L12: Multi-Linear and Polynomial Regression

Feb 18, 2026

FoDA

Jeff M. Phillips

Input

$$(X, y) \subset \mathbb{R}^d \times \mathbb{R}$$

$$(X, y) = \{ (x_1, y_1), (x_2, y_2), \dots (x_n, y_n) \}$$

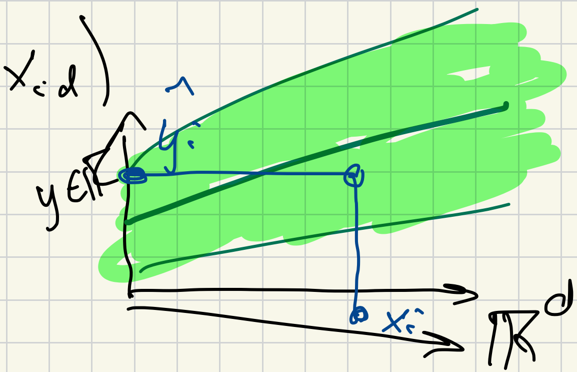
explanatory dependent

$$x_i \in \mathbb{R}^d \qquad y_i \in \mathbb{R}$$

$$x_i = (x_{i1}, x_{i2}, x_{i3} \dots x_{id})$$

Goal

$$\alpha = (\alpha_0, \alpha_1, \dots, \alpha_d) \in \mathbb{R}^{d+1}$$



$$\hat{y}_i = M_\alpha(x_i) = \alpha_0 + \sum_{j=1}^{d} \alpha_j x_{ij}$$

$$= \alpha_0 1 + \alpha_1 x_{i1} + \alpha_2 x_{i2} + \dots + \alpha_d x_{id}$$

$$= \langle \alpha, (1, x_i) \rangle = \langle \alpha, (1, x_{i1}, x_{i2} \dots x_{id}) \rangle$$

time: X_1	jiggle: X_2	scroll: X_3	sales: y
232	33	402	2201
10	22	160	0
6437	343	231	7650
512	101	17	5599
441	212	55	8900
453	53	99	1742
2	2	10	0
332	79	154	1215
182	20	89	699
123	223	12	2101
424	32	15	8789

x_1

x_2

y_2

X_1 X_2 X_3

$\hat{X} \in \mathbb{R}^{n \times (d+1)} \implies \alpha \in \mathbb{R}^{d+1}$

(X, y)

$X = [X_1; X_2; X_3]$

$\hookrightarrow \tilde{X} [1; X_1, X_2, X_3]$

Goal new

$x_1 = 500$

$x_2 = 30$

$x_3 = 50$

$M_\alpha(x_1, x_2, x_3)$

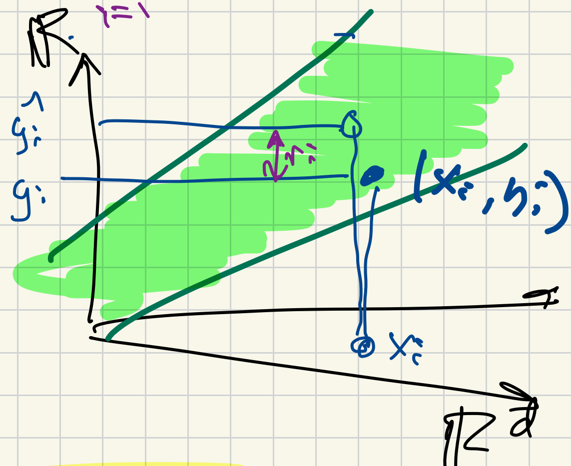
$= \alpha_0 + 500\alpha_1 + 30\alpha_2 + 50\alpha_3$

