

729G78 Artificiell intelligens

Maskininlärning – Linjär regression

Robin Keskisärkkä

HCS/IDA

Val av modell

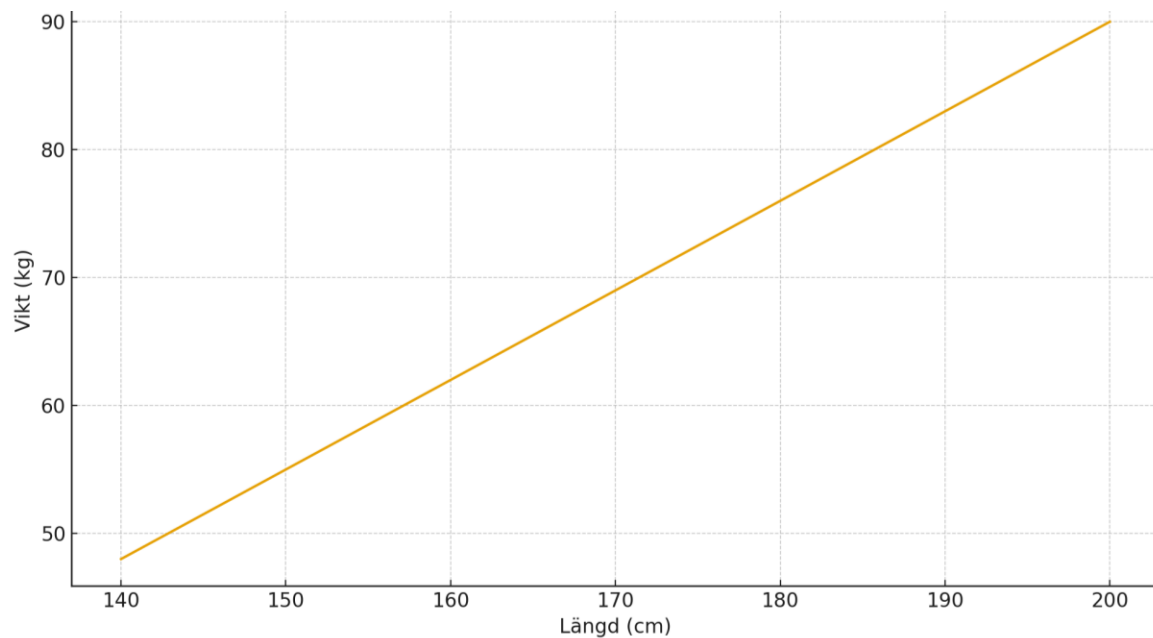
- Välj den modell som är bäst för osedd data
 - Förutsätter att testdata kommer från samma fördelning som träningsdata
 - Bästa modellen har minst andel fel (*error rate*)
- Dela data i tre delar
 - Träningsdata
 - Valideringsdata (*validation set*). Används för att justera och optimera modellen under träning.
 - Testdata

Effektiv användning av träningsdata

- Minimera risk för överträning (overfitting) och underträning (underfitting)
- K-fold korsvalidering
 - Dela data i K lika stora delar
 - En del används som valideringsdata, resten används som träningsdata
 - Upprepa K-gånger där all data används för validering en gång
- Minskar bias genom att använda flera olika valideringsmängder

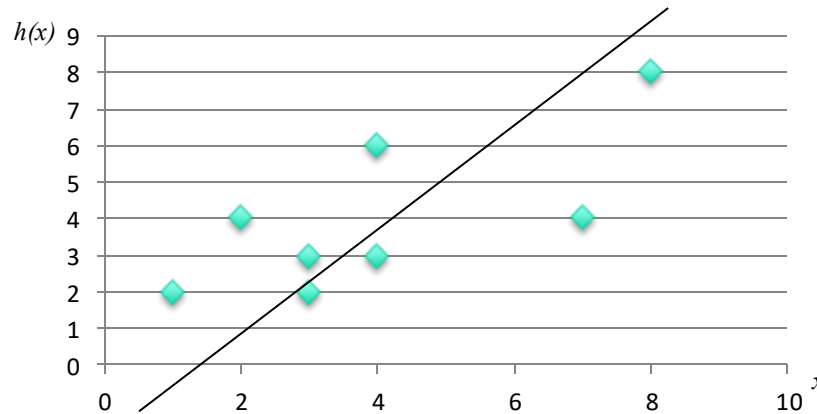
Räta linjens ekvation

$y = kx + m$ där k är lutningen och m förskjutningen från origo



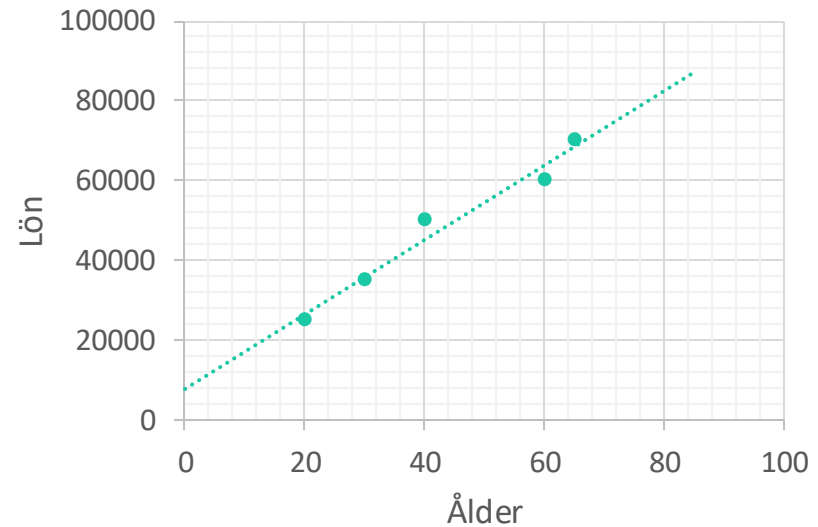
Linjär regression

- Modellantagande: sambandet kan beskrivas med en rät linje
 - Hitta den linje som bäst beskriver en datamängd!
- $h(x) = w_1x + w_0$ där w_1 är lutningen och w_0 förskjutningen från origo
- Jämför avvikelserna mot linjen för varje punkt och minimera det totala felet



