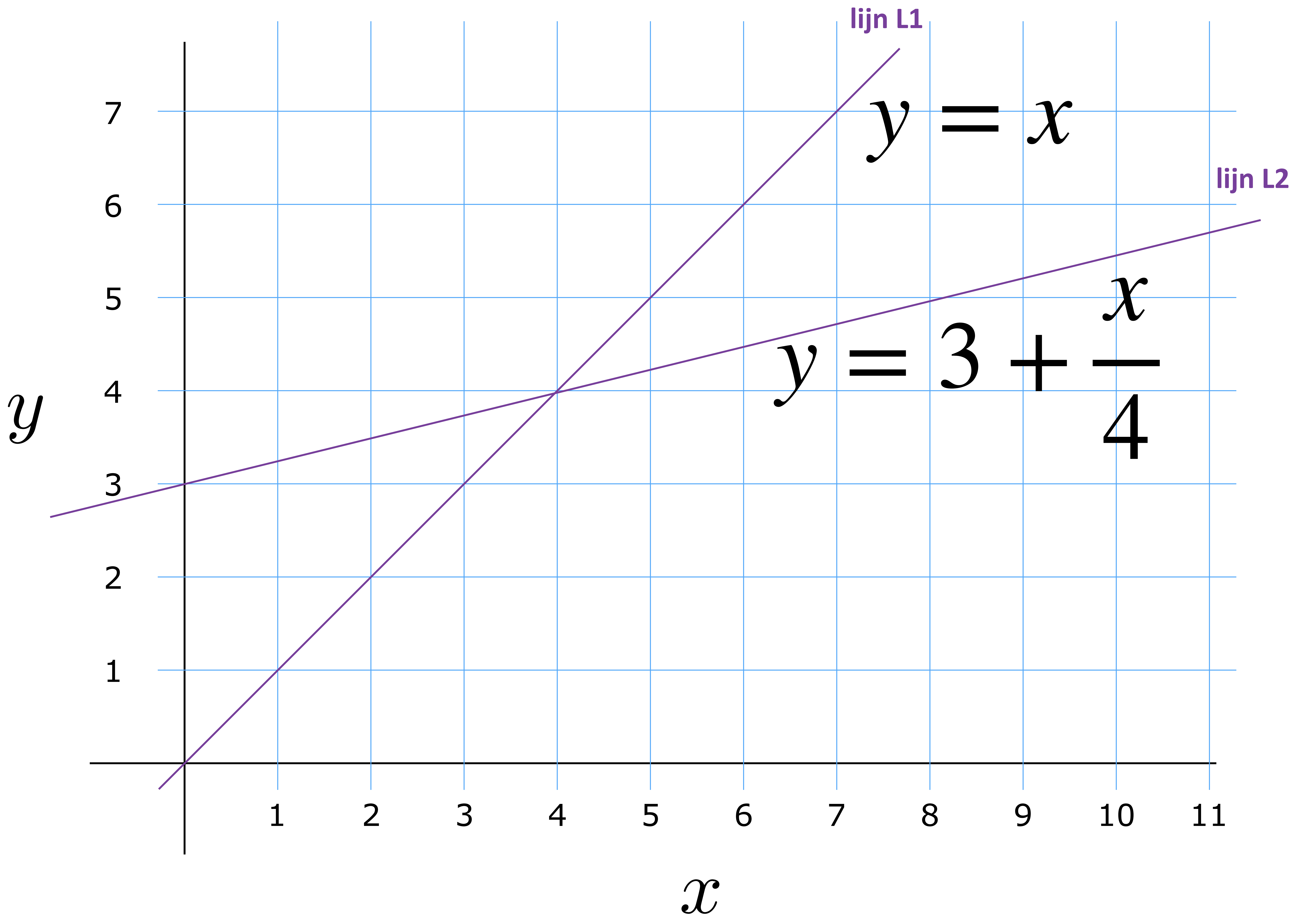
ml:regressie

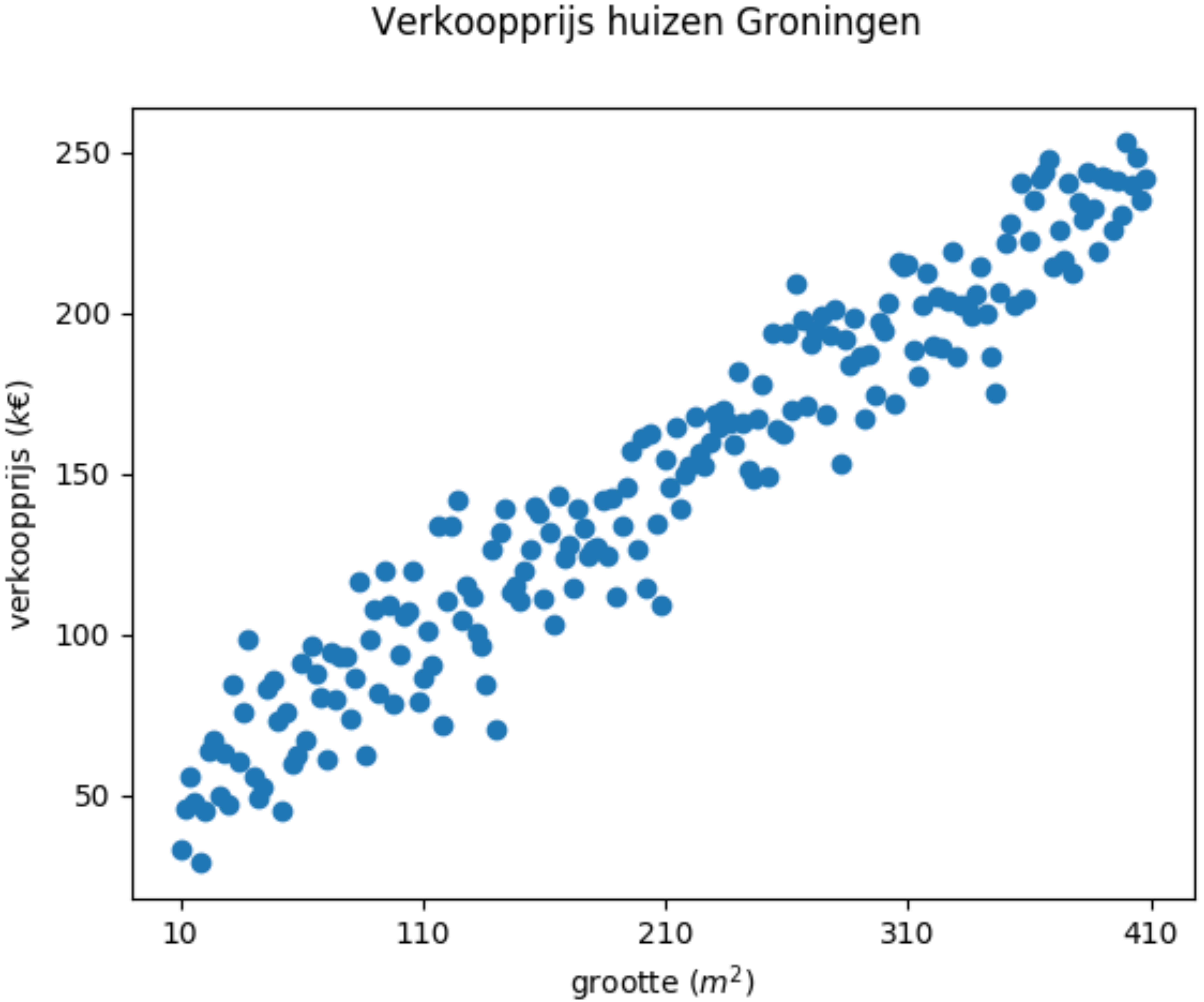


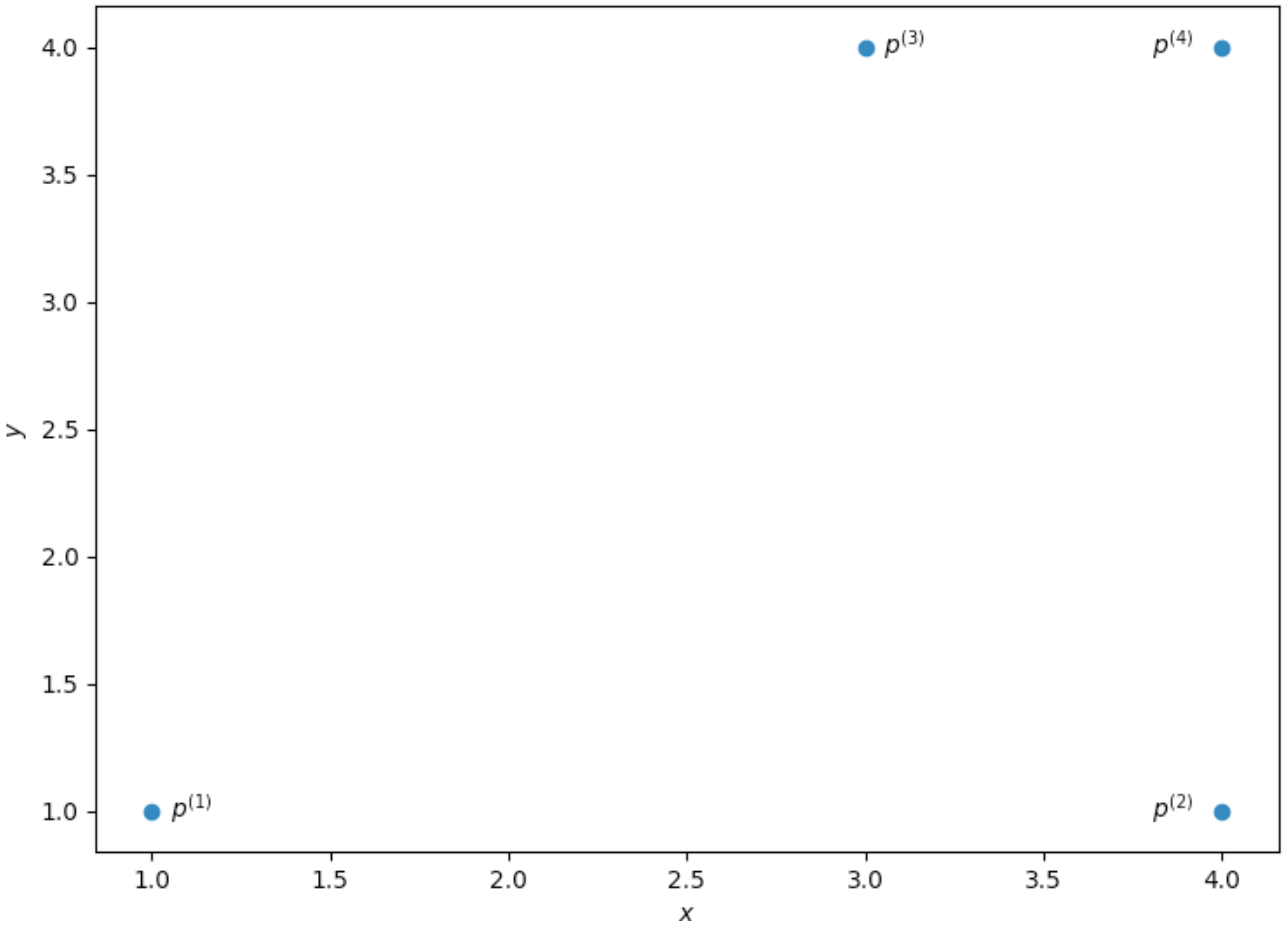
14
12
10
8
6
4
2
0
y
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
x_1