Input $X \in \mathbb{R}^{n \times d}$ $y \in \mathbb{R}^n$

Goal $SSE((X, y), M_\alpha) = S(\alpha)$

$= \sum_{i=1}^{n} (y_i - M_\alpha(x_i))^2 = \sum_{i=1}^{n} (r_i)^2$

$S(\alpha) = \sum_{i=1}^{n} (y_i - \langle \alpha, (1, x_i) \rangle)^2$



$$\alpha^* = \underset{\alpha \in \mathbb{R}^{d+1}}{\arg\min}\ SSE((X, y), M_\alpha)$$

$$\underset{\alpha \in \mathbb{R}^{d+1}}{\arg\min}\ S(\alpha) = \alpha^* = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T y$$

algorithm

Algorithm for Multi linear Regression

0. Input $(X, y) \in \mathbb{R}^{n \times d} \times \mathbb{R}^{n}$

1. Convert $X \to \tilde{X} \in \mathbb{R}^{n \times (d+1)}$ $\quad \tilde{X} = \begin{bmatrix} 1 & \\ \vdots & X \\ 1 & \end{bmatrix}$

   Σ all 1s column

2. $Z = \tilde{X}^T \tilde{X} \in \mathbb{R}^{(d+1) \times (d+1)}$

3. $(\tilde{X}^T \tilde{X})^{-1}$ only if Z is
   - square
   - full rank.

   good if $n >> d$.

4. $\alpha^* = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T y$

$$\alpha^* = (\hat{X}^T \hat{X})^{-1} \hat{X}^T y$$

$d=1$

$$a = \frac{\langle x, y \rangle}{\|x\|^2}$$

not solving each α_i at a time

solving for all $\alpha_i \in \alpha$ globally

all at once.

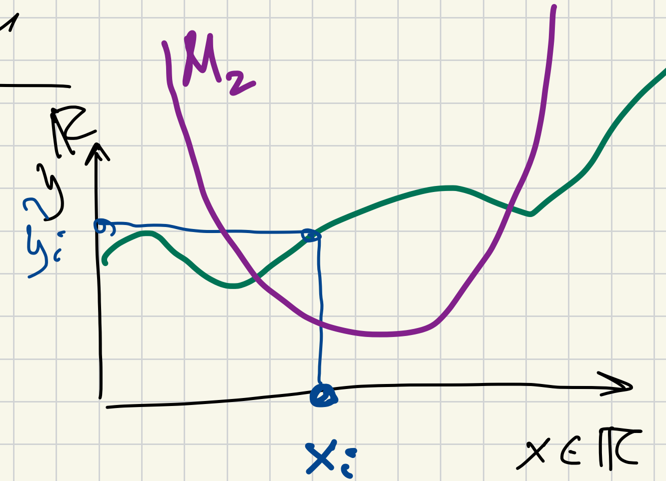
maybe x_{i1} is important?

do not read too much info.

$$\hat{y}_i = \alpha_0 + \alpha_1 x_{i1} + \alpha_2 x_{i2} + \dots + \alpha_d x_{id}$$

if large

Polynomial Regression



Input $(x, y) \in \mathbb{R} \times \mathbb{R}$

$= \{(x_i, y_i)\}$

$x_i \in \mathbb{R}$

$y_i \in \mathbb{R}$

$$\hat{y} = M_{p=2}(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2$$

$$\hat{y} = M_p(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \dots + \alpha_p x^p$$

$$= \alpha_0 + \sum_{j=1}^{p} \alpha_j x^p = \sum_{j=0}^{p} \alpha_j x^p = \langle \alpha, (1, x, x^2, \dots, x^p) \rangle$$