Exempel

Ålder	Lön
20	25000
30	35000
40	50000
60	60000
65	70000



Minimera förlust

- Vill inte bara minska felet
 - Ex. värre att klassificera icke-spam som spam än tvärtom
- Vill maximera nyttan, minimera förlusten (*loss*)
 - $L(t, h(x))$ ger differensen i "nyttan" mellan att välja korrekt svar t och det predicerade svaret $h(x)$
Ex. $L(\text{spam}, \text{icke-spam}) = 1$, $L(\text{icke-spam}, \text{spam}) = 10$
- Minimera *loss* över alla par av korrekt värde/predicerat värde

Felfunktioner (*loss functions*)

- Finns flera sätt att räkna ut felet mellan det korrekta värdet t och värdet $h(x)$

- Absolutvärdet, L_1 : $\sum_{i=1}^N |t_i - h(x_i)|$

- Kvadratfelet, L_2 : $\sum_{i=1}^N (t_i - h(x_i))^2$

Vanligast. Stora fel betraktas som värre än små och negativa fel kan inte ta ut positiva.

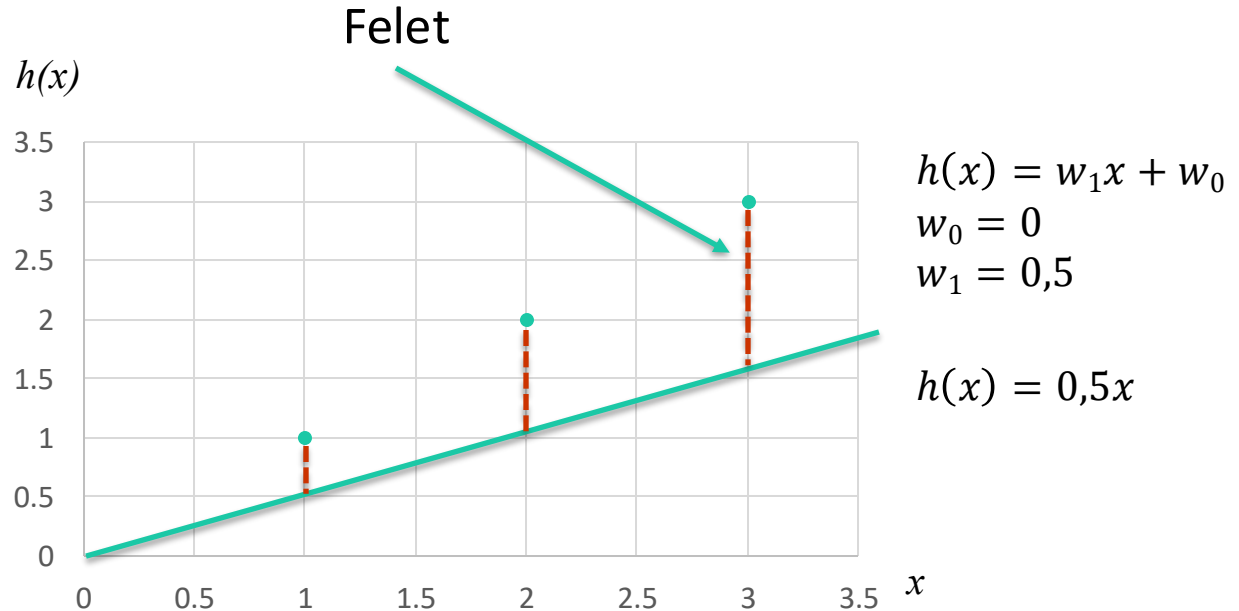
- Värsta fallet: $\max_i |t_i - h(x_i)|$

Förlusten (loss)

$$t_1 = 1$$
$$h(x_1) = 0,5$$

$$t_2 = 2$$
$$h(x_2) = 1$$

$$t_3 = 3$$
$$h(x_3) = 1,5$$



$$\text{Kvadratfelet, } L_2: (1 - 0,5)^2 + (2 - 1)^2 + (3 - 1,5)^2 = 0,25 + 1 + 2,25 = 3,5$$