$y = 4 + 3x$
y
x_1
0
2
4
6
8
10
12
14
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

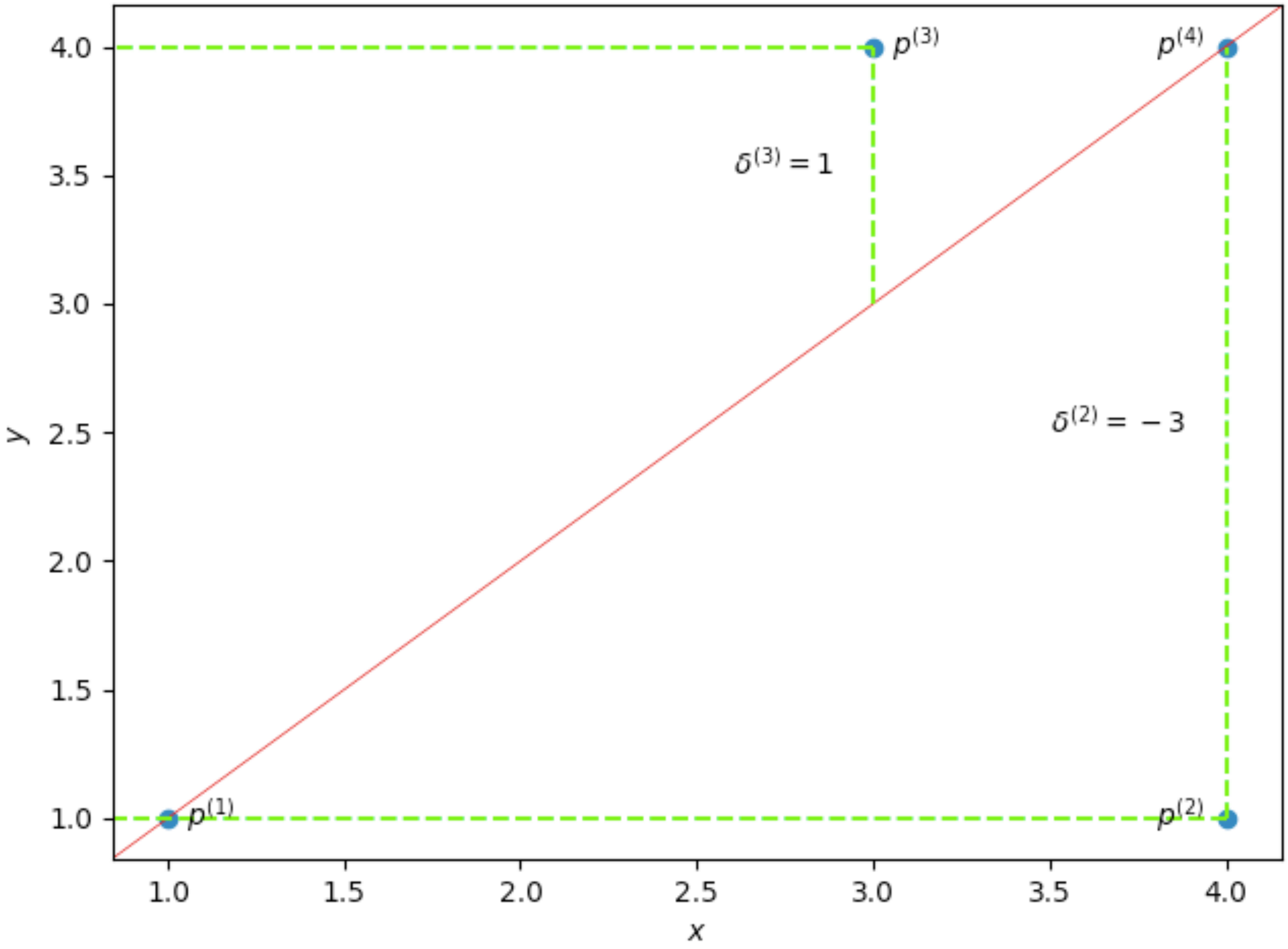

lijn L1
lijn L2
7
6
5
4
3
2
1
y
1
2
3
4
5
6
7
8
9
10
11
x


lijn L1
lijn L2
$y = x$
$y = 3 + \frac{x}{4}$
y
x
1 2 3 4 5 6 7
1 2 3 4 5 6 7 8 9 10 11

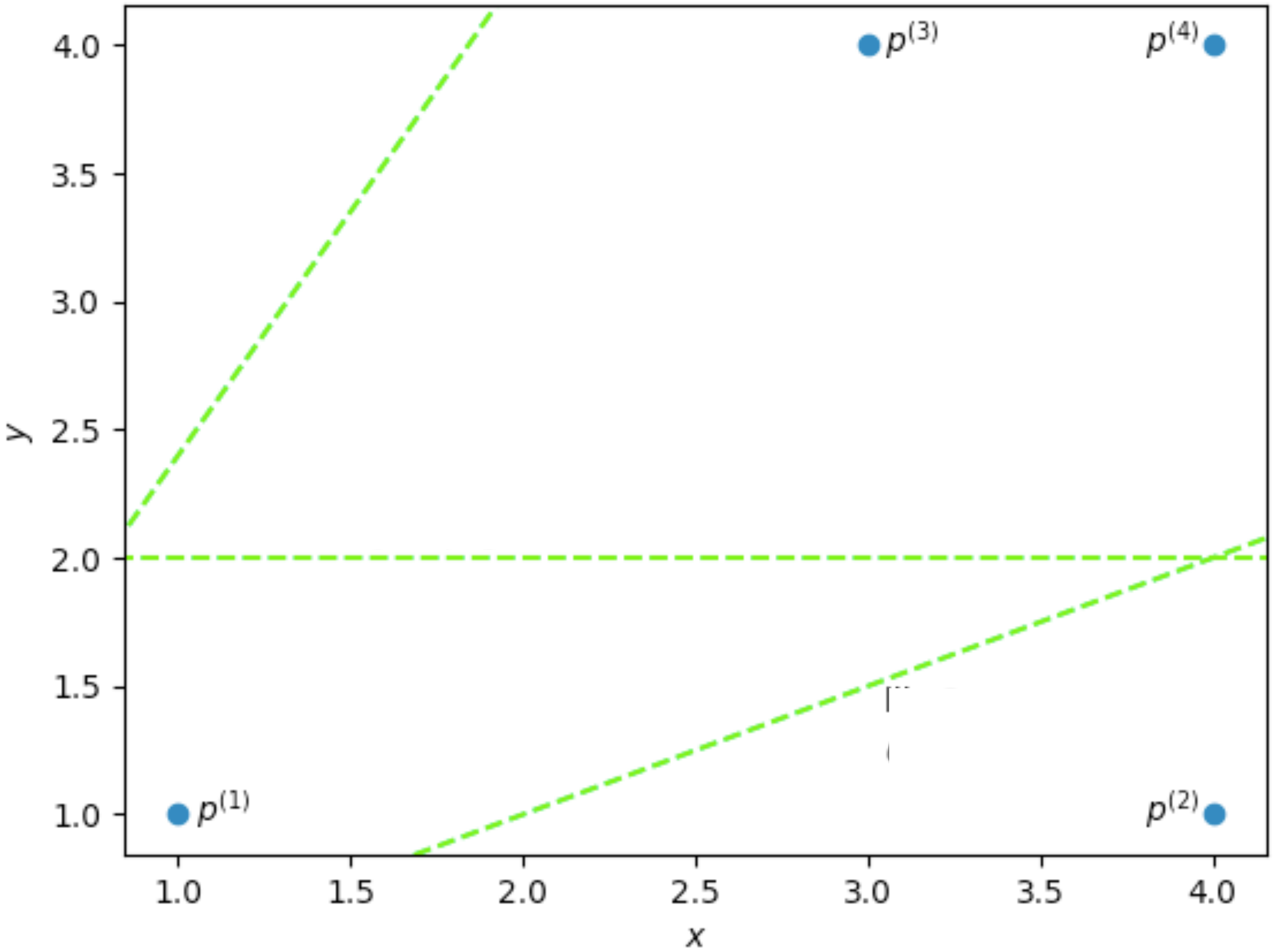

Verkoopprijs huizen Groningen
verkoopprijs ($k€$)
grootte (m^2)
250
200
150
100
50
10
110
210
310
410