Polynomial Regression, degree p

$(\cancel{X}, y)$ $\quad x_i \in \mathbb{R}$

$\hookrightarrow v_i = (1, x, x^2, \dots x^p) \in \mathbb{R}^{p+1}$

$M_{p,\alpha}(x_i) = \langle \alpha, v_i \rangle$

$= \langle \alpha, (1, x_i, x_i^2, \dots, x_i^p) \rangle$

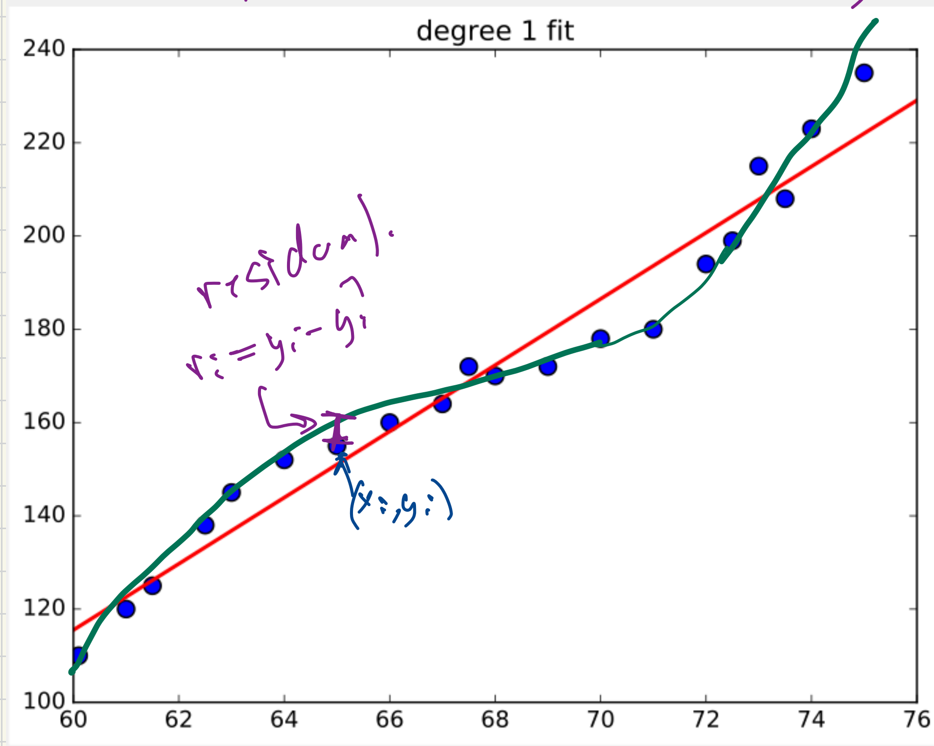
vector

$$\begin{bmatrix} \\ x \\ \\ \end{bmatrix} \in \mathbb{R}^n \Rightarrow \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^p \\ 1 & x_2 & x_2^2 & \cdots & x_2^p \\ 1 & x_3 & & & \\ 1 & \vdots & & & \\ 1 & x_n & x_n^2 & \cdots & x_n^p \end{bmatrix} \in \mathbb{R}^{n \times (p+1)} = \tilde{X}_p$$

height (in)	weight (lbs)
66	160
68	170
60	110
70	178
65	155
61	120
74	223
73	215
75	235
67	164
61.5	125
73.5	208
62.5	138
63	145
64	152
71	180
69	172
72.5	199
72	194
67.5	172

$$SSE((x,y), M_{p,\alpha}) = \sum_{i=1}^{n} (y_i - M_{p,\alpha}(x_i))^2$$

$$\alpha^* = \arg\min_{\alpha \in \mathbb{R}^{p+1}} SSE((x,y), M_{p,\alpha})$$



Input $(x, y) \in \mathbb{R}^n \times \mathbb{R}^n$

Model $M_{p,\alpha}(x) = \sum_{j=0}^{p} \alpha_j x^j \qquad \alpha = (\alpha_0, \alpha_1, \dots, \alpha_p)$

Goal $\alpha^* = \underset{\alpha \in \mathbb{R}^{p+1}}{\arg\min} \sum_{i=1}^{n} \left(y_i - M_{p,\alpha}(x_i)\right)^2$

$$\tilde{X}_p = \begin{bmatrix} 1 & x_1 & x_1^2 & & x_1^p \\ 1 & x_2 & x_2^2 & \dots & x_2^p \\ \vdots & & & & \\ 1 & x_n & x_n^2 & & x_n^p \end{bmatrix} \qquad v_i \in \mathbb{R}^{p+1}$$

$$\alpha^* = \left(\tilde{X}_p^T \tilde{X}_p\right)^{-1} \tilde{X}_p^T y$$

algorithm

$$x = \begin{bmatrix} 2 \\ 4 \\ 3 \end{bmatrix} \qquad y = \begin{bmatrix} 1 \\ 6 \\ 5 \end{bmatrix}$$

$$p = 5$$

$$\hat{X}_p = \begin{bmatrix} 1 & 2 & 4 & 8 & 16 & 32 \\ 1 & 4 & 16 & 64 & 256 & 1024 \\ 1 & 3 & 9 & 27 & 81 & 243 \end{bmatrix}$$

$$\in \mathbb{R}^{3 \times 6}$$

$$\left(\hat{X}_p^T \hat{X}_p\right)^{-1}$$