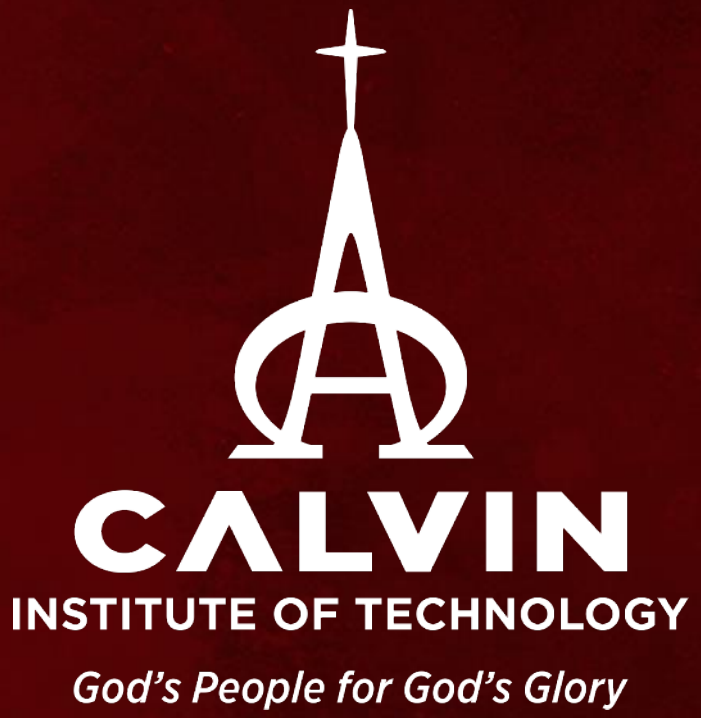


Regression

Hendrik Santoso Sugiarto

IBDA2032 – *Artificial Intelligence*

Capaian Pembelajaran

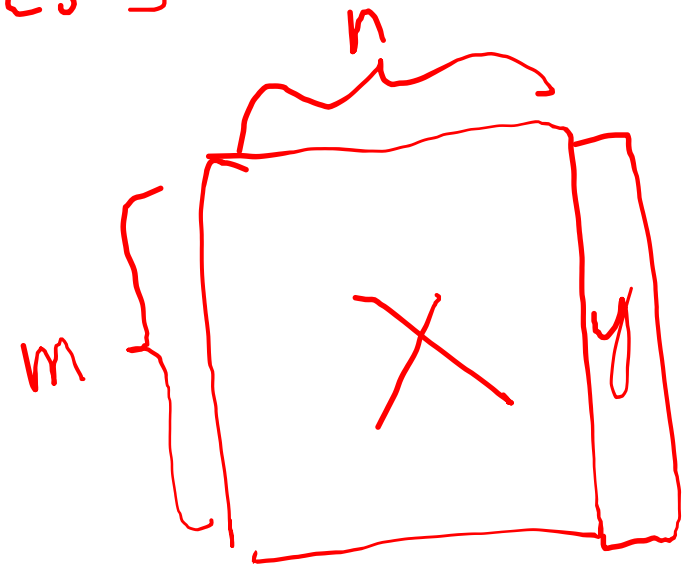
- Konsep Regresi
- Regresi linear univariat
- Optimisasi
- Regresi linear multivariat
- Regresi nonlinear
- Regresi logistic

CALVIN INSTITUTE OF TECHNOLOGY

Regresi

Supervised Learning

- Bentuk formal:
  - Input: $\boldsymbol{x} \in \mathcal{X} \in \mathbb{R}^n$
  - Output: $y \in \mathcal{Y} \in \begin{cases} \mathbb{R} \rightarrow regresi \\ \{+1, -1\} \rightarrow klasifikasi\ biner \\ \{1,2, \dots, K\} \rightarrow klasifikasi\ multikelas \end{cases}$
  - Target function: $f: \mathcal{X} \rightarrow \mathcal{Y}\ (unknown)$
  - Training data: $D = \{(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \dots, (\boldsymbol{x}^{(m)}, y^{(m)})\}$
  - Hypothesis: $h: \mathcal{X} \rightarrow \mathcal{Y}$
  - Hypothesis space: $h \in \mathcal{H}$

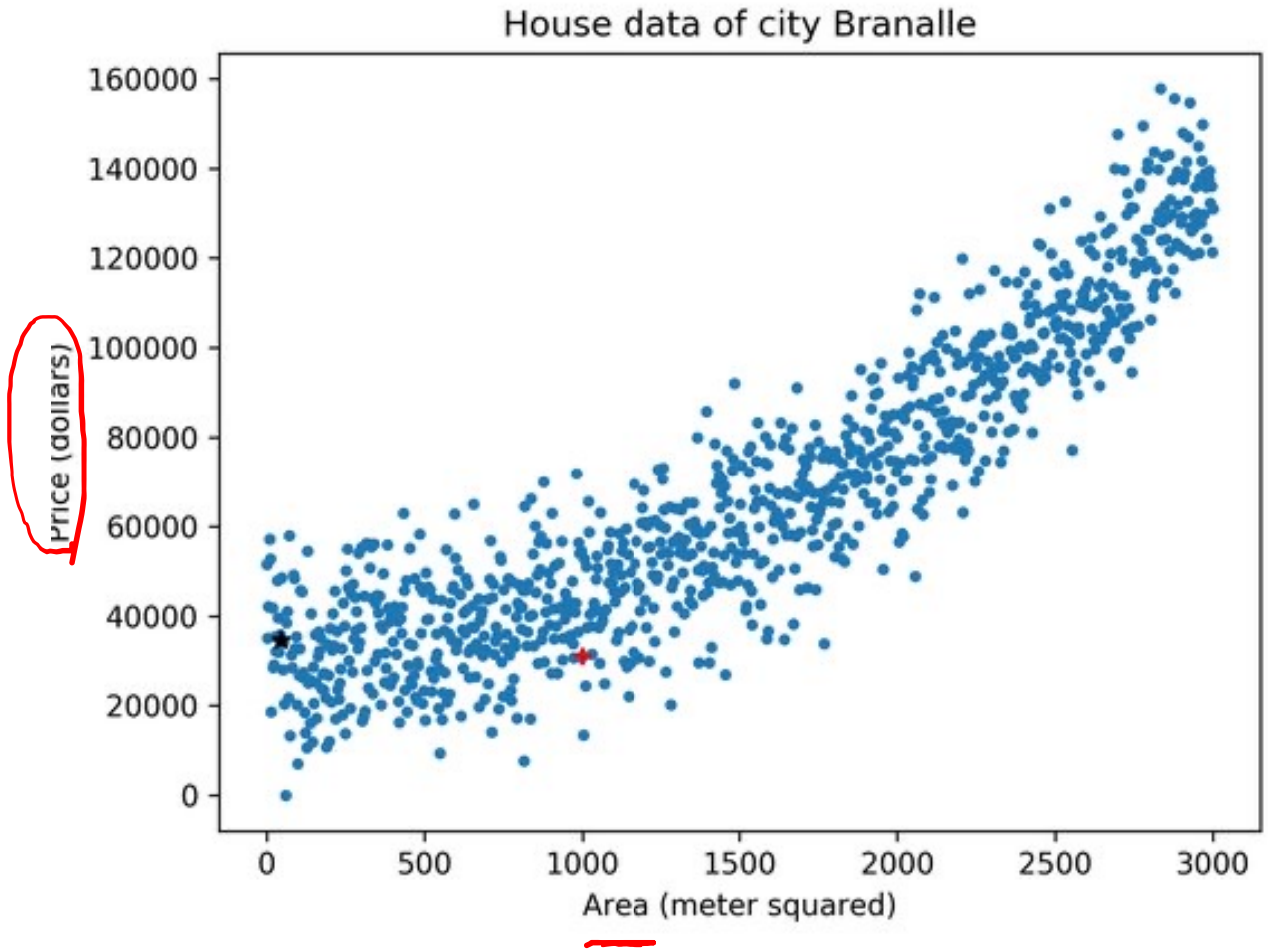


Problem Regresi

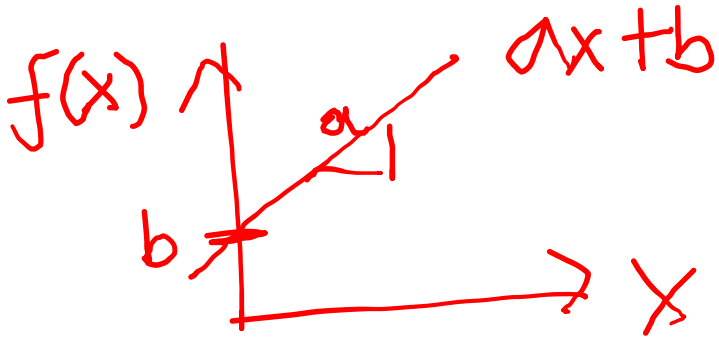
- Input → unknown function → output
- $\boldsymbol{x} \rightarrow h(\boldsymbol{x}) \approx f(\boldsymbol{x}) \rightarrow y = f(\boldsymbol{x})$

$2x+1$

x^2+3x+1



Hypothesis Space Univariat



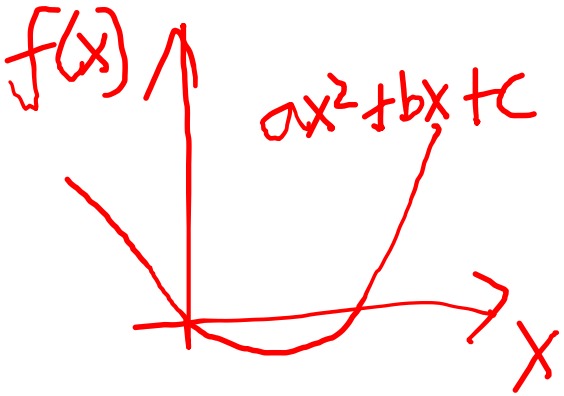
- Model Linear:

$$h(x) = ax + b \approx f(x)$$

  - Kemungkinan hipotesis tak hingga
  - Setiap pilihan koefisien a dan b menghasilkan 1 kemungkinan hipotesis

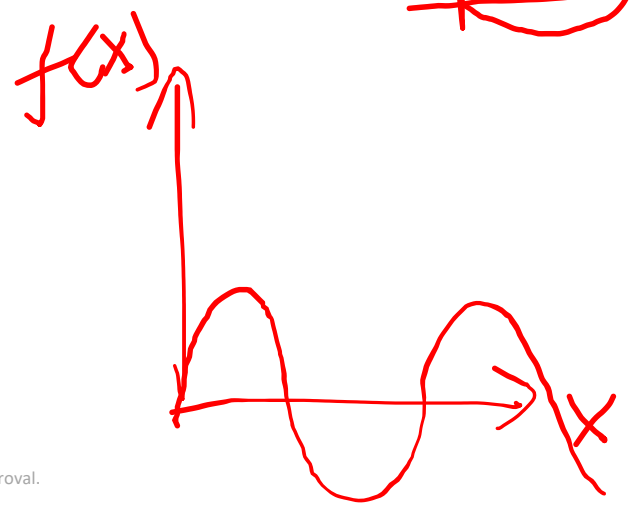
- Model Polynomial:

$$h(x) = ax^2 + bx + c \approx f(x)$$
$$h(x) = \theta_n x^n + \cdots + \theta_1 x + \theta_0 \approx f(x)$$

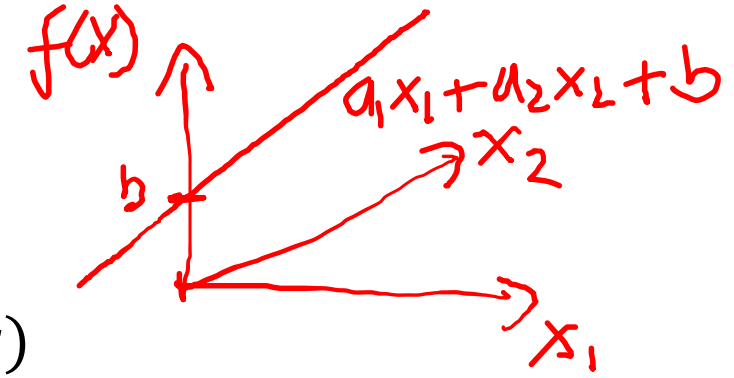


- Model Nonlinear apapun:

$$h(x) = g(x) \approx f(x)$$



Hypothesis Space Multivariat



- Model Linear:

$$h(x) = a_1x_1 + \cdots + a_mx_m + b \approx f(x)$$

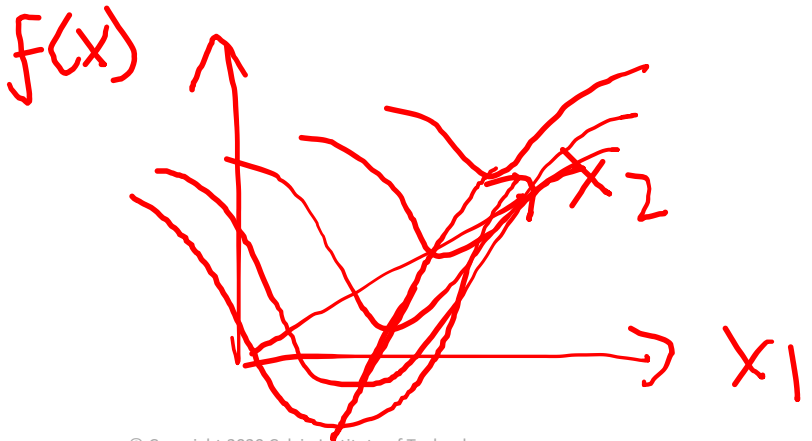
  - Kemungkinan hipotesis tak hingga
  - Setiap pilihan koefisien a dan b menghasilkan 1 kemungkinan hipotesis