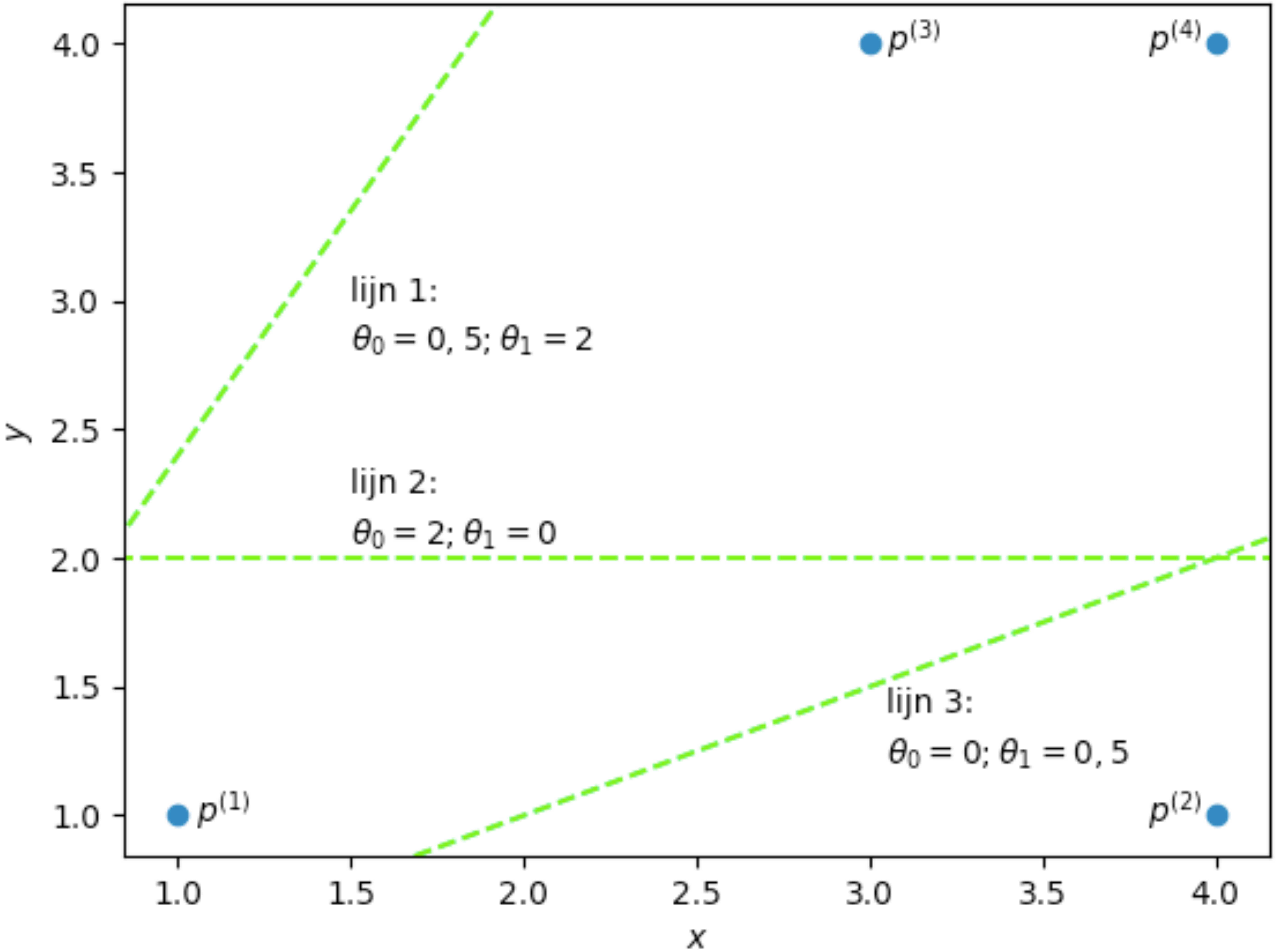
$p^{(3)}$
$p^{(4)}$
$p^{(1)}$
$p^{(2)}$
4.0
3.5
3.0
2.5
2.0
1.5
1.0
1.0
1.5
2.0
2.5
3.0
3.5
4.0
y
x



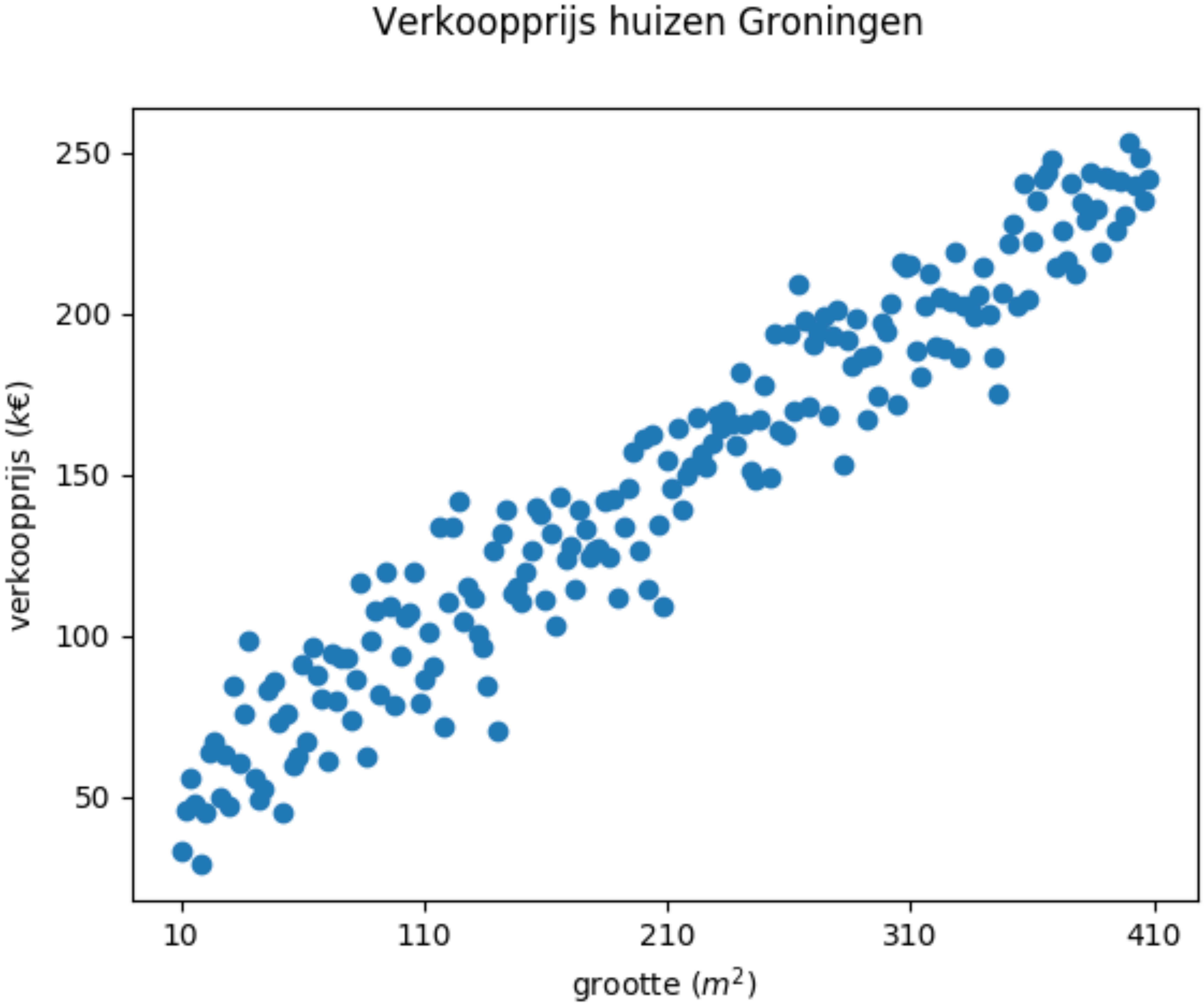
4.0
3.5
3.0
2.5
2.0
1.5
1.0
y
1.0
1.5
2.0
2.5
3.0
3.5
4.0
x
$p^{(1)}$
$p^{(2)}$
$p^{(3)}$
$p^{(4)}$
$\delta^{(3)} = 1$
$\delta^{(2)} = -3$



4.0
3.5
3.0
2.5
2.0
1.5
1.0
y
1.0
1.5
2.0
2.5
3.0
3.5
4.0
x
$p^{(1)}$
$p^{(2)}$
$p^{(3)}$
$p^{(4)}$



4.0
3.5
3.0
2.5
2.0
1.5
1.0
y
1.0
1.5
2.0
2.5
3.0
3.5
4.0
x
$p^{(1)}$
$p^{(2)}$
$p^{(3)}$
$p^{(4)}$
lijn 1:
$\theta_0 = 0,5; \theta_1 = 2$
lijn 2:
$\theta_0 = 2; \theta_1 = 0$
lijn 3:
$\theta_0 = 0; \theta_1 = 0,5$