Felfunktionen för L_2

Kvadratfelet blir större ju mer träningsdata som används, därför används ofta medelkvadratfelet istället:

$$\frac{1}{N} \sum_{i=1}^N (t_i - h(x_i))^2$$

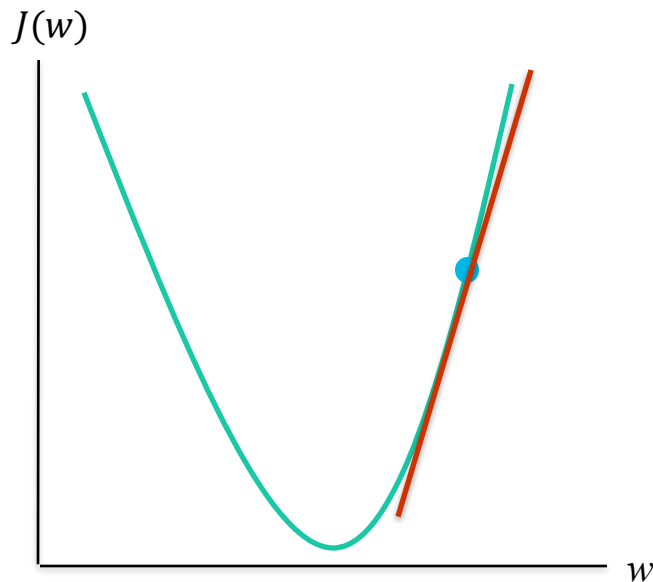
där t_i är korrekt värde och $h(x_i) = w_1 x_i + w_0$

Felfunktionen för L_2

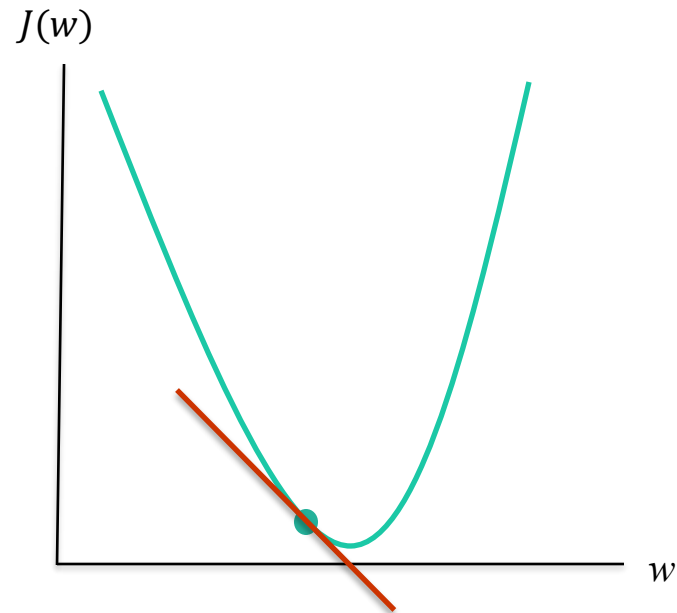
$$J(w_0, w_1) = \frac{1}{N} \sum_{i=1}^N (t_i - h(x_i))^2$$

- Felfunktionen $J(w_0, w_1)$ ger oss ett enkelt sätt att teckna felet för olika värden på parametrarna w_0 och w_1
- Målet är att justera parametrarna så att felet blir så litet som möjligt
- En vanlig metod är gradientsökning

Gradientsökning, intuition



$w = w - \text{stort värde}$



$w = w + \text{litet värde}$

Kurvan beskriver felfunktionen. Lutningen säger hur vi borde justera värdet på w

Gradientsökning

Steg 0: Börja med ett godtyckligt värde för w

Steg 1: Räkna ut felfunktionens lutning (derivatan) i den punkten

Steg 2: Gå i lutningens motsatta riktning:

- Om tangenten har positiv lutning, minska värdet på w

- Om tangenten har negativ lutning, höj värdet på w

- Låt storleken på justeringen vara lutningen multiplicerat med någon inlärningsparameter

Upprepa steg 1 och 2 tills felet blir tillräckligt litet

Derivata, recap

- Derivatan mäter hur en funktion förändras med avseende på en variabel
- För en funktion $f(x)$ definieras derivatan som lutningen vid en viss punkt:

$$f'(x) = \lim_{h \rightarrow 0} \left(\frac{f(x+h) - f(x)}{h} \right)$$

- Om $y = f(x)$ kan vi också skriva $\frac{dy}{dx}$ eller $\frac{df(x)}{dx}$

Förändringen av $f(x)$ med avseende på x i en viss punkt

Men om det finns flera variabler?

- Derivata kan tillämpas om det finns flera variabler givet att man håller vissa variabler konstanta
- Ex. $f(x, y) = x^2 + y^3$
 - Partiella derivatan med avseende på x :
$$\frac{\partial f(x, y)}{\partial x} = 2x$$
 - Partiella derivatan med avseende på y :
$$\frac{\partial f(x, y)}{\partial y} = 3y$$
- Det innebär att vi kan justera lutningen för flera variabler separat!

Gradienten

Justera w_0 och w_1 i gradientens riktning

$$J(w_0, w_1) = \frac{1}{N} \sum_{i=1}^N (t_i - h(x_i))^2 \quad \text{där } h(x_i) = w_1 x_i + w_0$$

$$\frac{\partial J(w_0, w_1)}{\partial w_0} = \frac{1}{N} \sum_{i=1}^N -2(t_i - h(x_i))$$