- Model Polynomial:

$$h(x) = a_1x_1^2 + \cdots + a_mx_m^2 + b_1x_1 + \cdots + b_mx_m + c \approx f(x)$$

- Model Nonlinear apapun:

$$h(\boldsymbol{x}) = g(\boldsymbol{x}) \approx f(\boldsymbol{x})$$



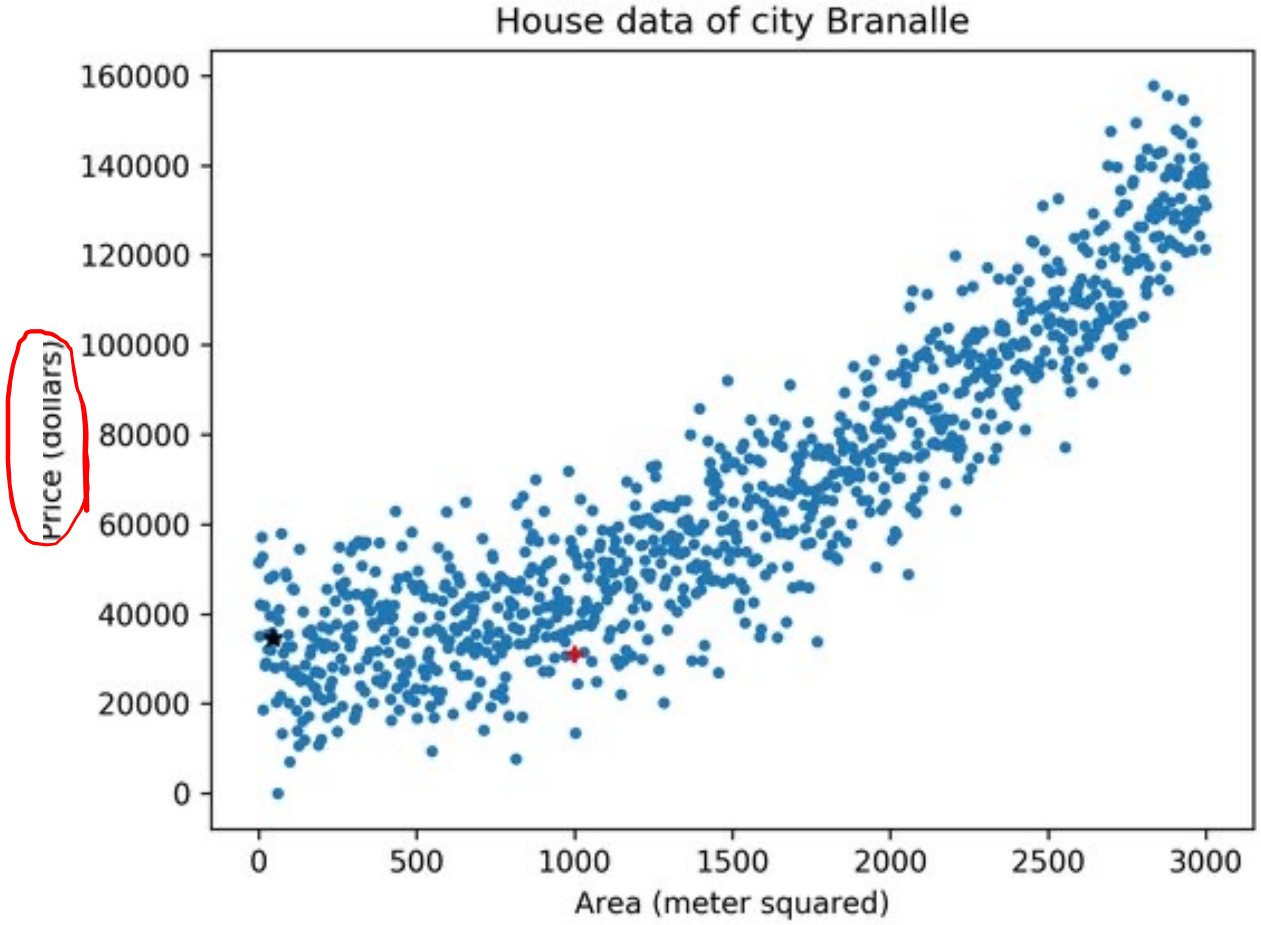
CALVIN INSTITUTE OF TECHNOLOGY

Regresi Linear Univariat

Regresi dengan 1 variabel

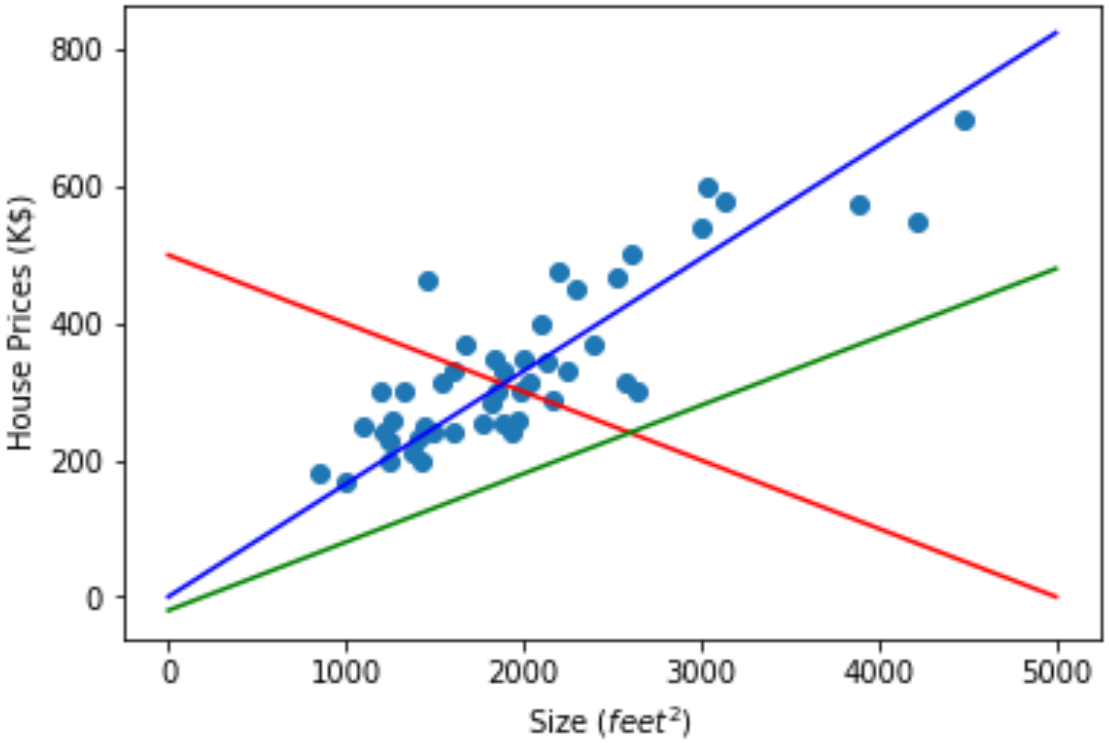
- Training set house price

Area (x)	Price (y)
1535	35500
2107	61500
858	25000
1515	45000
525	20000
2558	80500
1899	50000
...	...



Representasi linear univariat

- Bagaimana representasi hipotesis h?
- Dimana $h(x) = \theta_0 + \theta_1 x$
- Bagaimana memilih parameter θ_0, θ_1?



Cost Function

fungsi harga

- Hypothesis:

$$h(x) = \theta_0 + \theta_1 x$$

index bawah

- Parameters:

$$\theta_0, \theta_1$$

- Cost function: mean squared error (MSE)

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h\left(x^{(i)}\right) - y^{(i)} \right)^2$$

error

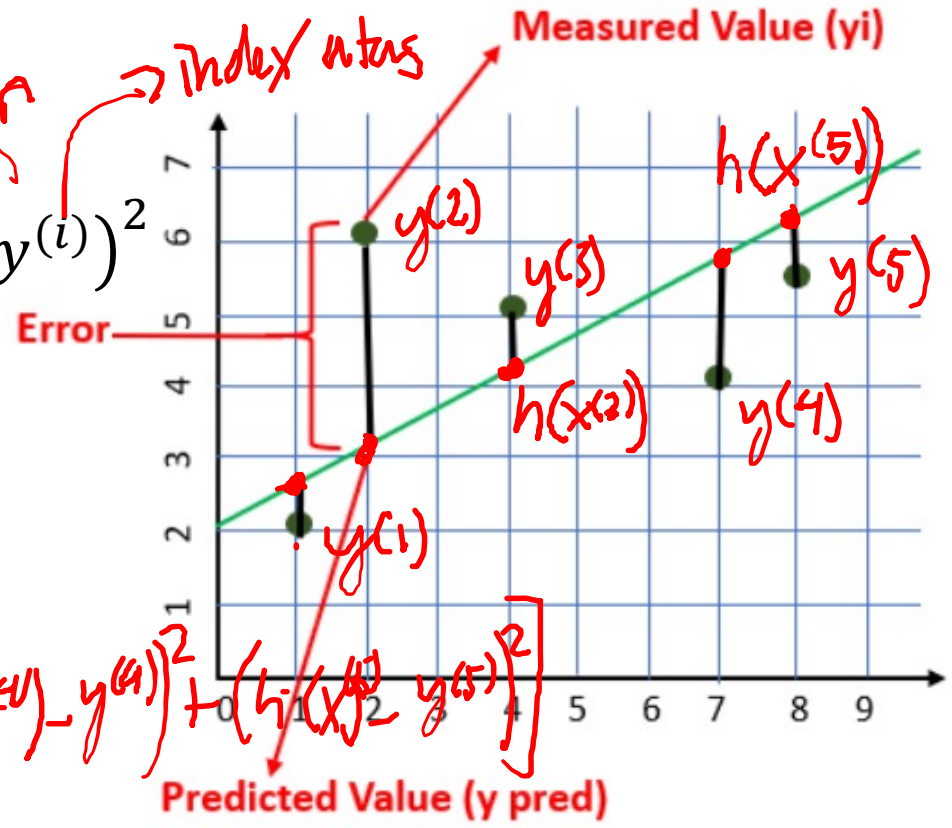
index atas

- Goal:

$$\min_{\theta_0, \theta_1} J(\theta_0, \theta_1)$$

i = 1 i = 2 i = 3

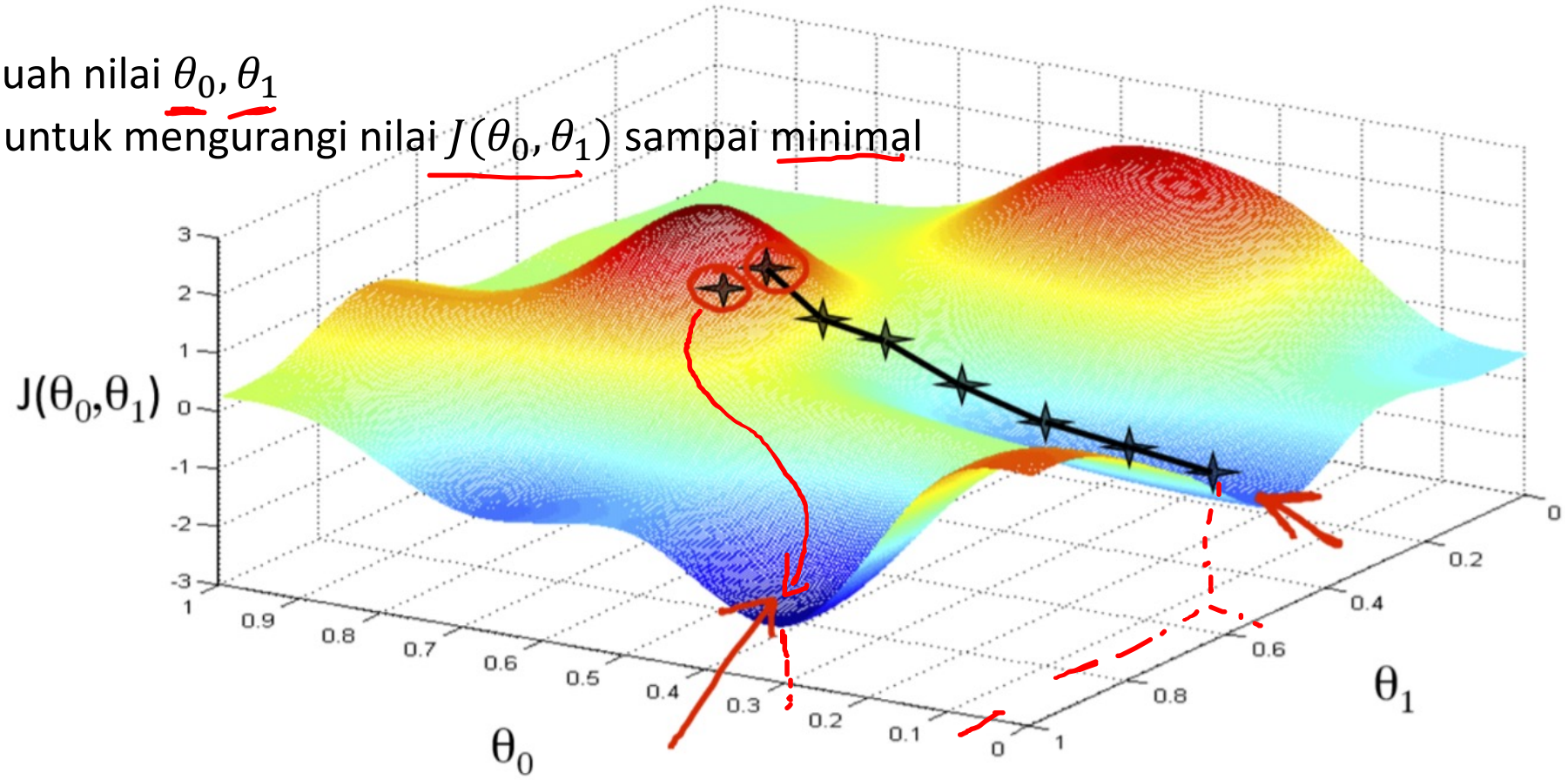
$$\frac{1}{2 \times 5} \left[\left(h(x^{(1)}) - y^{(1)} \right)^2 + \left(h(x^{(2)}) - y^{(2)} \right)^2 + \left(h(x^{(3)}) - y^{(3)} \right)^2 + \left(h(x^{(4)}) - y^{(4)} \right)^2 + \left(h(x^{(5)}) - y^{(5)} \right)^2 \right]$$