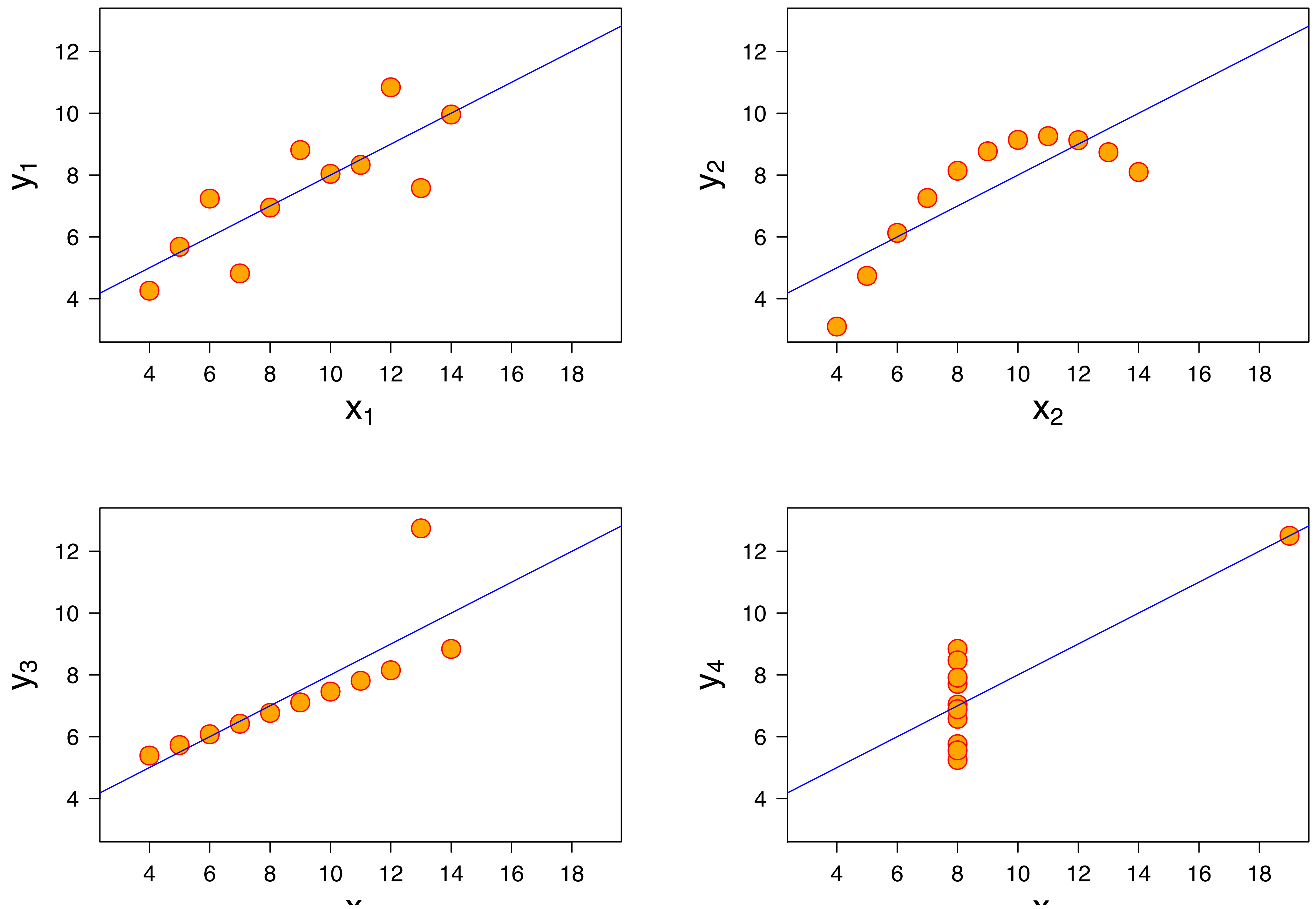


Verkoopprijs huizen Groningen
250
200
150
100
50
verkoopprijs ($k€$)
10
110
210
310
410
grootte (m^2)

Kostenfunctie (J)

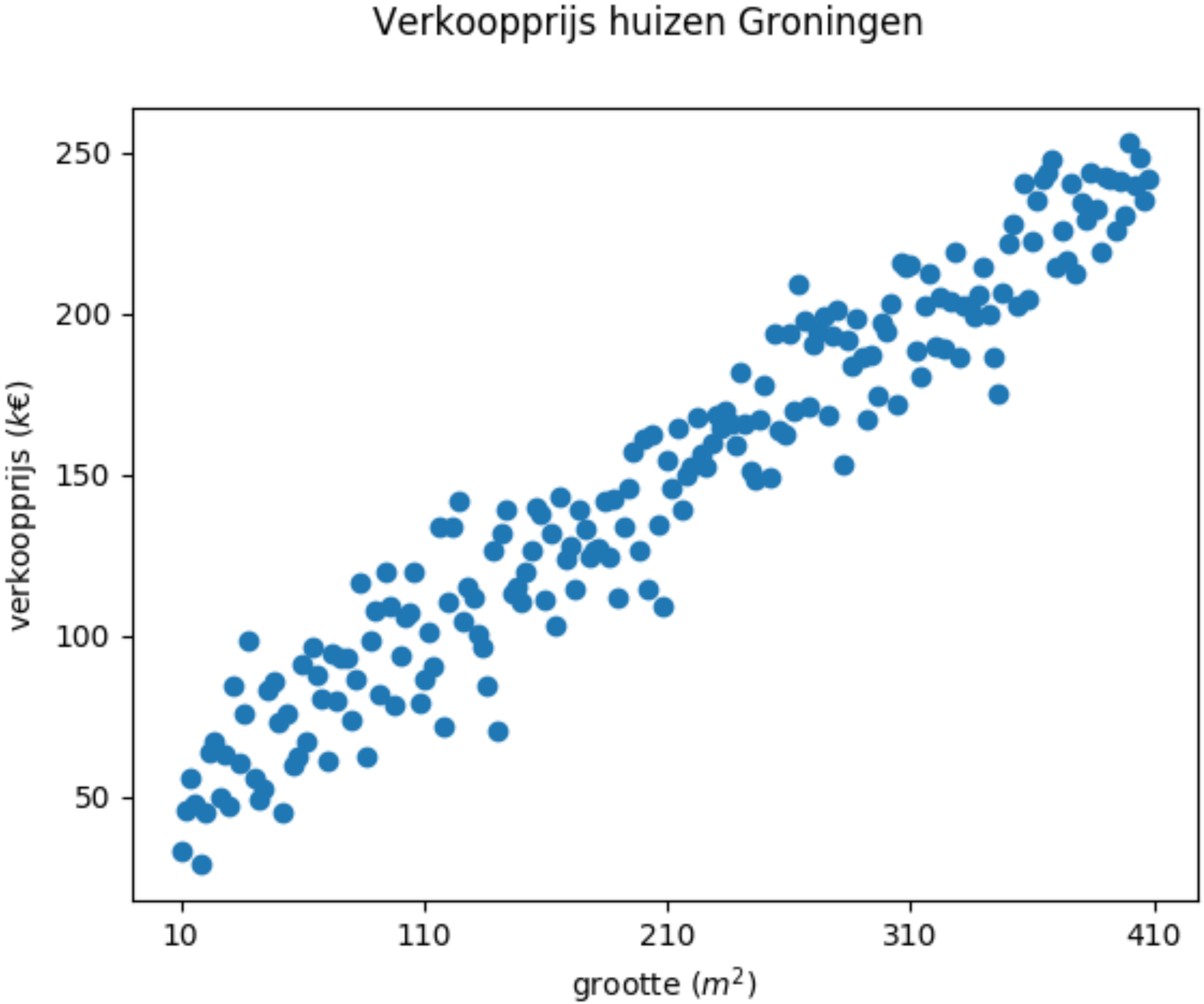
$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