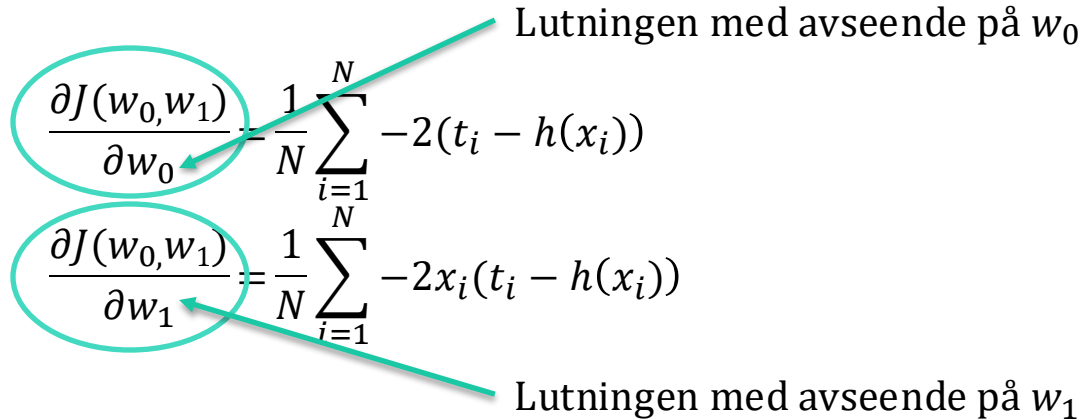
$$\frac{\partial J(w_0, w_1)}{\partial w_1} = \frac{1}{N} \sum_{i=1}^N -2x_i(t_i - h(x_i))$$

OBS: Vi behöver inte veta hur derivatan beräknas, bara att vi kan göra det och hur man utläser det

Gradienten

Justera w_0 och w_1 i gradientens riktning

$$J(w_0, w_1) = \frac{1}{N} \sum_{i=1}^N (t_i - h(x_i))^2 \quad \text{där } h(x_i) = w_1 x_i + w_0$$


$$\frac{\partial J(w_0, w_1)}{\partial w_0} = \frac{1}{N} \sum_{i=1}^N -2(t_i - h(x_i))$$

Lutningen med avseende på w_0

$$\frac{\partial J(w_0, w_1)}{\partial w_1} = \frac{1}{N} \sum_{i=1}^N -2x_i(t_i - h(x_i))$$