CALVIN INSTITUTE OF TECHNOLOGY

Optimisasi cost function

- Terdapat suatu objective function $J(\theta_0, \theta_1)$
- Ingin optimisasi $\min_{\theta_0,\theta_1} J(\theta_0, \theta_1)$
- Outline:
  - Mulai dengan sebuah nilai θ_0, θ_1
  - Terus ubah θ_0, θ_1 untuk mengurangi nilai $J(\theta_0, \theta_1)$ sampai minimal



Algoritma Gradient Descent

- Inisiasi θ_0, θ_1
- Ulangi sampai konvergen:
  - update secara simultan

$$J(\theta_0,\theta_1) = \frac{1}{2m}\sum_{i=1}^{m}\left(h(x^{(i)}) - y^{(i)}\right)^2$$

$$h(x^{(i)}) \rightarrow \theta_0 + \theta_1 x^{(i)}$$

$$\theta_0 := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$

$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$

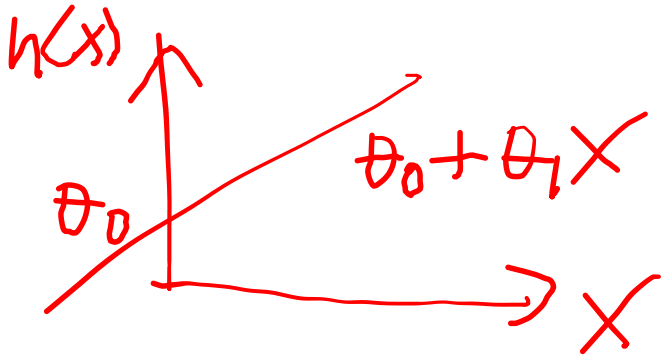
- Dimana α adalah learning rate (hyperparameter)

$$\frac{\partial J}{\partial \theta_0} = \frac{\partial J}{\partial \Sigma()} \frac{\partial \Sigma()}{\partial ()} \frac{\partial ()}{\partial \theta_0} = \frac{1}{2m}\sum_{i=1}^{m} 2\left(h(x^{(i)}) - y^{(i)}\right)$$

$$\frac{\partial J}{\partial \theta_1} = \frac{\partial ()}{\partial \theta_1} = \frac{1}{2m}\sum_{i=1}^{m} 2\left(h(x^{(i)}) - y^{(i)}\right)x^{(i)}$$

CALVIN INSTITUTE OF TECHNOLOGY

Gradient descent untuk regresi linear



- Model regresi linear:

$$h(x) = \theta_0 + \theta_1 x$$

- Cost function:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h\left(x^{(i)}\right) - y^{(i)}\right)^2$$

- Turunan parsial:

$$\frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = \frac{\partial}{\partial \theta_0} \frac{1}{2m} \sum_{i=1}^{m} \left(\theta_0 + \theta_1 x^{(i)} - y^{(i)}\right)^2 = \frac{1}{m} \sum_{i=1}^{m} \left(h\left(x^{(i)}\right) - y^{(i)}\right)$$

$$\frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \frac{\partial}{\partial \theta_1} \frac{1}{2m} \sum_{i=1}^{m} \left(\theta_0 + \theta_1 x^{(i)} - y^{(i)}\right)^2 = \frac{1}{m} \sum_{i=1}^{m} \left(h\left(x^{(i)}\right) - y^{(i)}\right) x^{(i)}$$

- Gradient descent:

$$\theta_0 := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h\left(x^{(i)}\right) - y^{(i)}\right)$$

$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h\left(x^{(i)}\right) - y^{(i)}\right) x^{(i)}$$

0.001

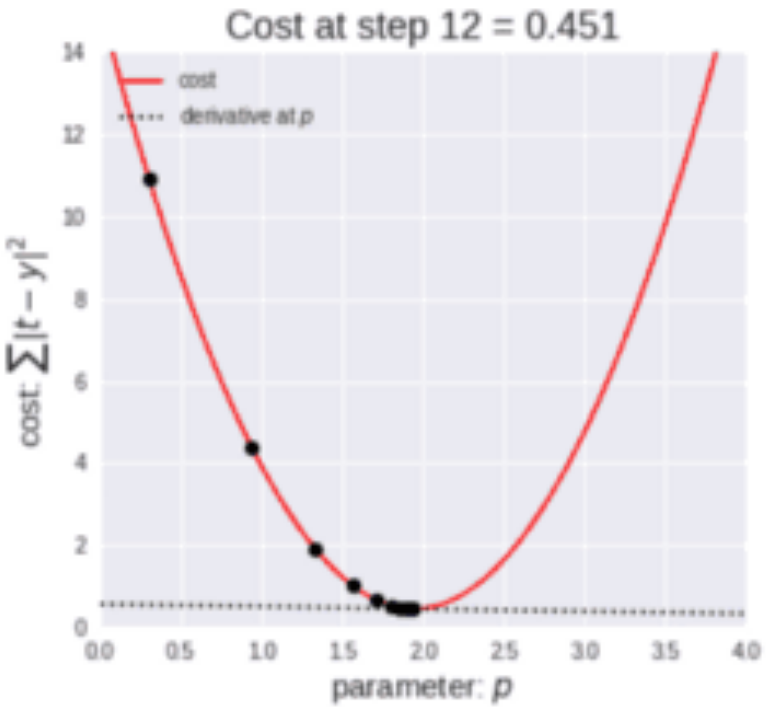
CALVIN INSTITUTE OF TECHNOLOGY

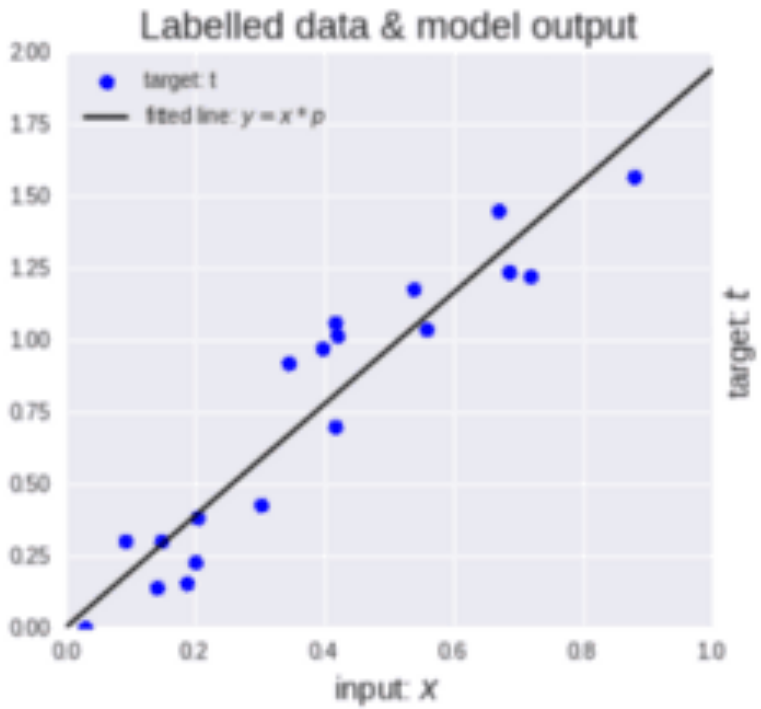
Gradient descent untuk regresi linear

- Semakin cost function bergerak menuju minimum, semakin fit model terhadap data latih

$J(\theta_0, \theta_1)$

$h(x) = \theta_0 + \theta_1 x$



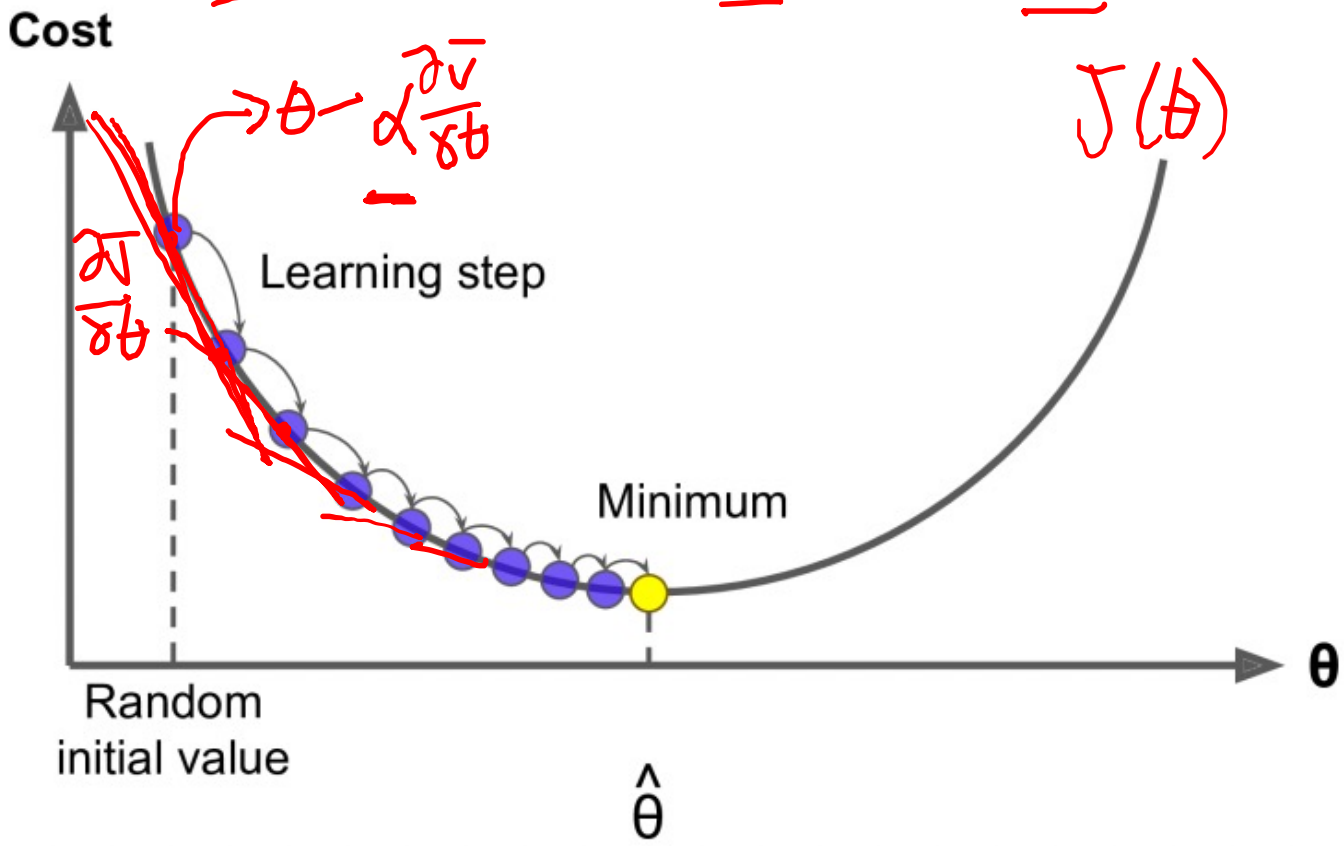


Optimisasi

Penjelasan gradient descent

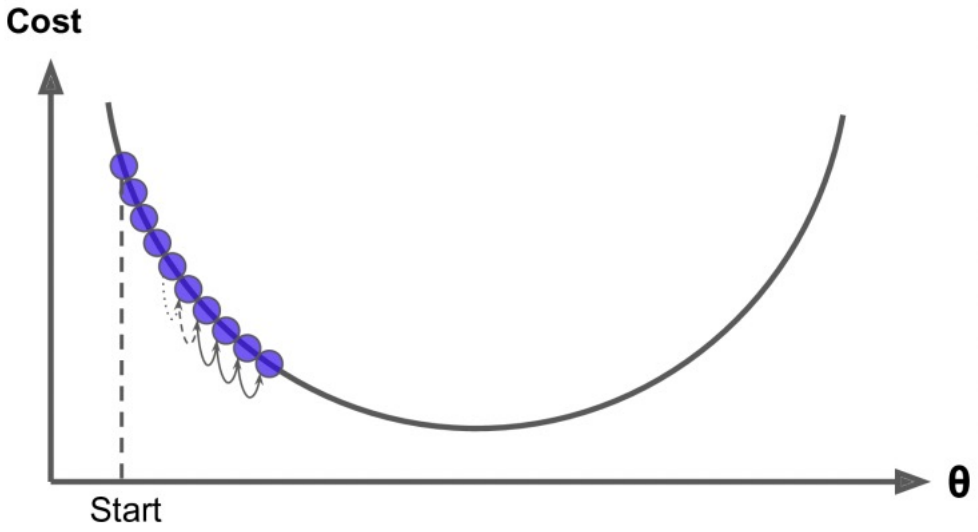
- Gradient descent didasarkan pada observasi jika fungsi $J(\theta)$ adalah defined dan differentiable di titik θ dan sekitarnya, maka J akan berkurang paling cepat jika bergerak dari posisi θ menuju arah negatif dari gradien (kemiringan) J terhadap θ