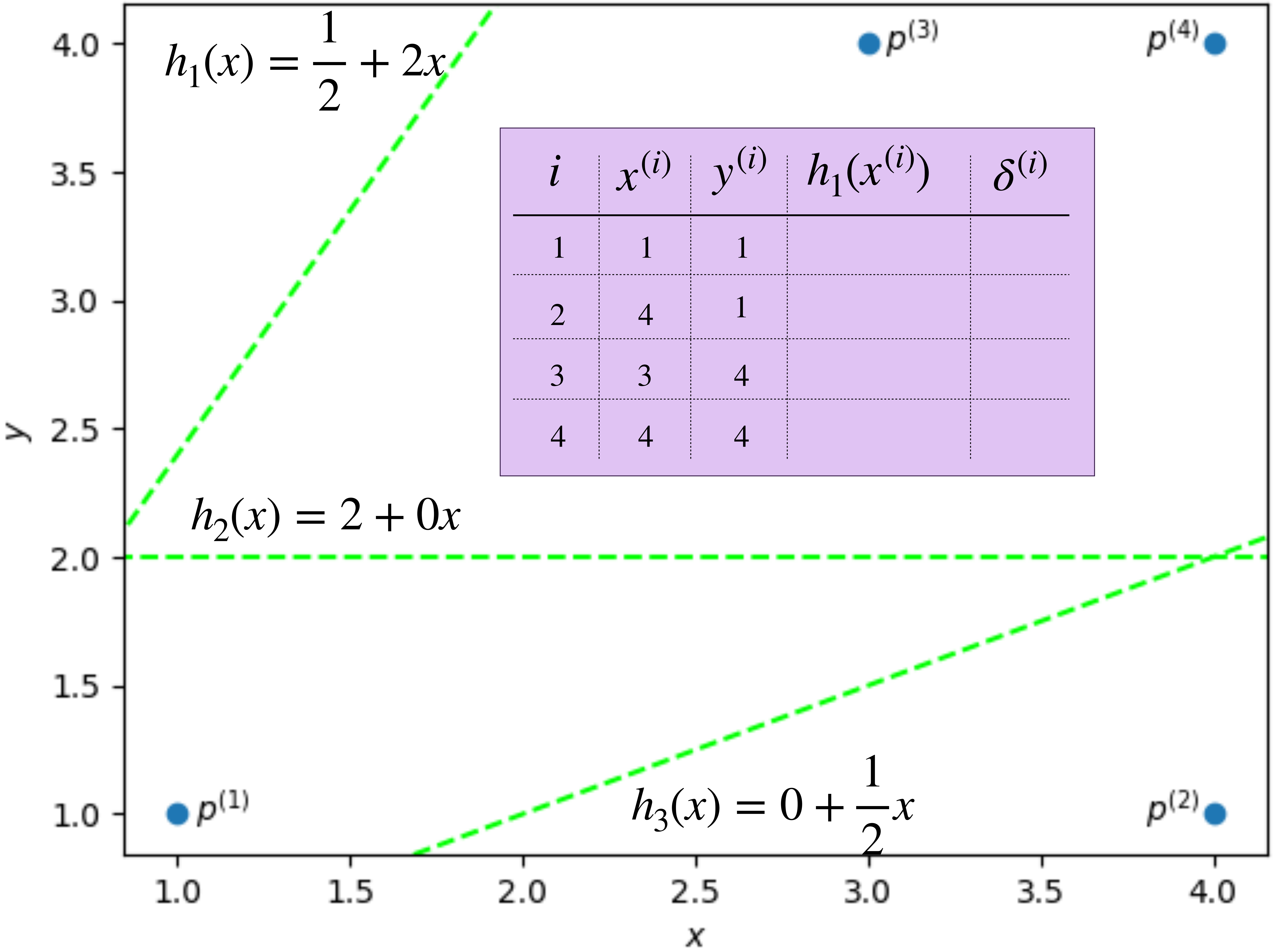
Anscombe’s Quartet



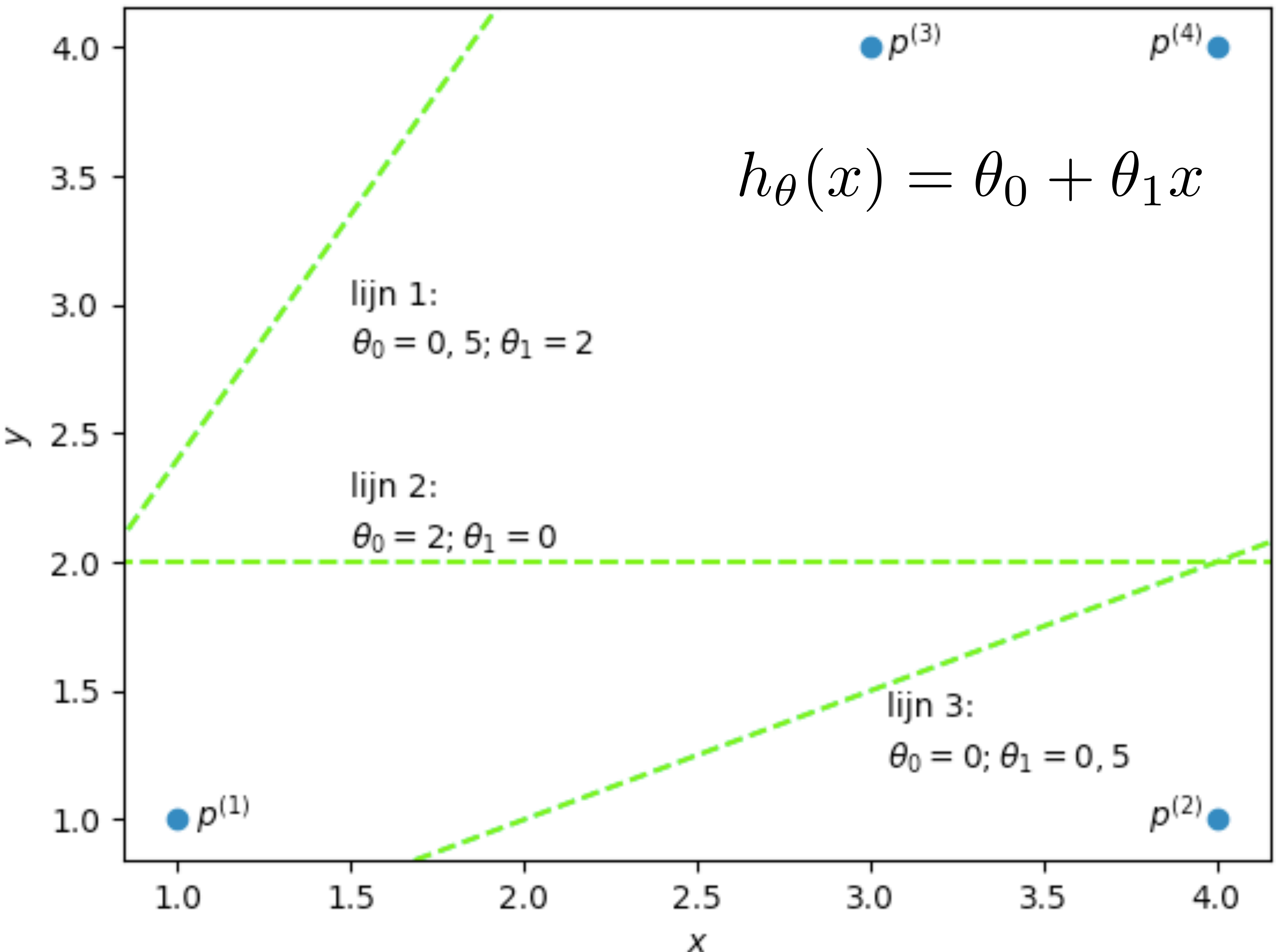
4 zeer verschillende verdelingen met dezelfde optimale regressielijn

ml:meerdere eigenschappen


Verkoopprijs huizen Groningen
verkoopprijs ($k€$)
250
200
150
100
50
10
110
210
310
410
grootte (m^2)



i	$x^{(i)}$	$y^{(i)}$	$h_1(x^{(i)})$	$\delta^{(i)}$
1	1	1		
2	4	1		
3	3	4		
4	4	4		



$h_\theta(x) = \theta_0 + \theta_1 x$
lijn 1:
$\theta_0 = 0,5; \theta_1 = 2$
lijn 2:
$\theta_0 = 2; \theta_1 = 0$
lijn 3:
$\theta_0 = 0; \theta_1 = 0,5$
$p^{(1)}$
$p^{(2)}$
$p^{(3)}$
$p^{(4)}$
x
y
1.0
1.5
2.0
2.5
3.0
3.5
4.0

grootte (m^2)	verkoopprijs (€)
127	279.500
101	195.000
120	167.500
135	290.000
183	534.500
180	315.000
96	189.000
70	115.000
160	449.000
…	…

grootte (m^2)	aantal kamers	tuin	energielabel	verkoopprijs (€)
127	3	j	A	279.500
101	2	n	C	195.000
120	2	j	B	167.500
135	4	j	C	290.000
183	3	n	D	534.500
…	…	…	…	…

notatie-afspraken

m	aantal observaties
n	aantal eigenschappen (per observatie)
$x^{(i)}$	observatie nummer i
$x_j^{(i)}$	eigenschap j van observatie nummer i
$y^{(i)}$	actuele waarde van observatie i
θ_j	factor waarmee waarde van feature j moet worden vermenigvuldigd

$$h_\theta(x) = \theta_0 + \theta_1 x$$

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_4$$

$$X \qquad \theta$$

$$\begin{bmatrix} 127 & 3 & j & A \\ 101 & 2 & n & C \\ 120 & 2 & j & B \\ 135 & 4 & j & C \\ 183 & 3 & n & D \end{bmatrix} \qquad \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix}$$

<u>Intermezzo:</u> vermenigvuldigen van matrices met vectoren

$$A = \begin{bmatrix} 1 & 3 & 2 \\ 4 & 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \\ 5 \end{bmatrix}$$

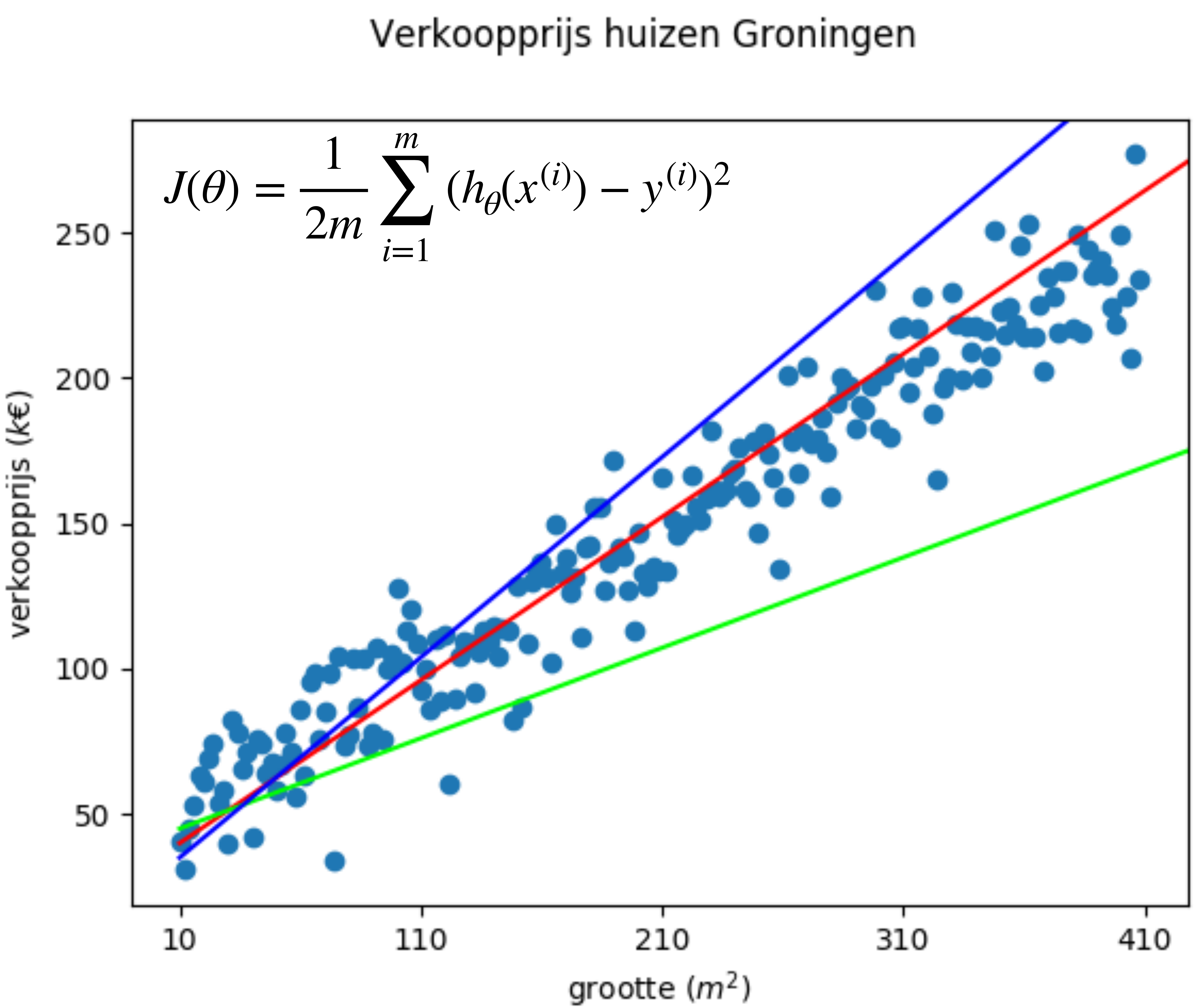
$$X \qquad\qquad\qquad\qquad \theta^T$$

$$\begin{bmatrix} 1 & 127 & 3 & j & A \\ 1 & 101 & 2 & n & C \\ 1 & 120 & 2 & j & B \\ 1 & 135 & 4 & j & C \\ 1 & 183 & 3 & n & D \end{bmatrix} \begin{bmatrix} \theta_0 & \theta_1 & \theta_2 & \theta_3 & \theta_4 \end{bmatrix}$$