Lutningen med avseende på w_1

Parameteruppdatering

Inlärningsparametern η (eta) avgör hur snabbt vi uppdaterar w

$$w_0 = w_0 + \eta \frac{1}{N} \sum_{i=1}^N (t_i - h(x_i))$$

$$w_1 = w_1 + \eta \frac{1}{N} \sum_{i=1}^N x_i (t_i - h(x_i))$$

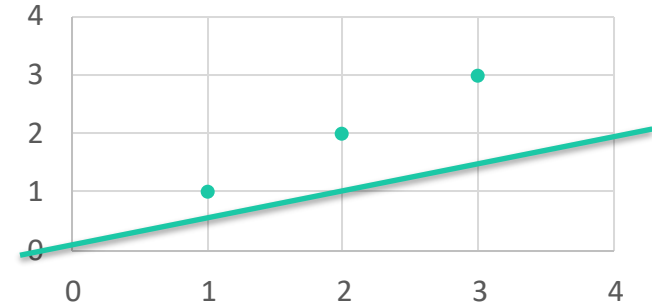
Exempel

$$\eta = 0,1, w_0 = 0$$

$$\text{dvs } h(x) = w_1 x$$

$$\text{Initialt } w_1 = 0,5$$

$$w_1 = w_1 + \eta \frac{1}{N} \sum_{i=1}^N x_i (t_i - h(x_i))$$



Exempel

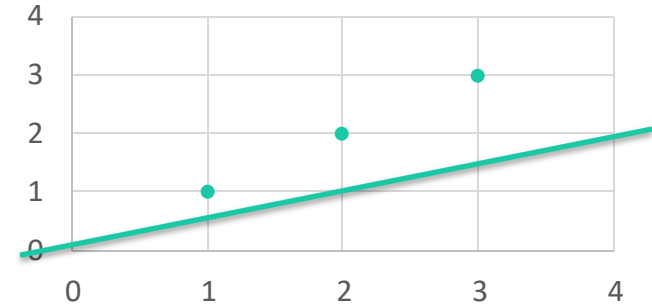
$$\eta = 0,1, w_0 = 0$$

$$\text{dvs } h(x) = w_1 x$$

$$\text{Initialt } w_1 = 0,5$$

$$w_1 = w_1 + \eta \frac{1}{N} \sum_{i=1}^N x_i (t_i - h(x_i))$$

$$w_1 = 0,5 + 0,1 \frac{1}{3} (1(1 - 0,5) + 2(2 - 1) + 3(3 - 1,5)) = 0,733$$



Exempel

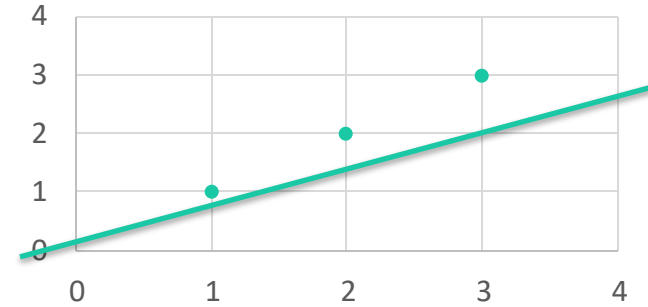
$$\eta = 0,1, w_0 = 0$$

$$\text{dvs } h(x) = w_1 x$$

$$\text{Initialt } w_1 = 0,5$$

$$w_1 = w_1 + \eta \frac{1}{N} \sum_{i=1}^N x_i (t_i - h(x_i))$$

$$w_1 = 0,5 + 0,1 \frac{1}{3} (1(1 - 0,5) + 2(2 - 1) + 3(3 - 1,5)) = 0,733$$



Exempel

$$\eta = 0,1, w_0 = 0$$

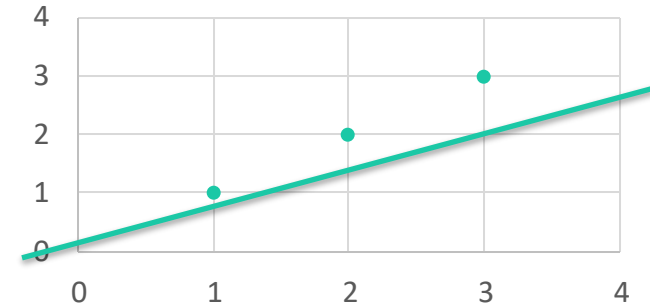
$$\text{dvs } h(x) = w_1 x$$

$$\text{Initialt } w_1 = 0,5$$

$$w_1 = w_1 + \eta \frac{1}{N} \sum_{i=1}^N x_i (t_i - h(x_i))$$

$$w_1 = 0,5 + 0,1 \frac{1}{3} (1(1 - 0,5) + 2(2 - 1) + 3(3 - 1,5)) = 0,733$$

$$w_1 = 0,733 + 0,1 \frac{1}{3} (1(1 - 0,733) + 2(2 - 1,466) + 3(3 - 2,199)) = 0,857$$



Exempel

$$\eta = 0,1, w_0 = 0$$

$$\text{dvs } h(x) = w_1 x$$

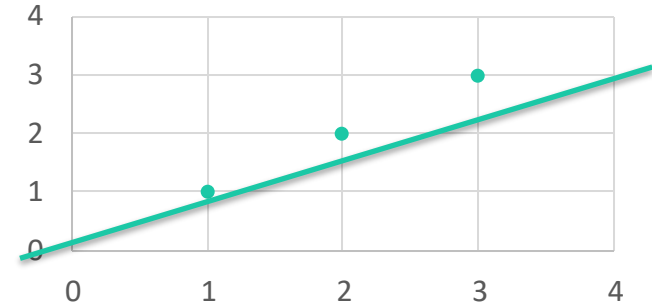
$$\text{Initialt } w_1 = 0,5$$

$$w_1 = w_1 + \eta \frac{1}{N} \sum_{i=1}^N x_i (t_i - h(x_i))$$

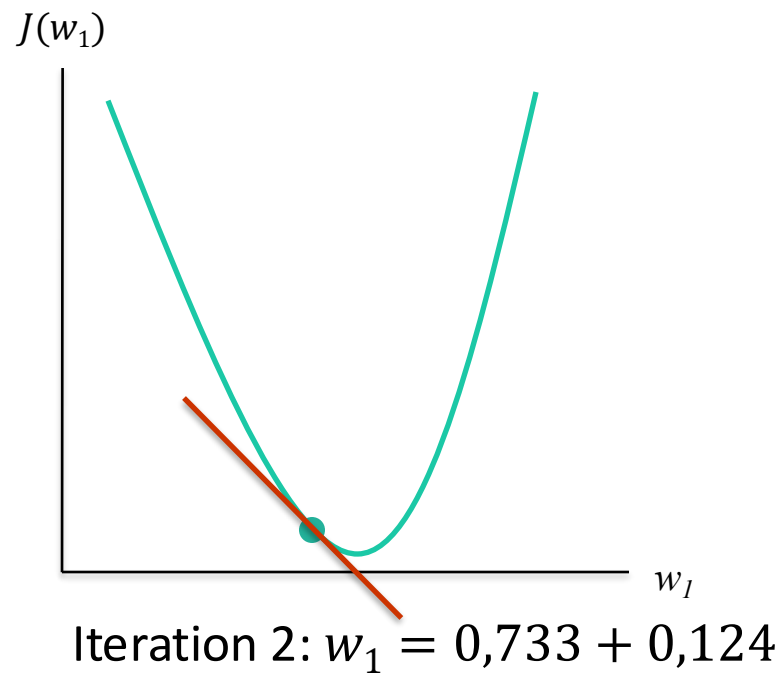
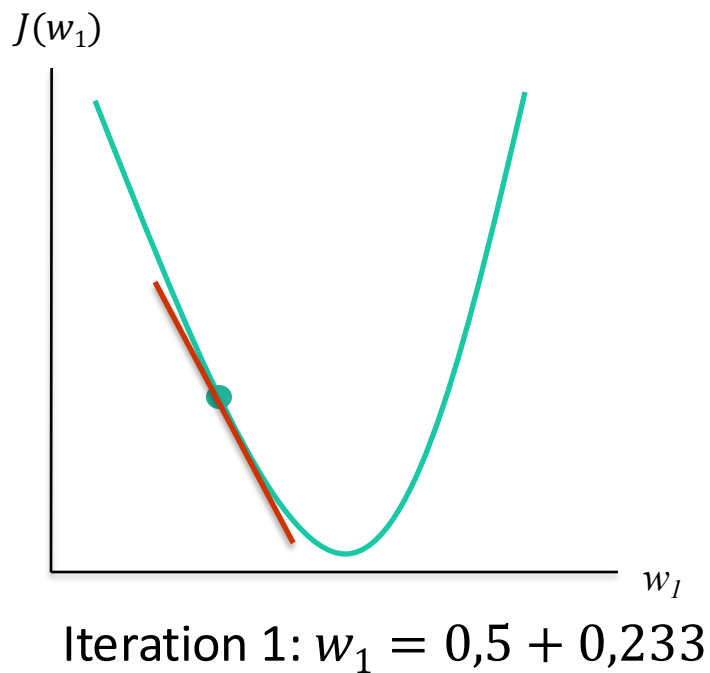
$$w_1 = 0,5 + 0,1 \frac{1}{3} (1(1 - 0,5) + 2(2 - 1) + 3(3 - 1,5)) = 0,733$$

$$w_1 = 0,733 + 0,1 \frac{1}{3} (1(1 - 0,733) + 2(2 - 1,466) + 3(3 - 2,199)) = 0,857$$

...