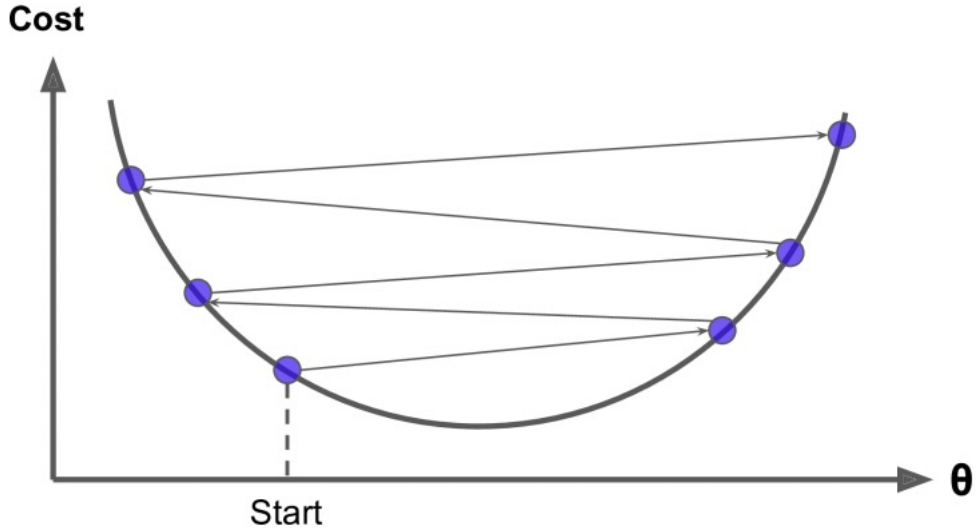
$$\frac{\partial J}{\partial \theta}$$



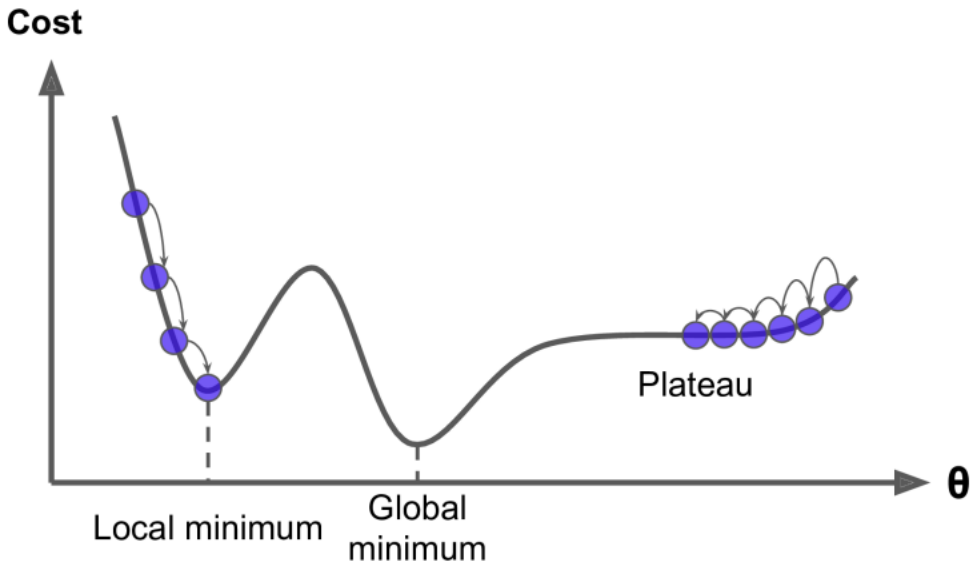
Learning rate

- Terlalu kecil: pembelajaran lama
- Terlalu besar: divergen



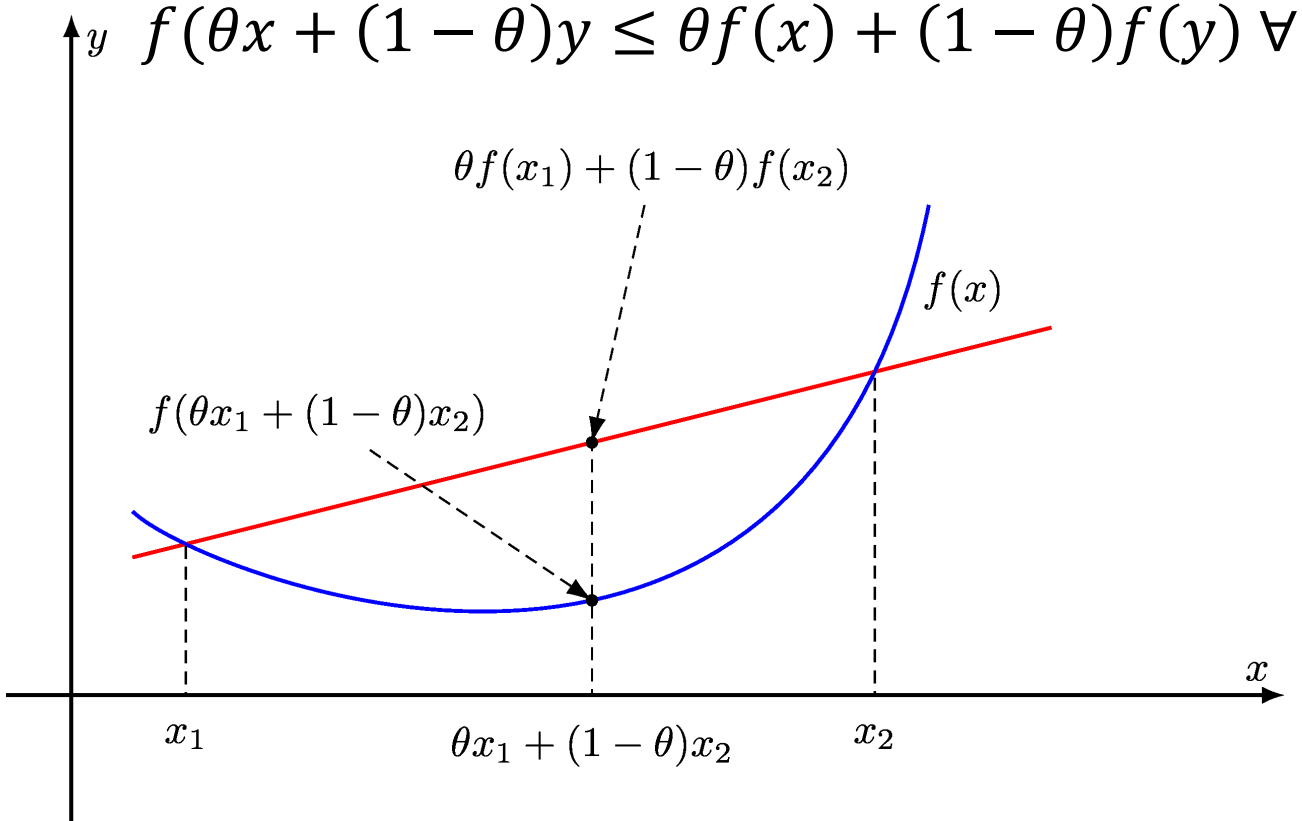


Kelemahan gradient descent



- Tidak bisa optimisasi fungsi non-convex
- Negatif dari fungsi convex adalah fungsi concave
- Convex→setiap minima lokal adalah minima global
- Suatu fungsi f disebut convex jika:

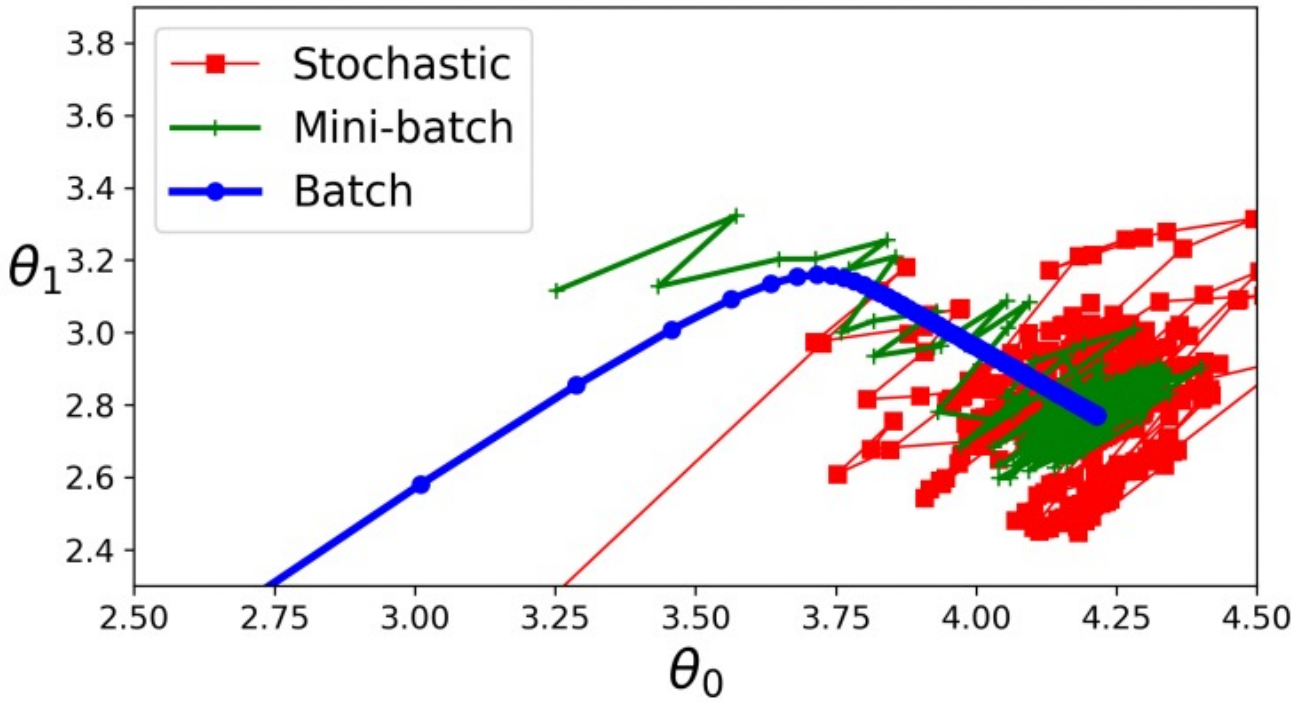
$$f(\theta x + (1-\theta)y \leq \theta f(x) + (1-\theta)f(y)\ \forall 0 \leq \theta \leq 1$$



$$\theta - \alpha \frac{\partial J}{\partial \theta} \downarrow 0$$

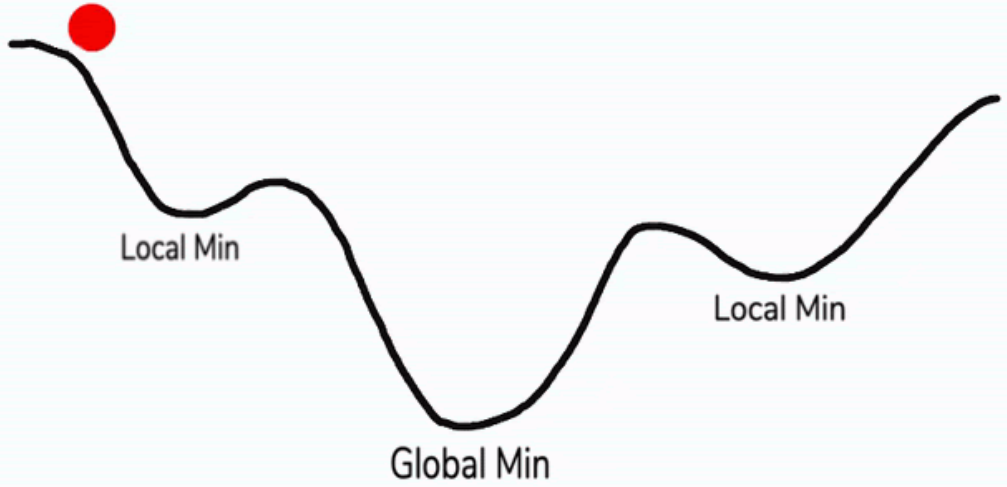
Batch vs stochastic

- Batch memakai semua data latih
- Mini-batch memakai sebagian data latih
- Stochastic memakai 1 data latih