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

$$h_\theta(x^{(i)}) = \theta^T x^{(i)}$$

ml:ordinary least square



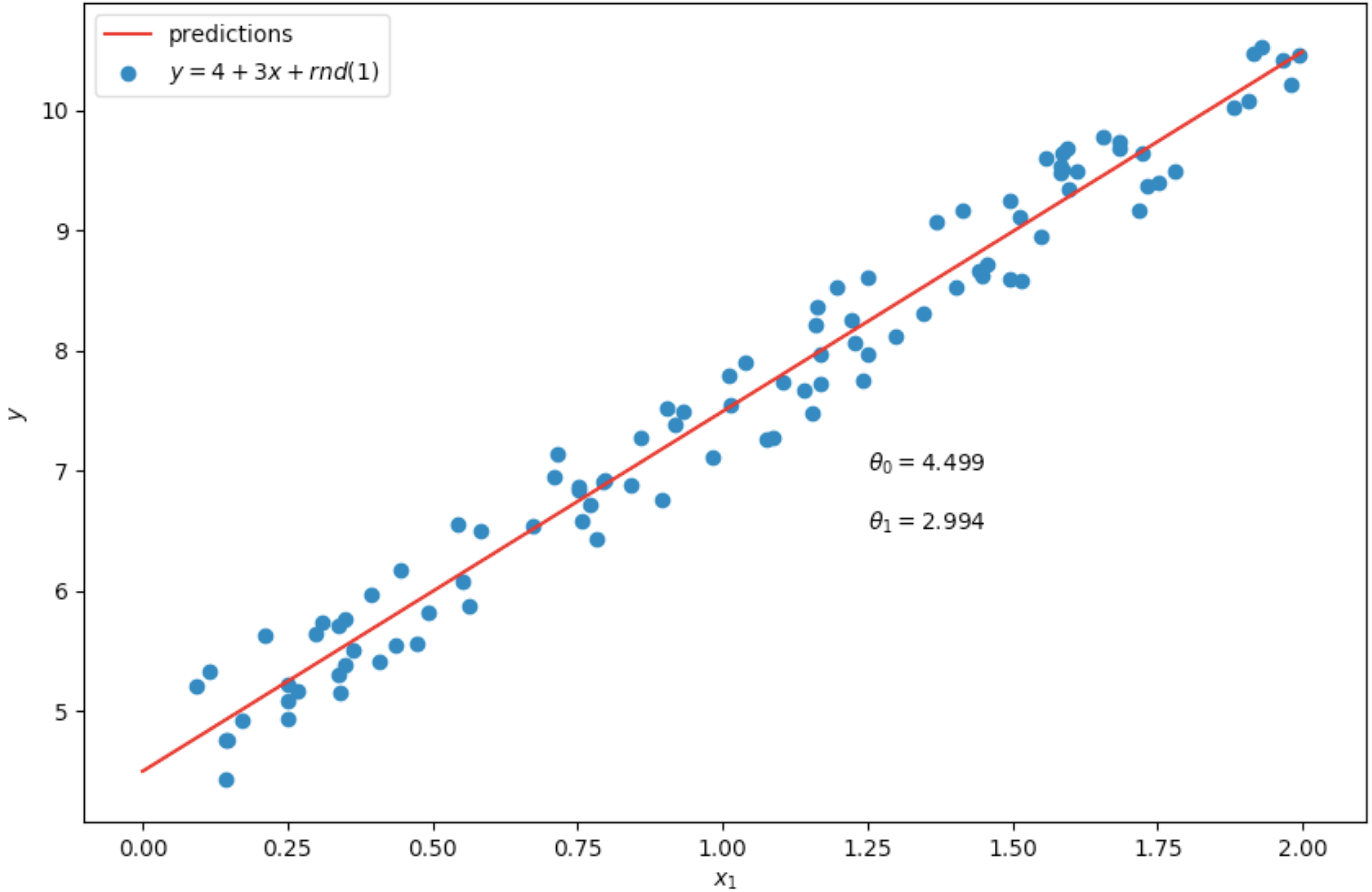
Verkoopprijs huizen Groningen
$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$
verkoopprijs (k€)
250
200
150
100
50
10
110
210
310
410
grootte (m^2)

Ordinary Least Square (OLS)

$$\theta = (X^T \cdot X)^{-1} \cdot X^T \cdot y$$

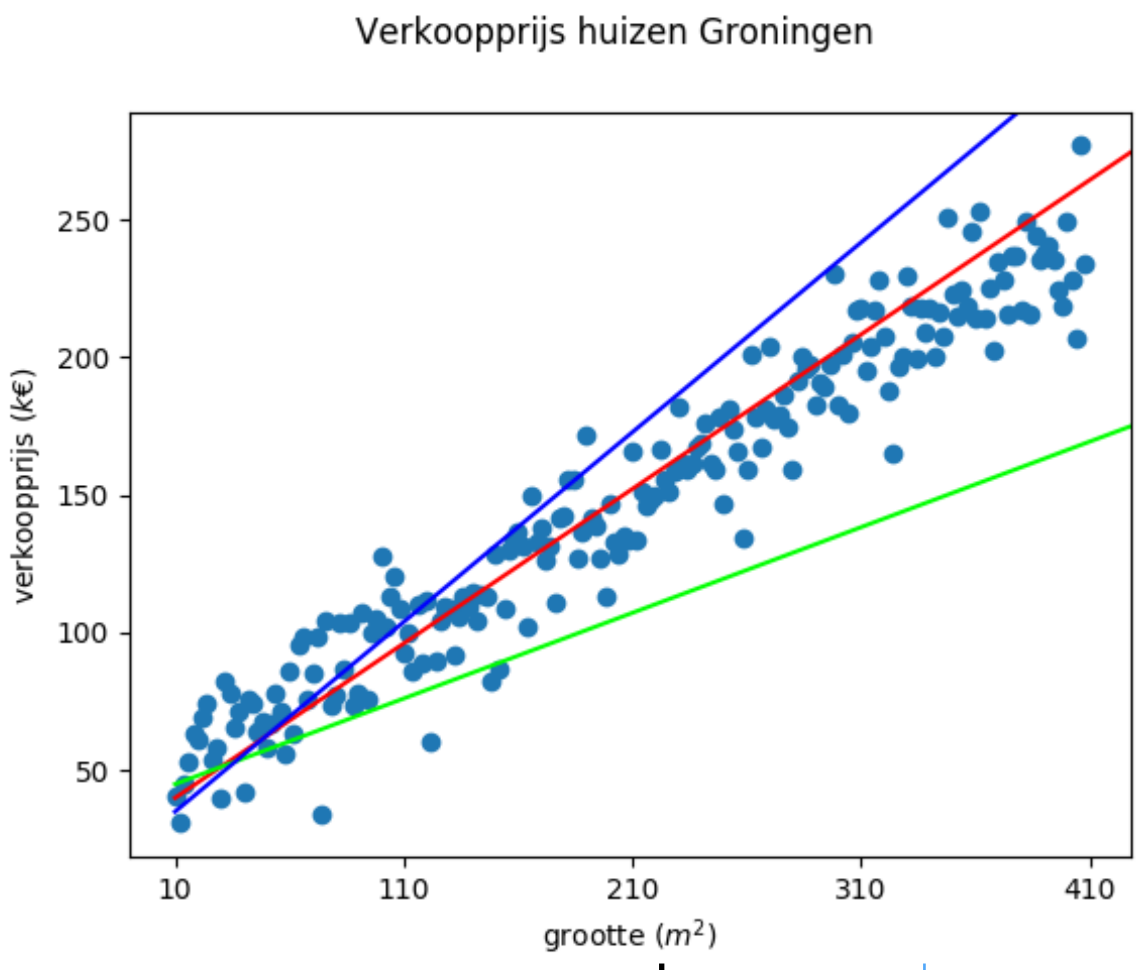
```
theta = np.linalg.inv(X.T.dot(X)).dot(X.T).dot(y)
```