Gradientsökning



Regularisering

- Används för att undvika overfitting
- Minskar generaliseringsfelet utan att träningsfelet blir mindre
- Exempel: L_2 -regularisering
Lägg till en term till felfunktionen som blir större ju mer komplex funktionen h är

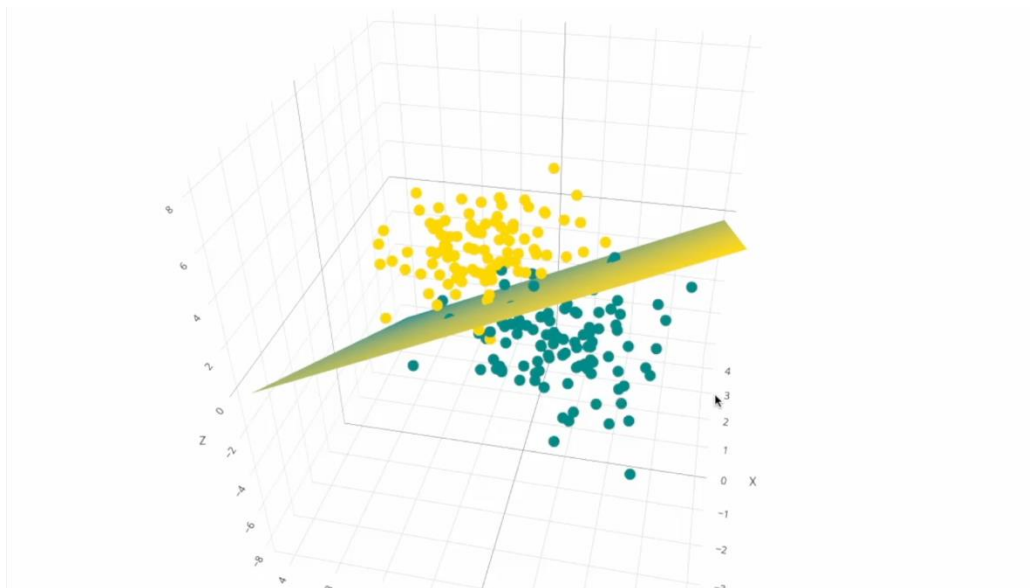
$$\sum_{i=1}^N (t_i - h(x_i))^2 + \lambda \text{Complexity}(h)$$

λ är en så kallad *hyperparameter*

En hyperparameter styr inläringen och väljs innan träningen börjar.
 η (inlärningsparametern) räknas också som en hyperparameter.

Multipel linjär regression

Fler särdrag, kraftfullare modell



Multipel linjär regression

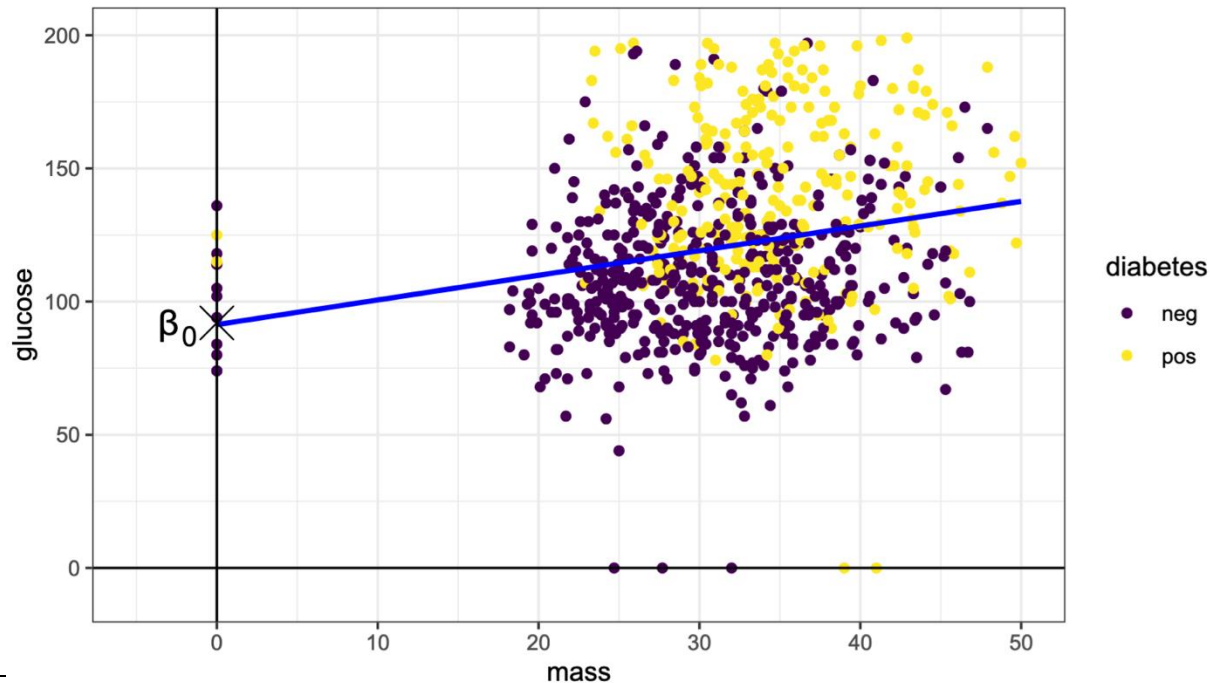
- Linjär regression: $h(x) = w_1x + w_0$
- Multipel linjär regression: $h(x) = w_1x_1 + w_2x_2 + \dots + w_nx_n + w_0$

$$h(x_1, \dots, x_n) = \sum_{i=0}^n w_i x_i$$

- Stödjer nu flera särdrag x och parametrar w
- Sambandet mellan indata och utdata blir nu ett *hyperplan*, dvs. yta som delar ett n -dimensionellt rum