Optimisasi

- Proses belajar terjadi melalui proses optimisasi parameter (θ) terhadap fungsi objektif (J) → model diharapkan menjadi semakin cerdas seiring dengan semakin optimalnya parameter tersebut
- Makin rumit sebuah model, makin sulit proses optimisasinya



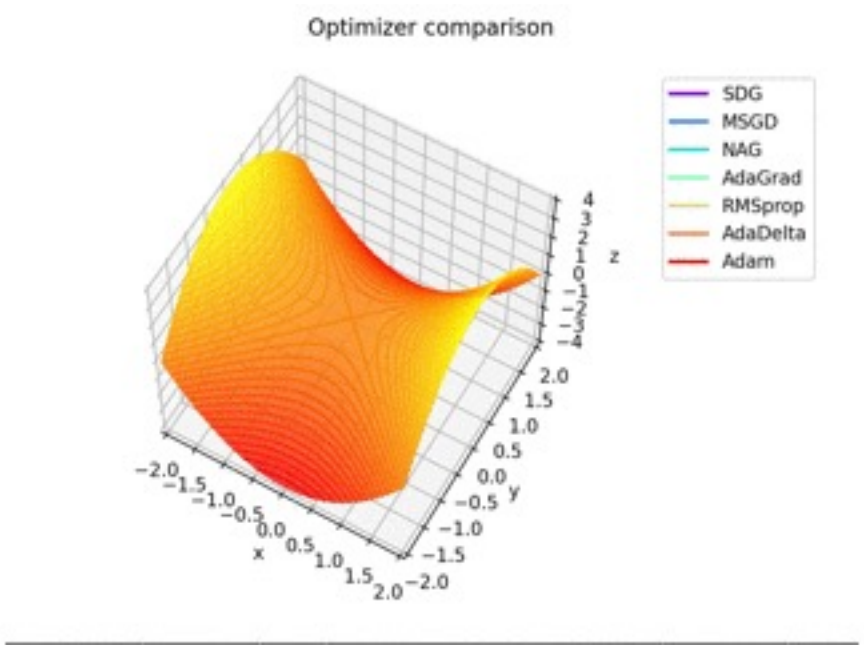
Optimisasi 1st order

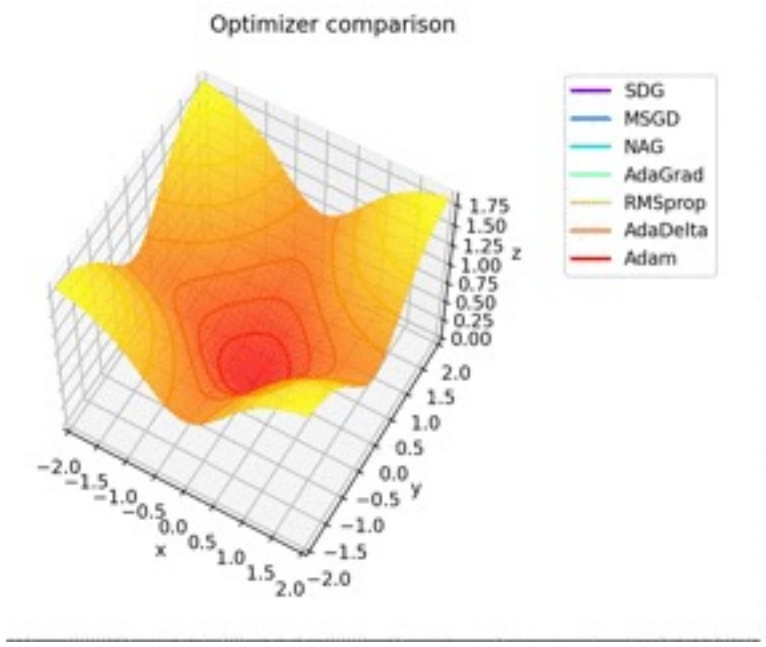
- Gradient Descents:
  - $\theta := \theta - \alpha \nabla_\theta J$
  - Hyperparameter krusial: α (learning rate)
- Momentum (analogi fisika):
  - $g = \nabla_\theta J(\theta) = \frac{\partial v}{\partial t}$
  - $v = \frac{\partial \theta}{\partial t}$
  - Hyperparameter krusial: α (learning rate), μ (friction)
- Momentum Nesterov (koreksi terhadap momentum):
  - Gradien dihitung setelah memperbaharui bobot
  - Hyperparameter krusial: α (learning rate), μ (friction)
- AdaGrad:
  - Parameter dengan gradien besar berkurang banyak, dan sebaliknya
  - Hyperparameter krusial: α (learning rate), δ (scaling)
- RMSProp:
  - Akumulasi gradien → weighted average
  - Hyperparameter krusial: α (learning rate), δ (scaling), ρ (decay rate)
- Adam (Adaptive Moments):
  - 1st moment: momentum, 2nd moment: akumulasi gradien
  - Hyperparameter krusial: α (learning rate), δ (scaling), $\rho_1 \& \rho_2$ (exponential decay rate)

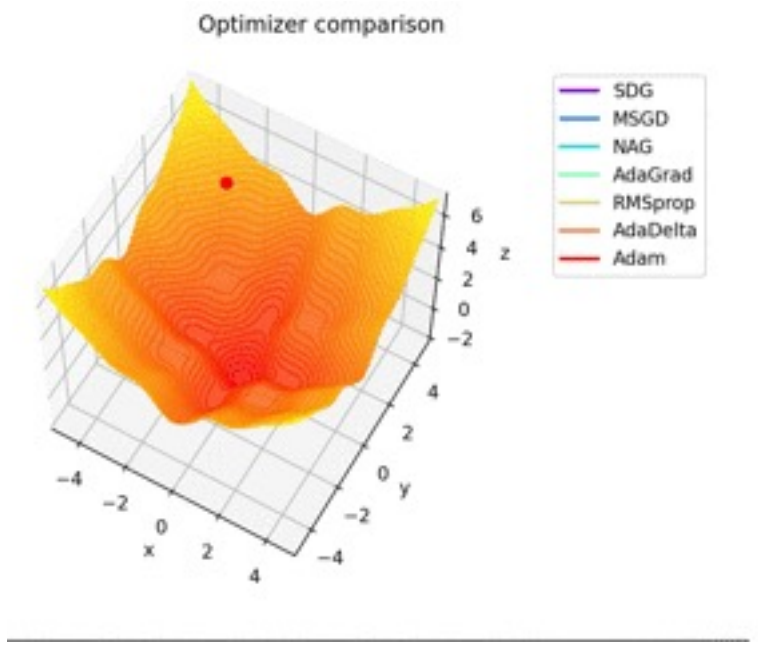
CALVIN INSTITUTE OF TECHNOLOGY

Perbandingan variasi optimisasi 1st order

- https://linuxtut.com/en/6695e0c79e888543e150/



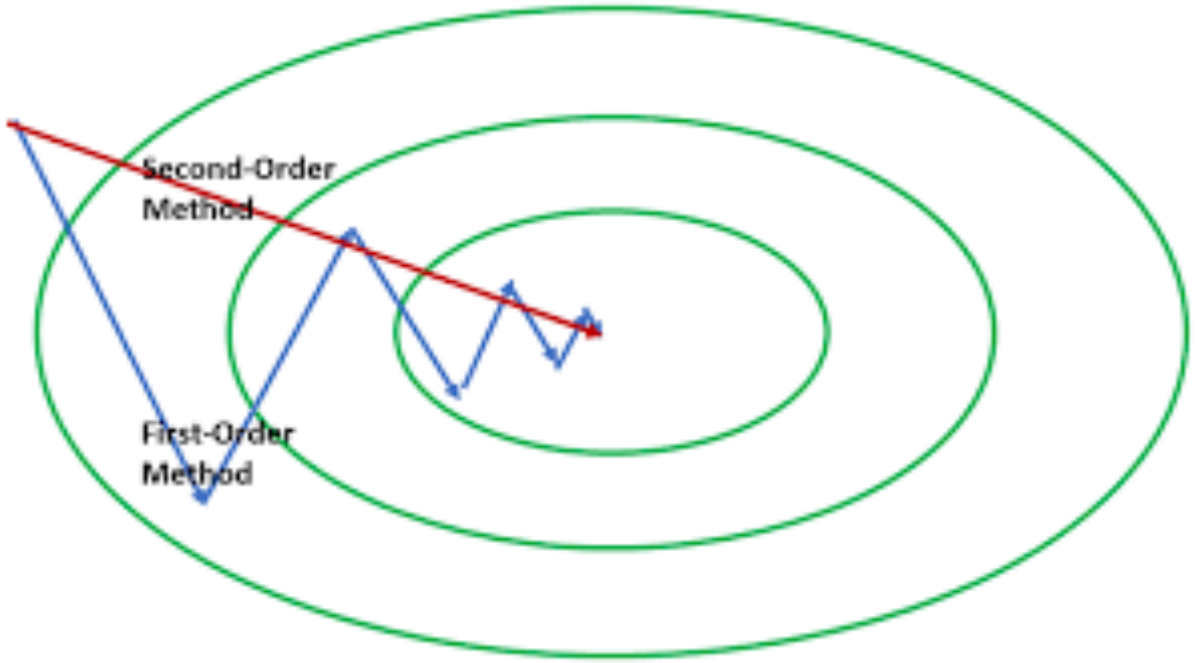


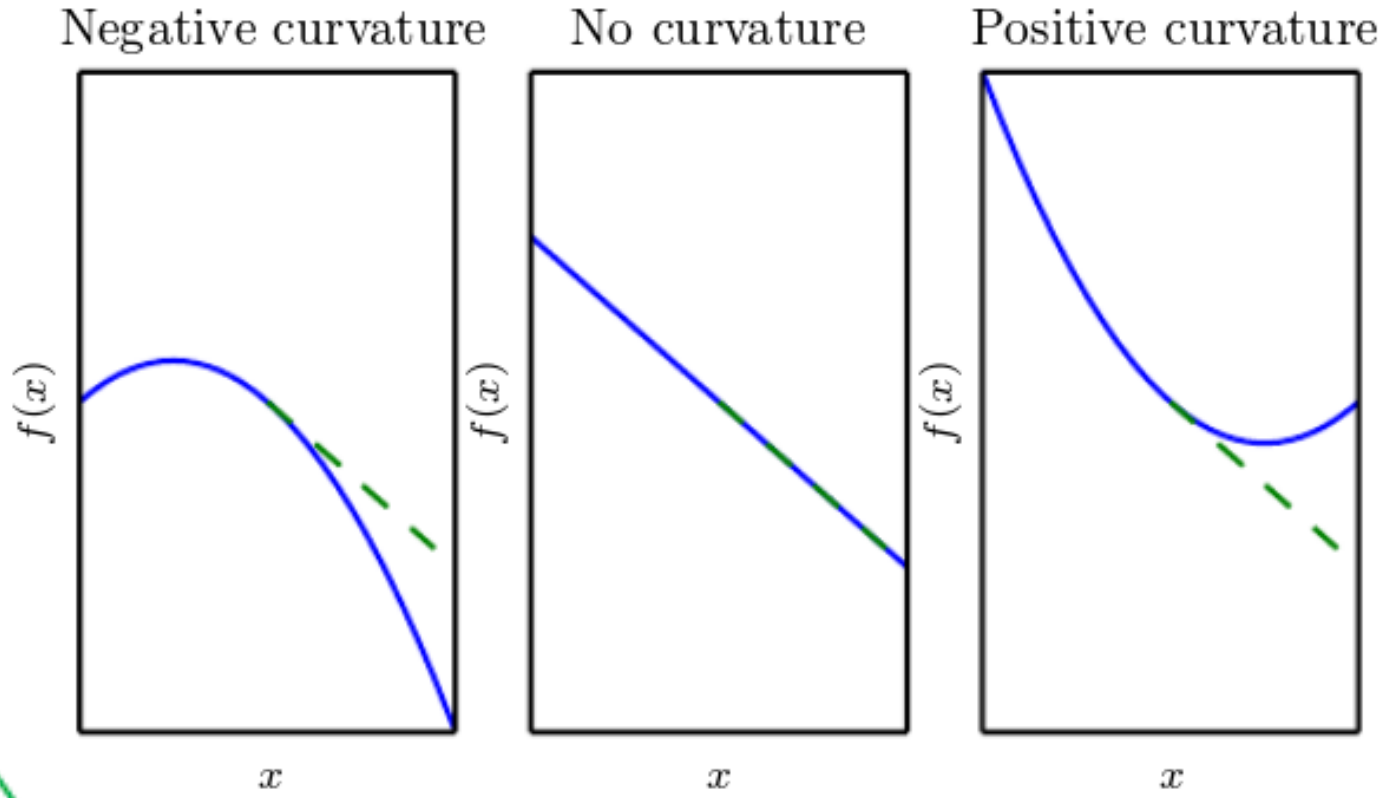


CALVIN INSTITUTE OF TECHNOLOGY

Optimisasi 2nd Order

- Metode Newton
- Metode konjugat
- BFGS





Metode Newton

- Gunakan ekspansi Taylor disekitar titik $\boldsymbol{\theta}_0$

$$J(\boldsymbol{\theta}) \approx J(\boldsymbol{\theta}_0) + (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^T \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_0) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^T \boldsymbol{H}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)$$