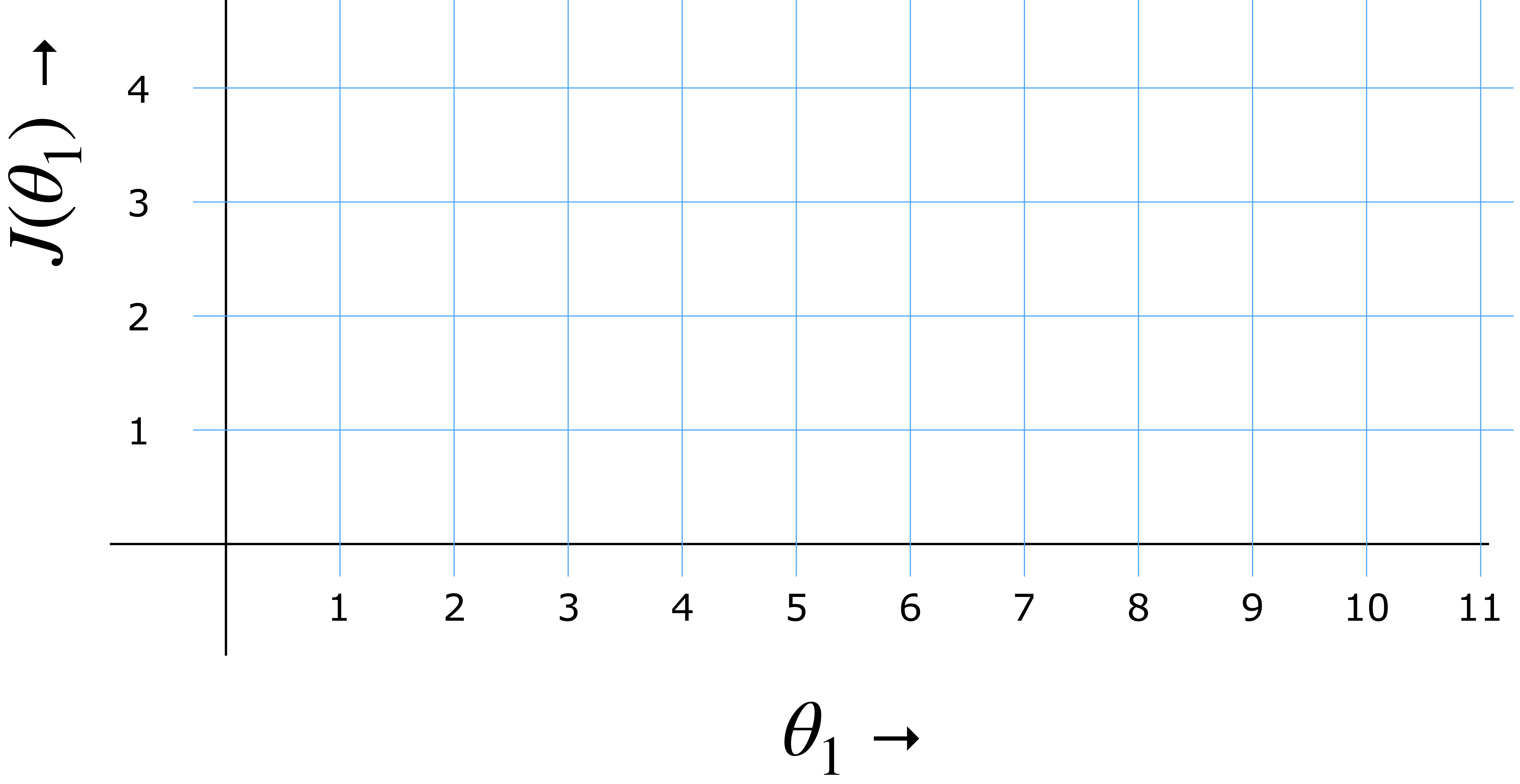
https://en.wikipedia.org/wiki/Ordinary_least_squares

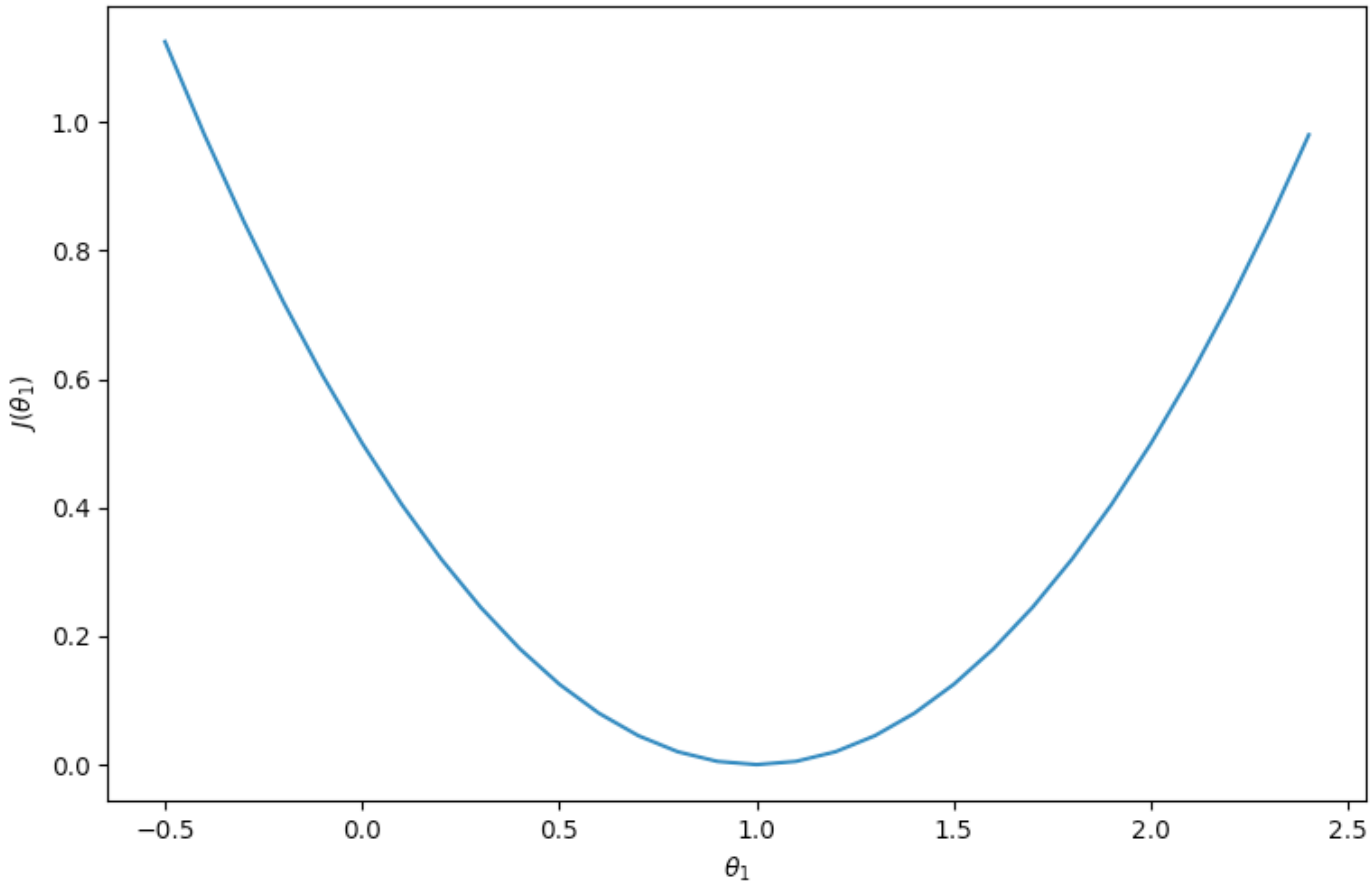


naar Géron 2017, pp 108ff.

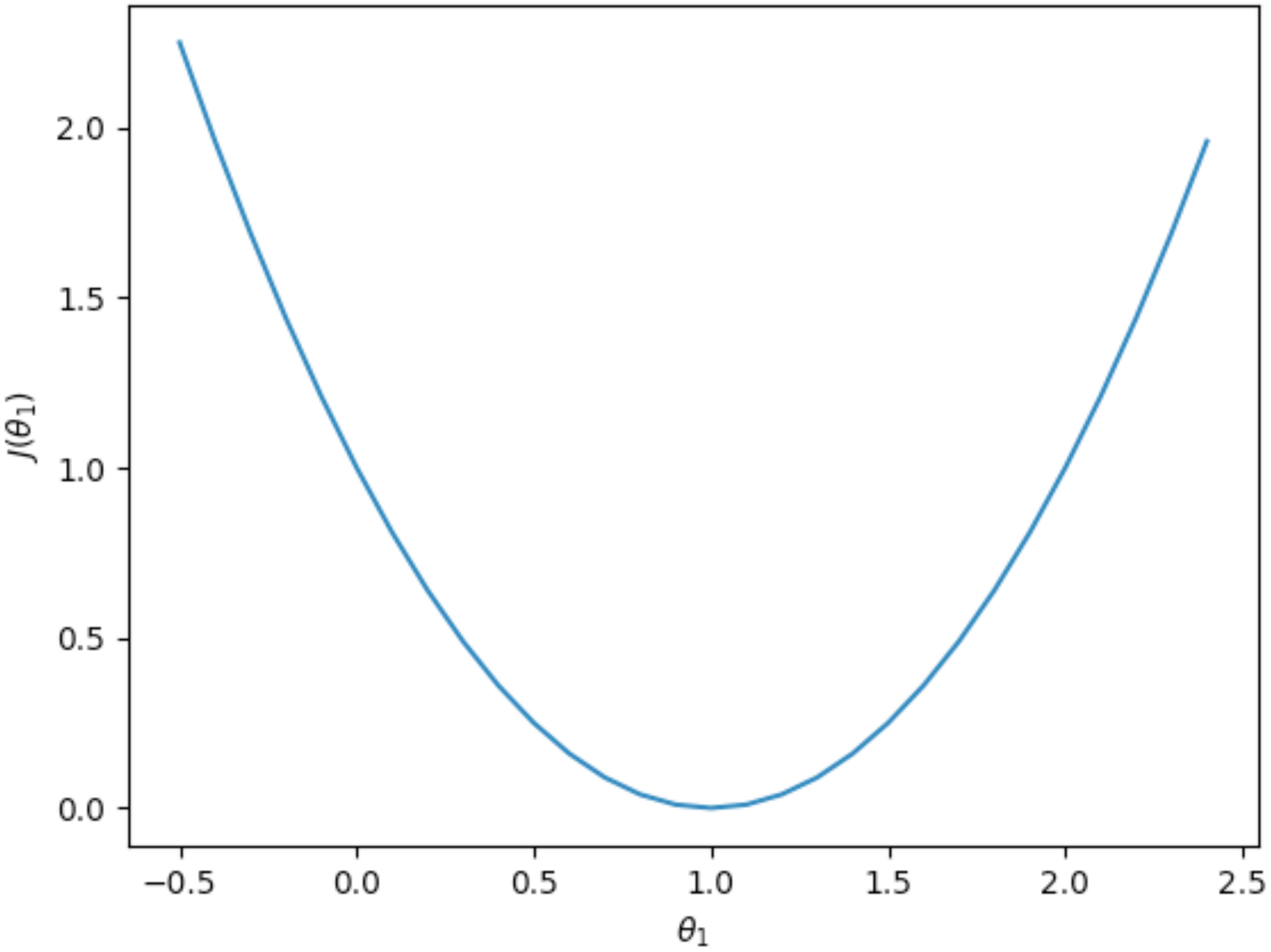
ml:gradient descent


Verkoopprijs huizen Groningen
verkoopprijs (k€)
250
200
150
100
50
10
110
210
310
410
grootte (m^2)