Exempel: Hyperplan

Glukosnivå som en funktion av vikt, svagt positiv lutning



Vektorer

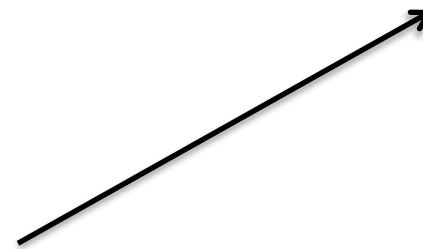
- När antalet variabler växer behöver vi förenkla notationen
- Särdrag och parametrar kan skrivas som vektorer, $\mathbf{x}$ och $\mathbf{w}$

$$h(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$$

jämför med

$$h(x_1, \dots, x_n) = \sum_{i=0}^n w_i x_i$$

- Skalär
 - Ett tal, ex. längd (2m), tid (12:30), temperatur (170K)
- Vektor
 - Riktning och längd, t.ex. hastighet, kraft

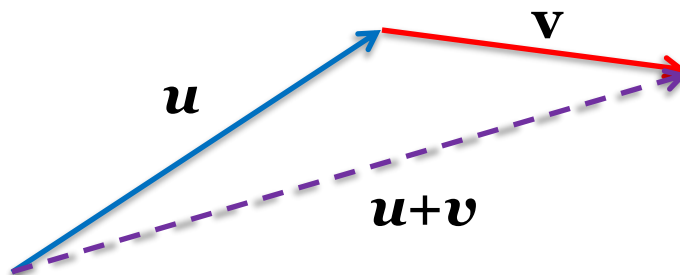


Några vektorsamband

- Likhet
 - Längd och riktning samma riktning men kan ha olika startpunkt



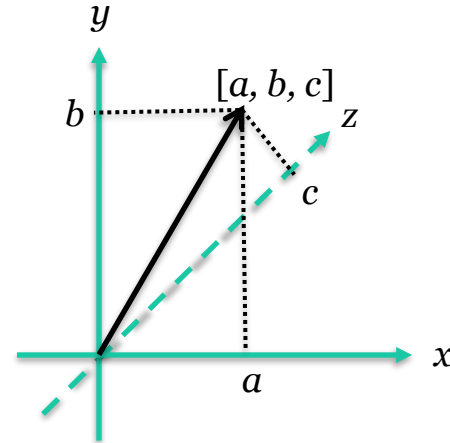
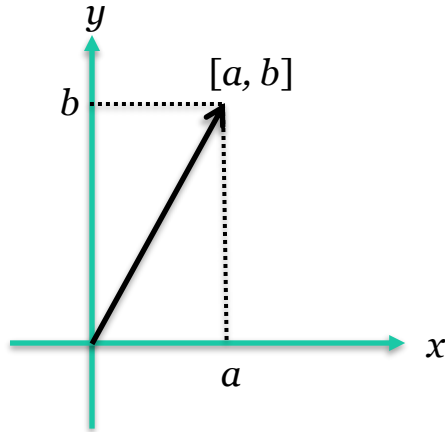
- Addition



- Multiplikation



Punkter och vektorer



Vektorer kan identifieras som punkter i ett koordinatsystem:

$[a, b]$ och $[a, b, c]$

... men kan generaliseras till vilken dimension som helst: $[a_0, a_1, \dots, a_n]$

Vektoroperationer

$$\mathbf{z} = [a_1, b_1, c_1]$$

$$\mathbf{w} = [a_2, b_2, c_2]$$

Addition

$$\mathbf{z} + \mathbf{w} = [a_1 + a_2, b_1 + b_2, c_1 + c_2]$$

Multiplikation med skalär

$$2\mathbf{z} = [2a_1, 2b_1, 2c_1]$$

Skalärprodukt

Produkten av två vektorer ger en skalär (ett tal)

$$\mathbf{z} = [a_1, b_1, c_1]$$

$$\mathbf{w} = [a_2, b_2, c_2]$$

$$\mathbf{z} \cdot \mathbf{w} = a_1a_2 + b_1b_2 + c_1c_2$$

Matriser

- Matriser är rektangulära tabeller med tal
- Smidig representation om man har mycket träningsdata och många parametrar
- Exempel $A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}$ och $B = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix}$
- A är en matris med 2 rader och 3 kolumner:
 A är en 2×3 -matris
 B är en 3×2 -matris
- Matriser multipliceras annorlunda

Matrismultiplikation

- Matrisen $C = AB$ fås genom att multiplicera rader i A med kolumner i B
- Antalet kolumner i A **måste** matcha antalet rader i B

- Exempel $A = \begin{bmatrix} 1 & 3 & 4 \\ -3 & 2 & 0 \end{bmatrix}$ och $B = \begin{bmatrix} 2 & 1 \\ 3 & 0 \\ -1 & 1 \end{bmatrix}$

$$C = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 3 + 4 \cdot -1 & 1 \cdot 1 + 3 \cdot 0 + 4 \cdot 1 \\ -3 \cdot 2 + 2 \cdot 3 + 0 \cdot -1 & -3 \cdot 1 + 2 \cdot 0 + 0 \cdot 1 \end{bmatrix} = \begin{bmatrix} 7 & 5 \\ 0 & -3 \end{bmatrix}$$