Dimana $\boldsymbol{H}$ adalah matriks Hessian

- Setiap iterasi harus menghitung Hessian
- Kompleksitas komputasi $O(n^3)$

BFGS

- Broyden–Fletcher–Goldfarb–Shanno (BFGS) mencoba memanfaatkan keunggulan metode Newton tanpa membawa beban komputasinya
  - Memerlukan beban memori $O(n^2)$

- Limited Memory BFGS (L-BFGS) mengurangi beban memori dengan tidak menyimpan aproksimasi inverse Hessian
  - Memerlukan beban memori $O(n)$

Optimisasi Khusus

gradient free

- Constrained optimization by linear approximation (COBYLA)
- Simultaneous Perturbation Stochastic Approximation (SPSA)
- Nelder-Mead
- Powell's method

Regresi Linear Multivariat

Regresi dengan Banyak Variabel

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	...	grade	sqft_above	sqft_baseme
0	7129300520	20141013T000000	221900.00	3	1.00	1180	5650	1.00	0	0	...	7	1180	0
1	6414100192	20141209T000000	538000.00	3	2.25	2570	7242	2.00	0	0	...	7	2170	400
2	5631500400	20150225T000000	180000.00	2	1.00	770	10000	1.00	0	0	...	6	770	0
3	2487200875	20141209T000000	604000.00	4	3.00	1960	5000	1.00	0	0	...	7	1050	910
4	1954400510	20150218T000000	510000.00	3	2.00	1680	8080	1.00	0	0	...	8	1680	0

Area → Price

$x \rightarrow y$

$h(x)\ \theta_0 + \theta_1 x \rightarrow y$

$x_1, x_2, x_3, \ldots \rightarrow y$

$\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \ldots \rightarrow y$

Regresi Linear Multivariat

- Hypothesis:
  - Regresi linear univariat:

$$h(x) = \theta_0 + \theta_1 x$$

  - Regresi linear multivariat:

$$h(\boldsymbol{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

  Dimana $\boldsymbol{x} \in \mathbb{R}^n$

- Notasi matriks:
  - Definisikan $x_0 = 1$

$$h(\boldsymbol{x}) = \sum_{j=0}^{n} \theta_j x_j = \boldsymbol{\theta}^T \boldsymbol{x}$$

  - Dimana $\boldsymbol{x} \in \mathbb{R}^{n+1}, \boldsymbol{\theta} \in \mathbb{R}^{n+1}$

Cost Function

- Hypothesis:

$$h(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

- Parameters:

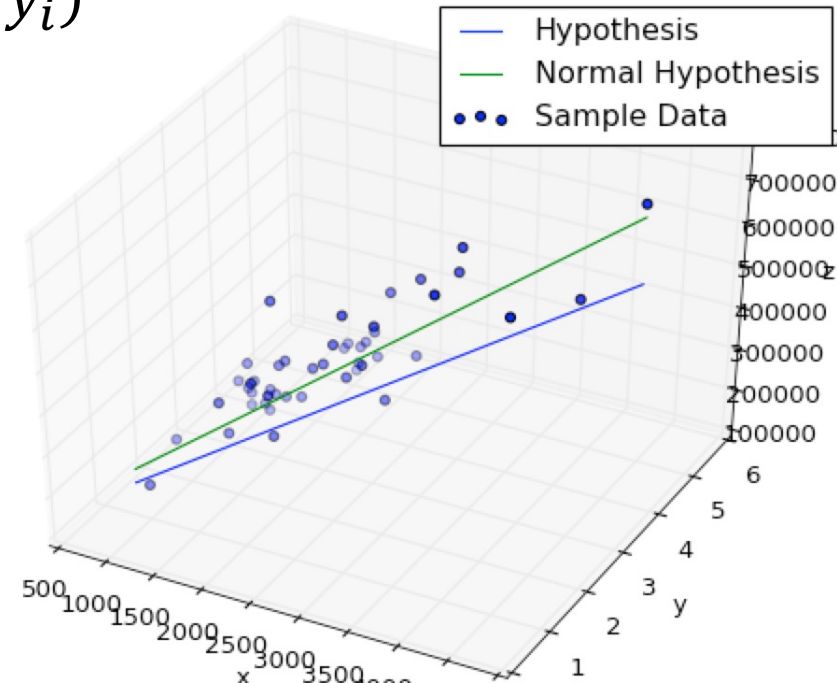
$$\theta_0, \theta_1, \theta_2, \ldots, \theta_n$$

- Cost function: mean squared error (MSE)

$$J(\theta_0, \theta_1, \theta_2, \ldots, \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h(x_i) - y_i)^2$$

- Goal:

$$\min_{\theta_0, \theta_1, \theta_2, \ldots, \theta_n} J(\theta_0, \theta_1, \theta_2, \ldots, \theta_n)$$



CALVIN INSTITUTE OF TECHNOLOGY

Gradient descent untuk regresi linear

- Model regresi linear:

$$h(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

- Cost function:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h\left(x^{(i)}\right) - y^{(i)} \right)^2$$

- Gradient descent:

$$\theta_0 := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h\left(x^{(i)}\right) - y^{(i)} \right)$$

$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h\left(x^{(i)}\right) - y^{(i)} \right) x_1^{(i)}$$

$\theta_2 := \theta_2 - \alpha \frac{\partial J}{\partial \theta_2}$

$x_2 \ldots$

...

CALVIN INSTITUTE OF TECHNOLOGY

Bentuk Matriks

- Model regresi linear:

$$h(\boldsymbol{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n = \boldsymbol{\theta}^T \boldsymbol{x}$$

- Cost function:

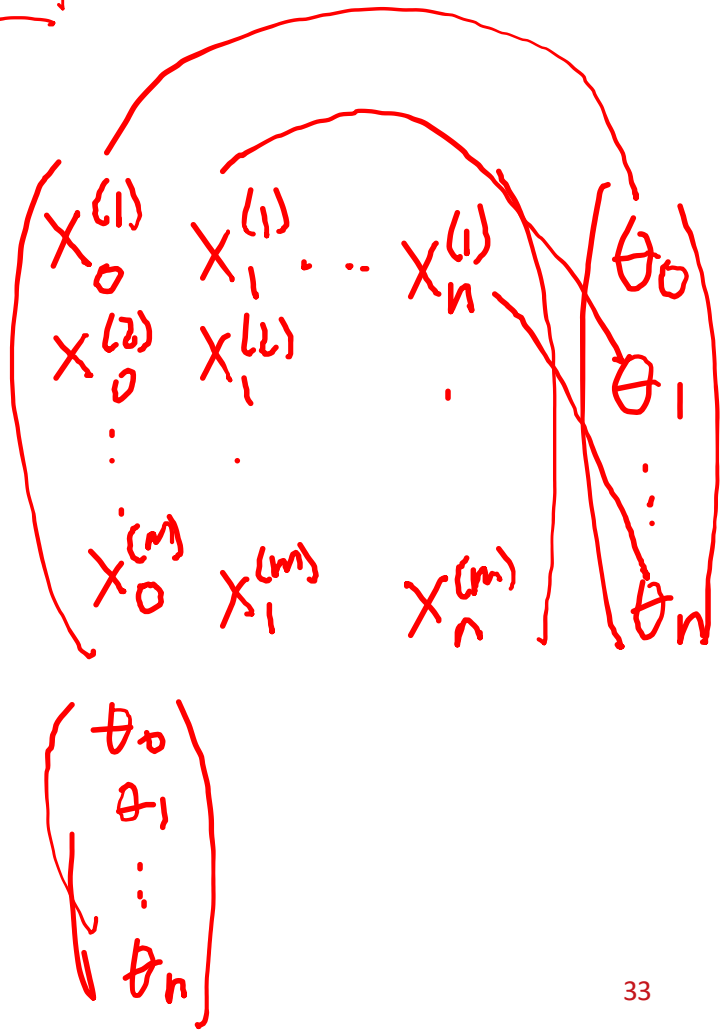
$$J(\boldsymbol{\theta}) = \frac{1}{2m} \sum_{i=1}^{m} \left(h\left(\boldsymbol{x}^{(i)}\right) - y^{(i)} \right)^2$$

- Turunan parsial:

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \begin{pmatrix} \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_0} \\ \vdots \\ \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_n} \end{pmatrix} = \frac{1}{m} \boldsymbol{X}^T (\boldsymbol{X}\boldsymbol{\theta} - \boldsymbol{y})$$

- Gradient descent:

$$\boldsymbol{\theta} := \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$$



Solusi Analitik

- Normal equation:

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(\boldsymbol{\theta}^T\boldsymbol{x}^{(i)} - y^{(i)}\right)^2 = \frac{1}{2m}(\boldsymbol{X\theta} - \boldsymbol{y})^T(\boldsymbol{X\theta} - \boldsymbol{y})$$

$$\boldsymbol{X\theta} - \boldsymbol{y} = \begin{pmatrix} x_0^{(1)} & x_1^{(1)} & \cdots & x_n^{(1)} \\ x_0^{(2)} & x_1^{(2)} & \cdots & x_n^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ x_0^{(m)} & x_1^{(m)} & \cdots & x_n^{(m)} \end{pmatrix}\begin{pmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{pmatrix} - \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{pmatrix}$$

- Turunan nol:

$$\mathbf{0} = \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} \frac{1}{2m}(\boldsymbol{X\theta} - \boldsymbol{y})^T(\boldsymbol{X\theta} - \boldsymbol{y}) = \boldsymbol{X}^T\boldsymbol{X\theta} - \boldsymbol{X}^T\boldsymbol{y} \rightarrow \boldsymbol{X}^T\boldsymbol{X\theta} = \boldsymbol{X}^T\boldsymbol{y}$$

- Solusi analitik:

$$\boldsymbol{\theta} = \left((\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\right)\boldsymbol{y}$$

Uji Pemahaman

X (i) → index data
3 → index variabel fitur

- Berapakah selisih hasil prediksi regresi dengan nilai aslinya jika:
  - $x = 2, y = 2.5, \theta_0 = 1, \theta_1 = 1$ → linear univariat
  - $x_1 = 1, x_2 = 2, y = 5, \theta_0 = 0, \theta_1 = 1, \theta_2 = 2$ → multivariat

$h(x) = \theta_0 + \theta_1 x$

$h(2) = 1 + 1 \cdot 2 = 3$

$h(2) - y = 3 - 2.5 = 0.5$

$h(x_1, x_2) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$

$h(1,2) = 0 + 1 \cdot 1 + 2 \cdot 2 = 5$

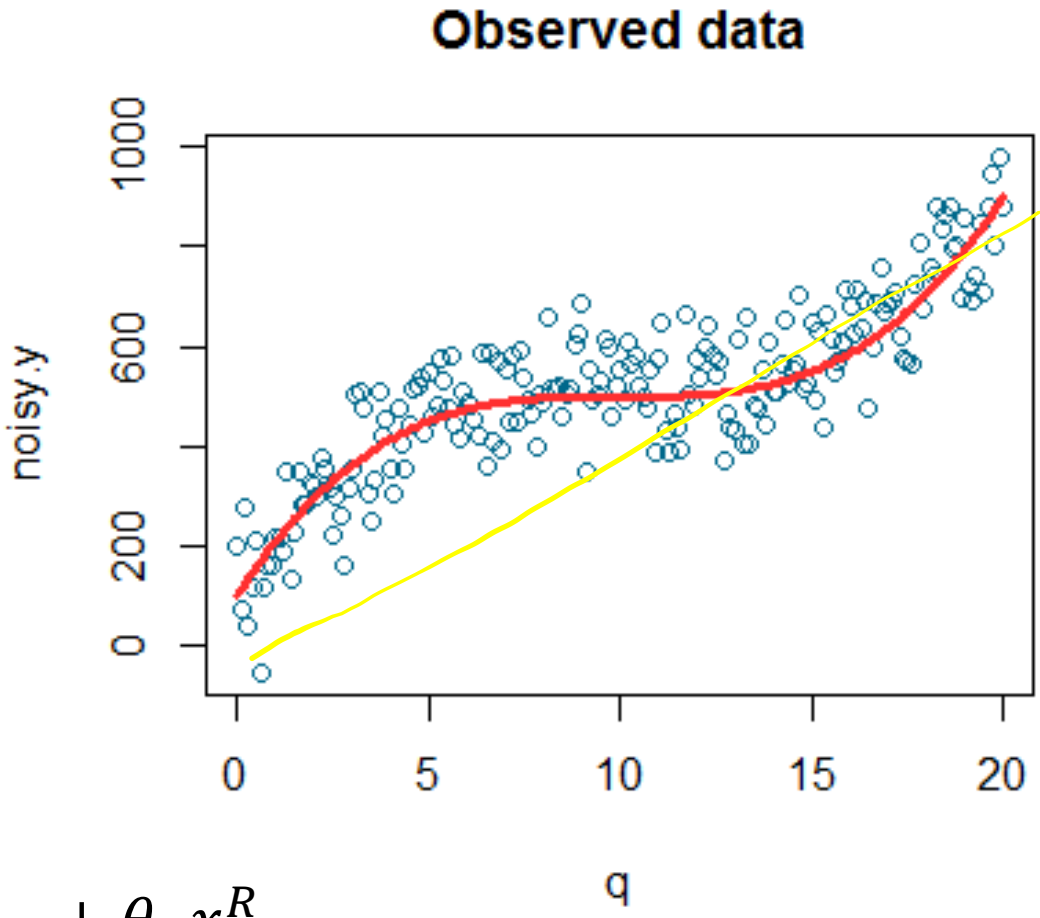
$h(1,2) - y = 5 - 5 = 0$

Regresi Nonlinear

Nonlinear

- Bagaimana jika data tidak linear?
- Misalnya perlu variabel komposit:
  - Luas = Panjang x lebar
  - Standard deviasi
  - dll