$J(\theta_1) \rightarrow$
4
3
2
1
1
2
3
4
5
6
7
8
9
10
11
$\theta_1 \rightarrow$

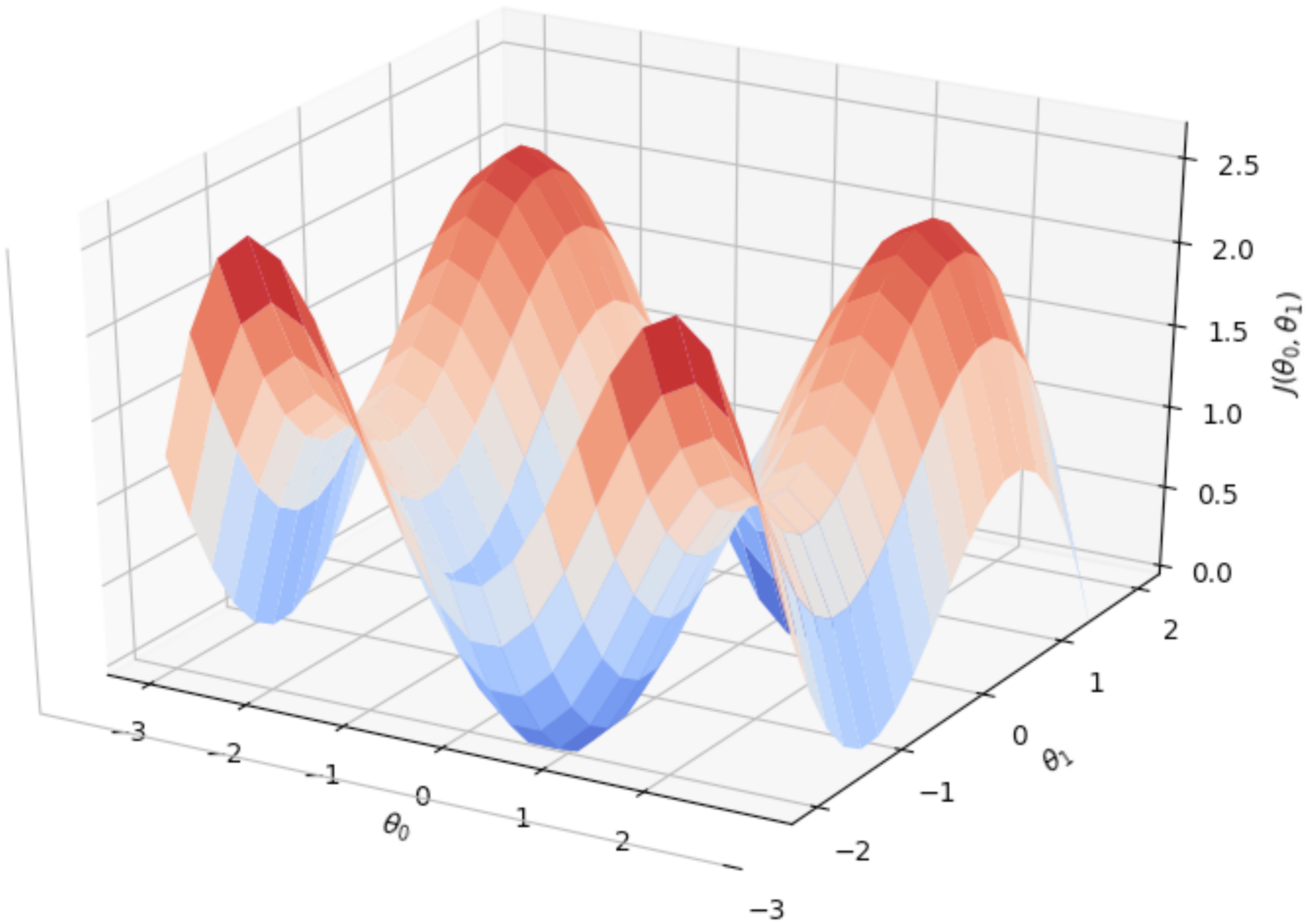


1.0
0.8
0.6
0.4
0.2
0.0
$J(\theta_1)$
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
θ_1



$$y = \frac{1}{2}(1 - x)^2$$

http://www.tuinposter.nl/herfst/



2.5
2.0
1.5
1.0
0.5
0.0
$J(\theta_0, \theta_1)$
−3
−2
−1
0
1
2
θ_0
2
1
0
−1
−2
−3
θ_1

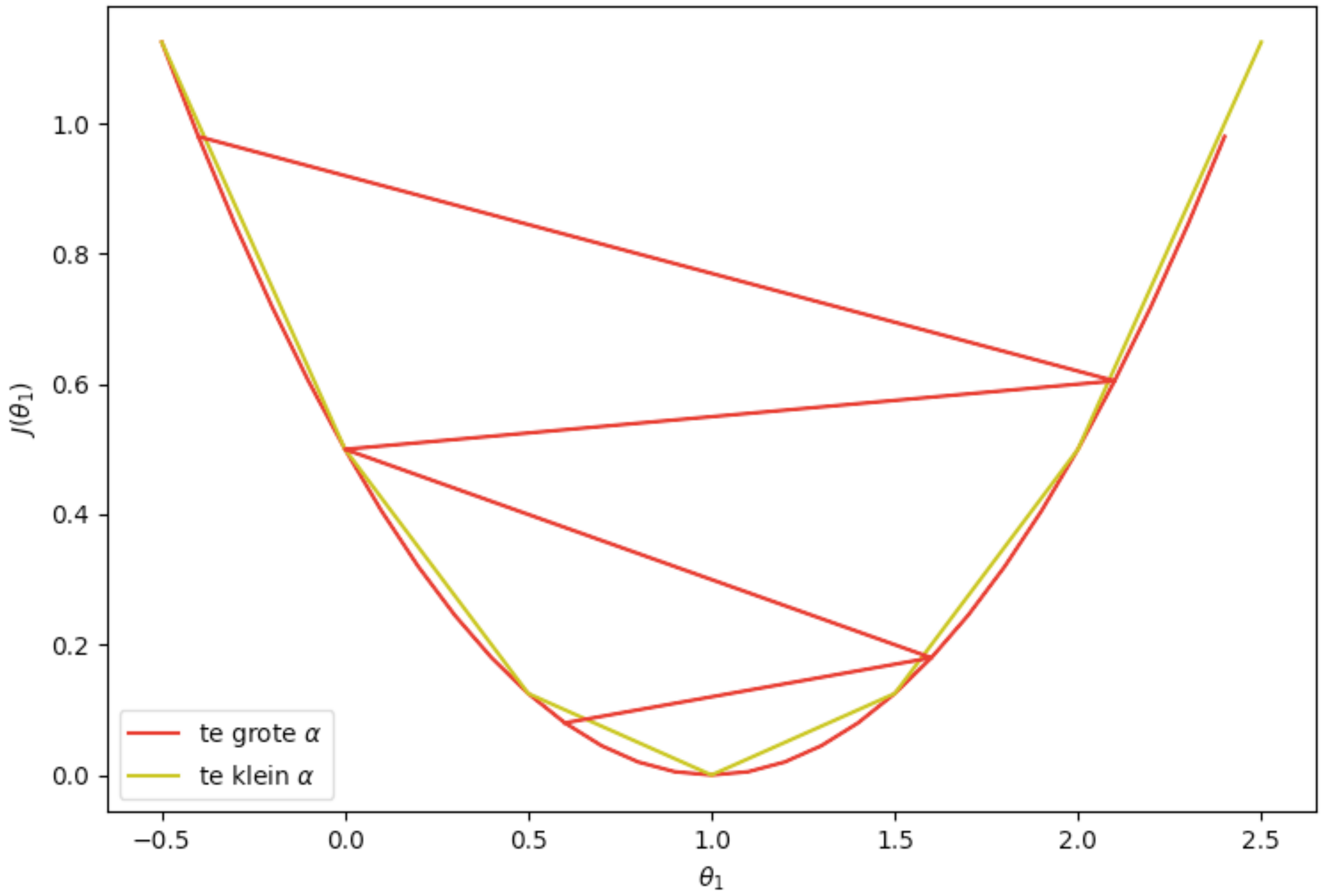
$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

$$h_\theta(x) = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

$$= \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$$



te grote α
te klein α
$J(\theta_1)$
θ_1
0.0
0.2
0.4
0.6
0.8
1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5

update alle $\theta_j, j = 1, j = 2, \ldots, j = n$

herhaal totdat een minimum bereikt is:

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

$$:= \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

Thanks to machine-learning algorithms,
the robot apocalypse was short-lived.