Matrismultiplikation

- Matrisen $C = AB$ fås genom att multiplicera rader i A med kolumner i B
- Antalet kolumner i A **måste** matcha antalet rader i B

- Exempel $A = \begin{bmatrix} 1 & 3 & 4 \\ -3 & 2 & 0 \end{bmatrix}$ och $B = \begin{bmatrix} 2 & 1 \\ 3 & 0 \\ -1 & 1 \end{bmatrix}$

$$C = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 3 + 4 \cdot -1 & 1 \cdot 1 + 3 \cdot 0 + 4 \cdot 1 \\ -3 \cdot 2 + 2 \cdot 3 + 0 \cdot -1 & -3 \cdot 1 + 2 \cdot 0 + 0 \cdot 1 \end{bmatrix} = \begin{bmatrix} 7 & 5 \\ 0 & -3 \end{bmatrix}$$

Matrismultiplikation

- Matrisen $C = AB$ fås genom att multiplicera rader i A med kolumner i B
- Antalet kolumner i A **måste** matcha antalet rader i B

- Exempel $A = \begin{bmatrix} 1 & 3 & 4 \\ -3 & 2 & 0 \end{bmatrix}$ och $B = \begin{bmatrix} 2 & 1 \\ 3 & 0 \\ -1 & 1 \end{bmatrix}$

$$C = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 3 + 4 \cdot -1 & 1 \cdot 1 + 3 \cdot 0 + 4 \cdot 1 \\ -3 \cdot 2 + 2 \cdot 3 + 0 \cdot -1 & -3 \cdot 1 + 2 \cdot 0 + 0 \cdot 1 \end{bmatrix} = \begin{bmatrix} 7 & 5 \\ 0 & -3 \end{bmatrix}$$

Matrismultiplikation

- Matrisen $C = AB$ fås genom att multiplicera rader i A med kolumner i B
- Antalet kolumner i A **måste** matcha antalet rader i B

- Exempel $A = \begin{bmatrix} 1 & 3 & 4 \\ -3 & 2 & 0 \end{bmatrix}$ och $B = \begin{bmatrix} 2 & 1 \\ 3 & 0 \\ -1 & 1 \end{bmatrix}$

$$C = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 3 + 4 \cdot -1 & 1 \cdot 1 + 3 \cdot 0 + 4 \cdot 1 \\ -3 \cdot 2 + 2 \cdot 3 + 0 \cdot -1 & -3 \cdot 1 + 2 \cdot 0 + 0 \cdot 1 \end{bmatrix} = \begin{bmatrix} 7 & 5 \\ 0 & -3 \end{bmatrix}$$

Matrismultiplikation

- Matrisen $C = AB$ fås genom att multiplicera rader i A med kolumner i B
- Antalet kolumner i A **måste** matcha antalet rader i B

- Exempel $A = \begin{bmatrix} 1 & 3 & 4 \\ -3 & 2 & 0 \end{bmatrix}$ och $B = \begin{bmatrix} 2 & 1 \\ 3 & 0 \\ -1 & 1 \end{bmatrix}$

$$C = \begin{bmatrix} 1 \cdot 2 + 3 \cdot 3 + 4 \cdot -1 & 1 \cdot 1 + 3 \cdot 0 + 4 \cdot 1 \\ -3 \cdot 2 + 2 \cdot 3 + 0 \cdot -1 & -3 \cdot 1 + 2 \cdot 0 + 0 \cdot 1 \end{bmatrix} = \begin{bmatrix} 7 & 5 \\ 0 & -3 \end{bmatrix}$$

Matrismultiplikation, 2

- Eftersom kolumner och rader måste matcha i antal behöver vi ibland *transponera* matriser (konvertera rader till kolumner)

- Exempel $A = \begin{bmatrix} 1 & 3 & 4 \\ -3 & 2 & 0 \end{bmatrix}$ $A^T = \begin{bmatrix} 1 & -3 \\ 3 & 2 \\ 4 & 0 \end{bmatrix}$

- $\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}$ $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$

- $h(\mathbf{x}) = \mathbf{w}^T \mathbf{x} = \begin{bmatrix} w_1 & w_2 & w_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = [w_1 x_1 + w_2 x_2 + w_3 x_3]$

Multipl linjär regression

- Multipl linjär regression: $h(x) = w_1x_1 + w_1x_1 + \dots + w_nx_n + w_0$

$$h(x_1, \dots, x_n) = \sum_{i=0}^n w_i x_i$$

kan nu formuleras i termer av vektorer/matriser

$$h(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$$

där $\mathbf{x}$ är en vektor med inputvärden och $\mathbf{w}$ är en vektor med vikter

För att kunna utföra multiplikationen så måste vi transponera $\mathbf{w}$

Multipel linjär regression

- Vad händer med gradienten/lutningen i flera dimensioner?
 - En vektor med alla lutningar per parameter som ska uppdateras, alltså en för alla $w_1, w_2, \dots, w_n$
- Beräkna för varje parameter separat

Sammanfattning

- Linjär regression
 - Matcha en rät linje mot datan
 - Beräkna felet
 - Uppdatera parametrar för att minska felet
- Multipel linjär regression
 - Fler än en variabel
 - Beräkna alla fel
 - Uppdatera parameterarna för att minska felet
- Vektor- och matrisoperationer