$X_1 X_2$



- Hipotesis paling sederhana adalah polinomial:

$$h(x) \neq \theta_0 + \theta_1 x$$

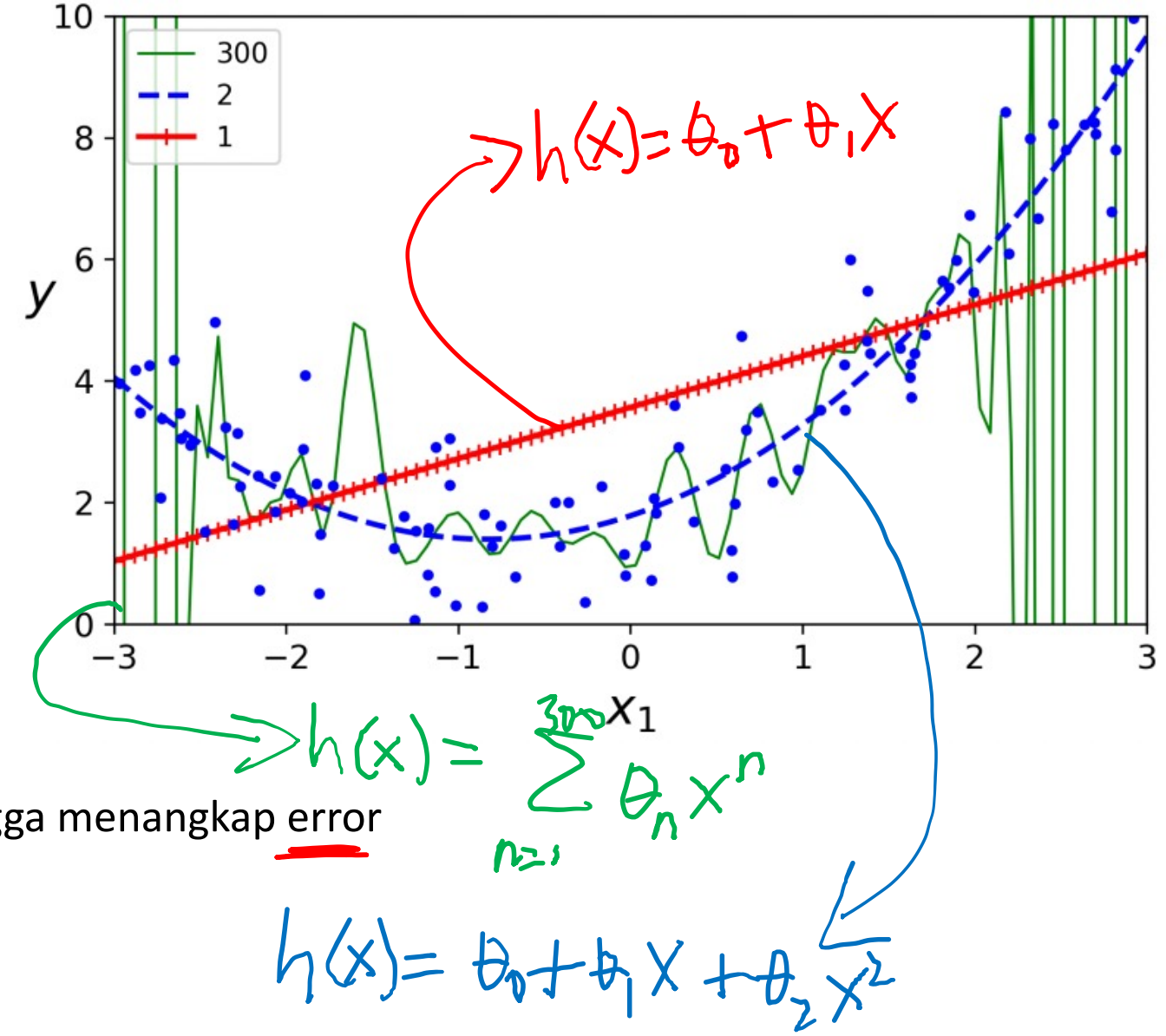
$$h(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \cdots + \theta_R x^R$$

$$h(x) = \sum_{r=0}^{\infty} \theta_r x^r$$

Underfit vs Overfit

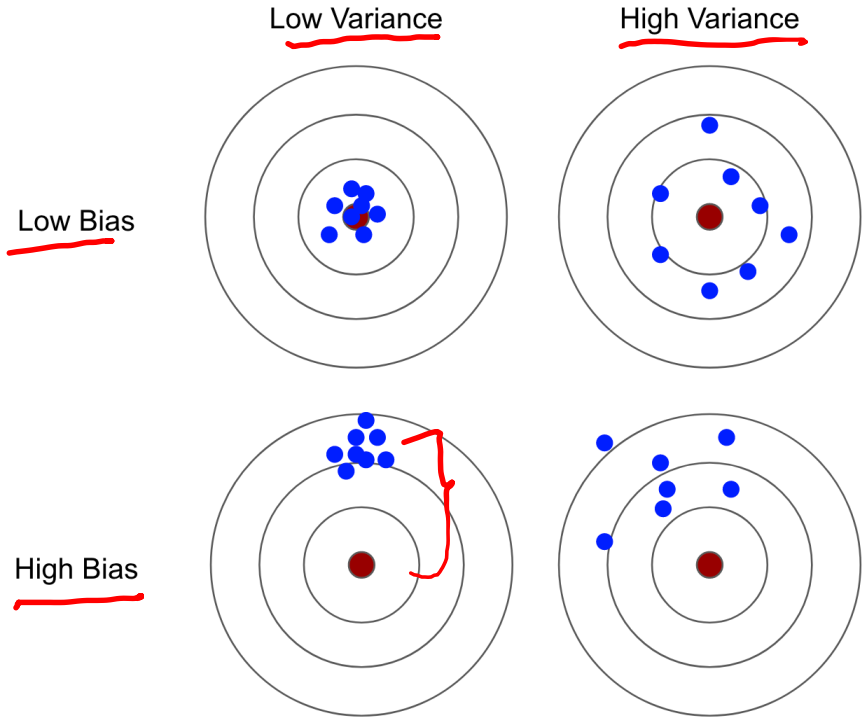
- Bias-variance trade off

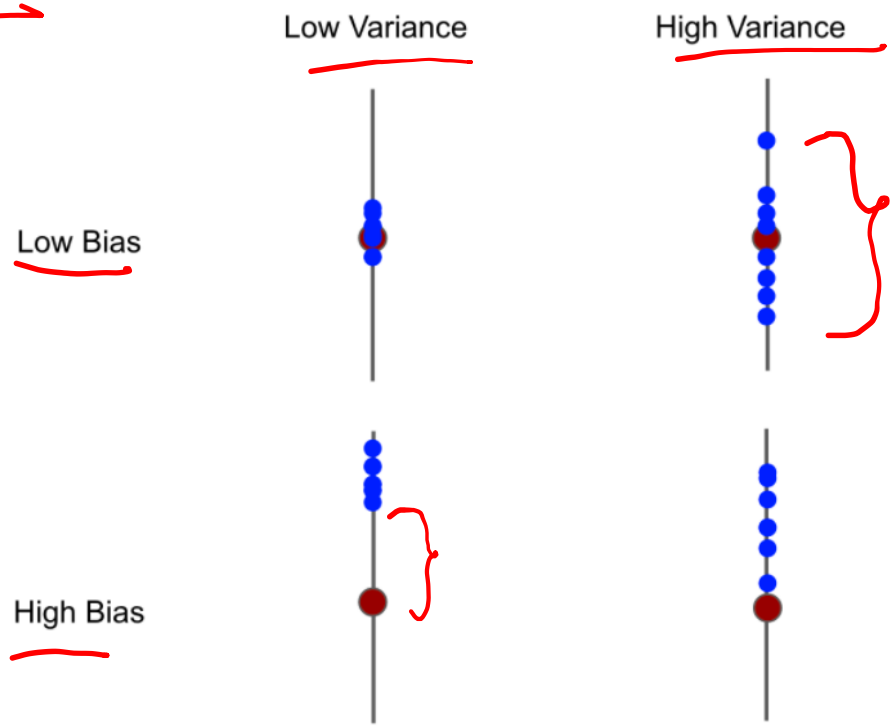
- Bias:
  - model terlalu sederhana
  - Tidak dapat fit terhadap data
  - underfit
- Variance:
  - model terlalu kompleks
  - Terlalu sensitif terhadap variasi data sehingga menangkap error
  - overfit



Bias vs Variance

- Bias → error karena asumsi model berbeda dengan realita (terdapat perbedaan dari prediksi ke nilai sebenarnya)
- Varians → error karena sensitifitas model terhadap fluktuasi kecil di training set
- Irreducable error → intrinsic noise pada dataset

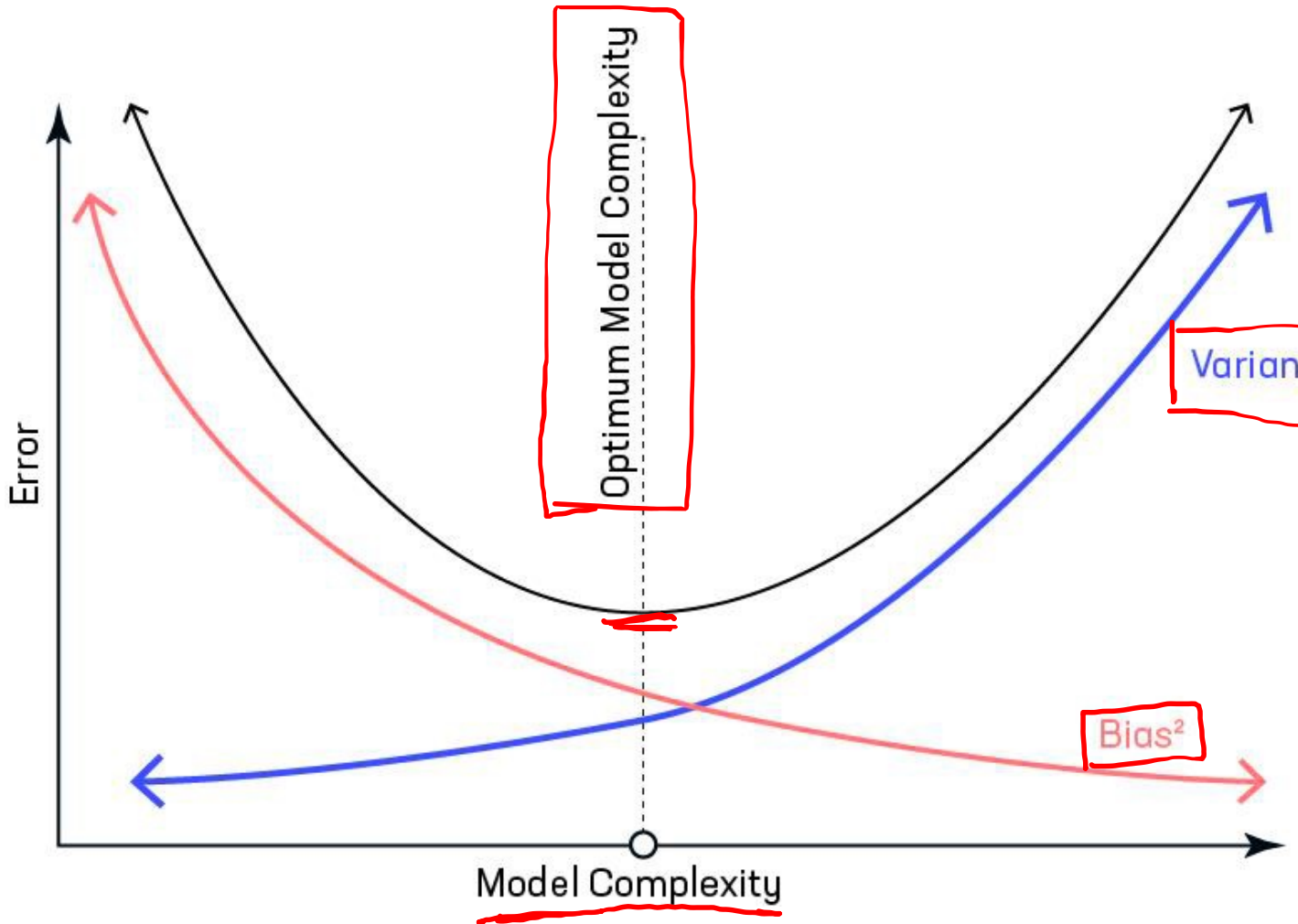


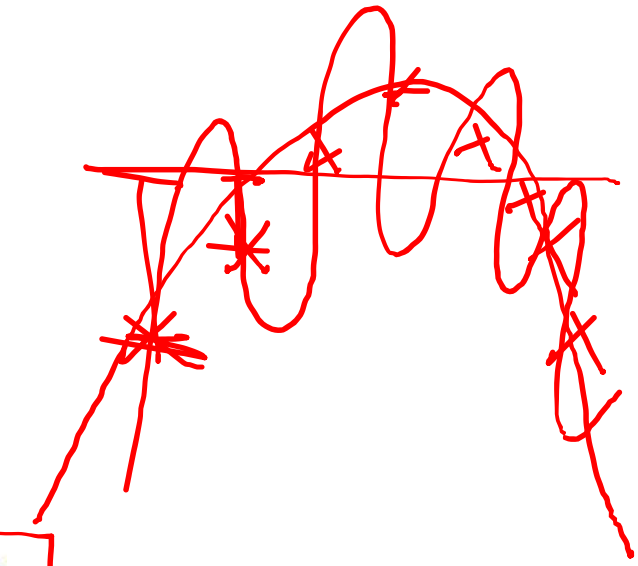


CALVIN INSTITUTE OF TECHNOLOGY

Bias – variance tradeoff

- Makin kompleks model, makin tinggi variance, makin rendah bias





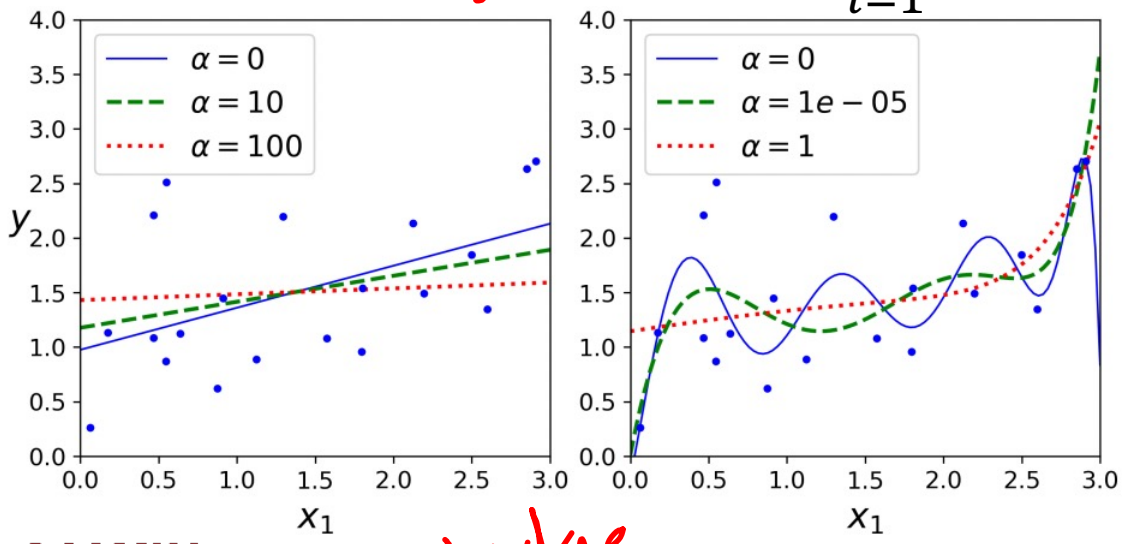
Regularisasi

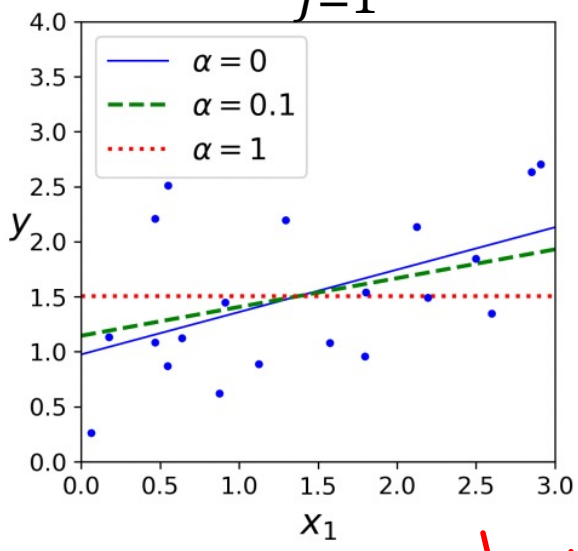
- Ridge/l2:

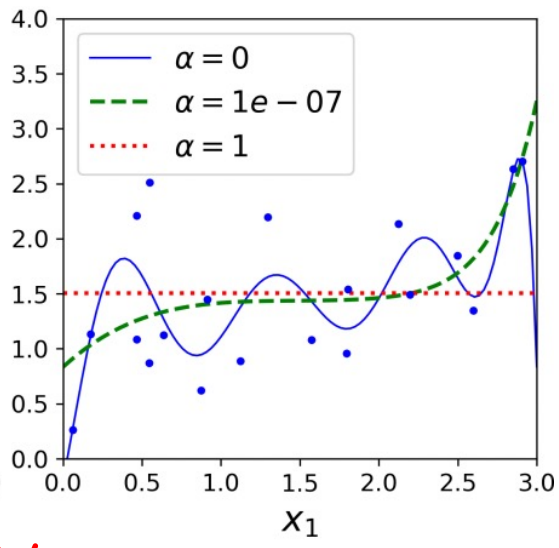
$$J(\boldsymbol{\theta}) = \frac{1}{2m}\sum_{i=1}^{m}\left(h\left(\boldsymbol{x}^{(i)}\right) - y^{(i)}\right)^2 + \frac{\alpha}{2}\sum_{j=1}^{n}\theta_j^2$$

- Lasso/l1:

$$J(\boldsymbol{\theta}) = \frac{1}{2m}\sum_{i=1}^{m}\left(h\left(\boldsymbol{x}^{(i)}\right) - y^{(i)}\right)^2 + \alpha\sum_{j=1}^{n}|\theta_j|$$







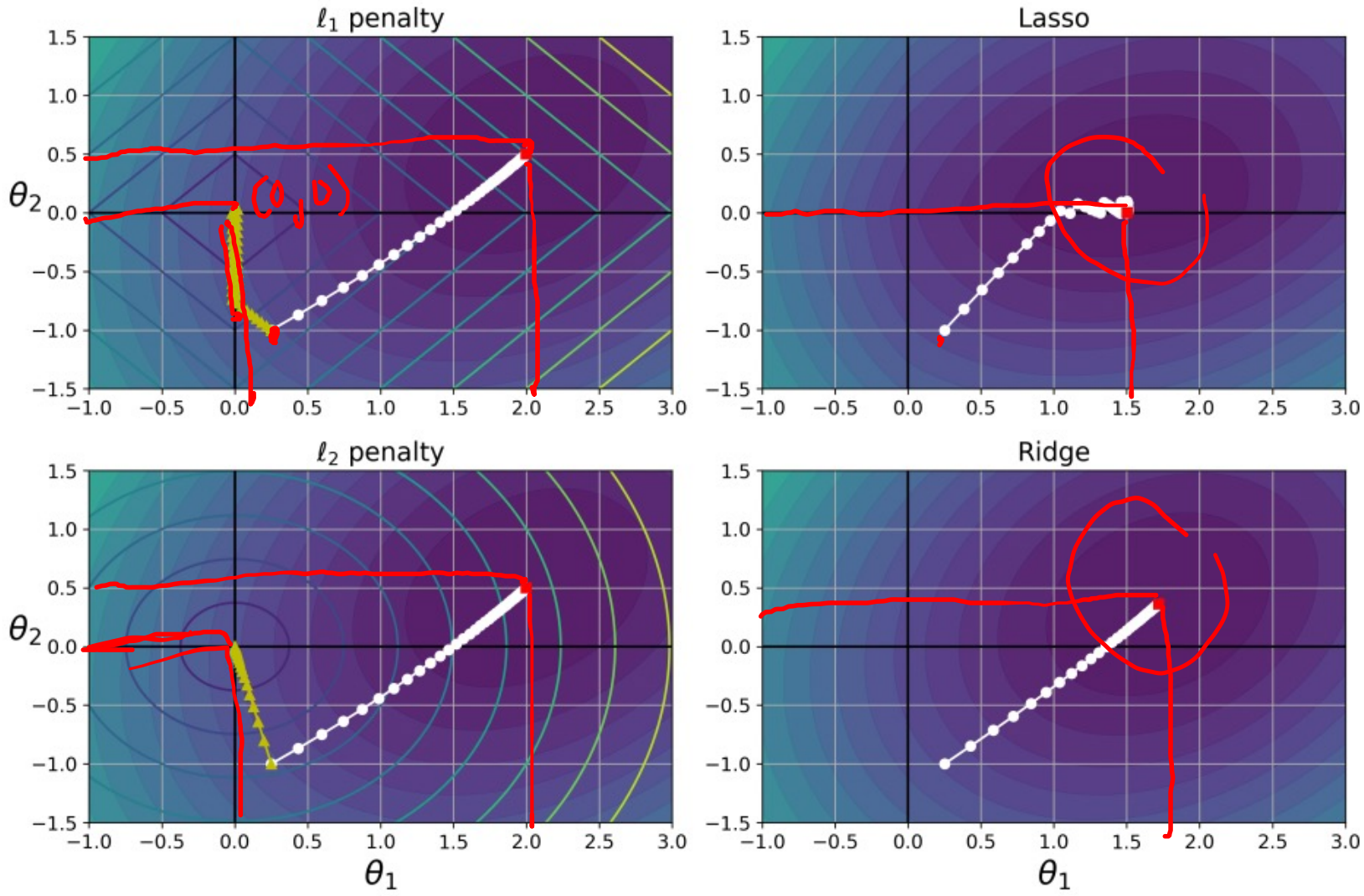
ridge

lasso

CALVIN INSTITUTE OF TECHNOLOGY

Regularized Cost Function

$$\sum (h(x^{(i)}) - y^{(i)})^2 + \sum \theta_j^2$$



ℓ_1

ℓ_2

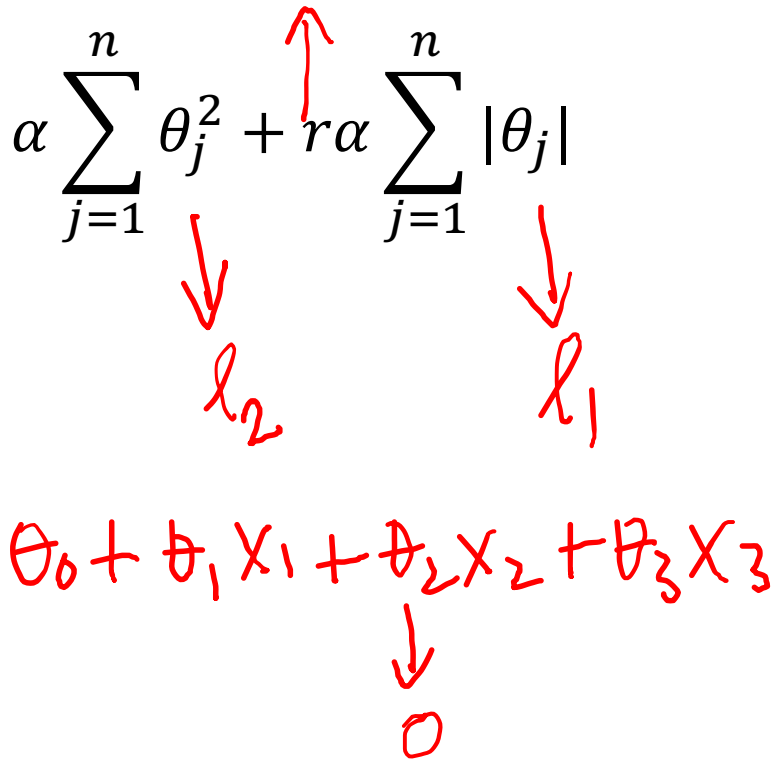
$\theta_2 = 0$

Elastic Net

- Kompromi antara ridge dan lasso
  - Ridge tidak bisa menseleksi fitur irelevan
  - Lasso tidak stabil ketika jumlah fitur lebih banyak dari jumlah sampel karena memilih paling banyak m fitur walaupun fitur relevan bisa lebih
- Cost function:

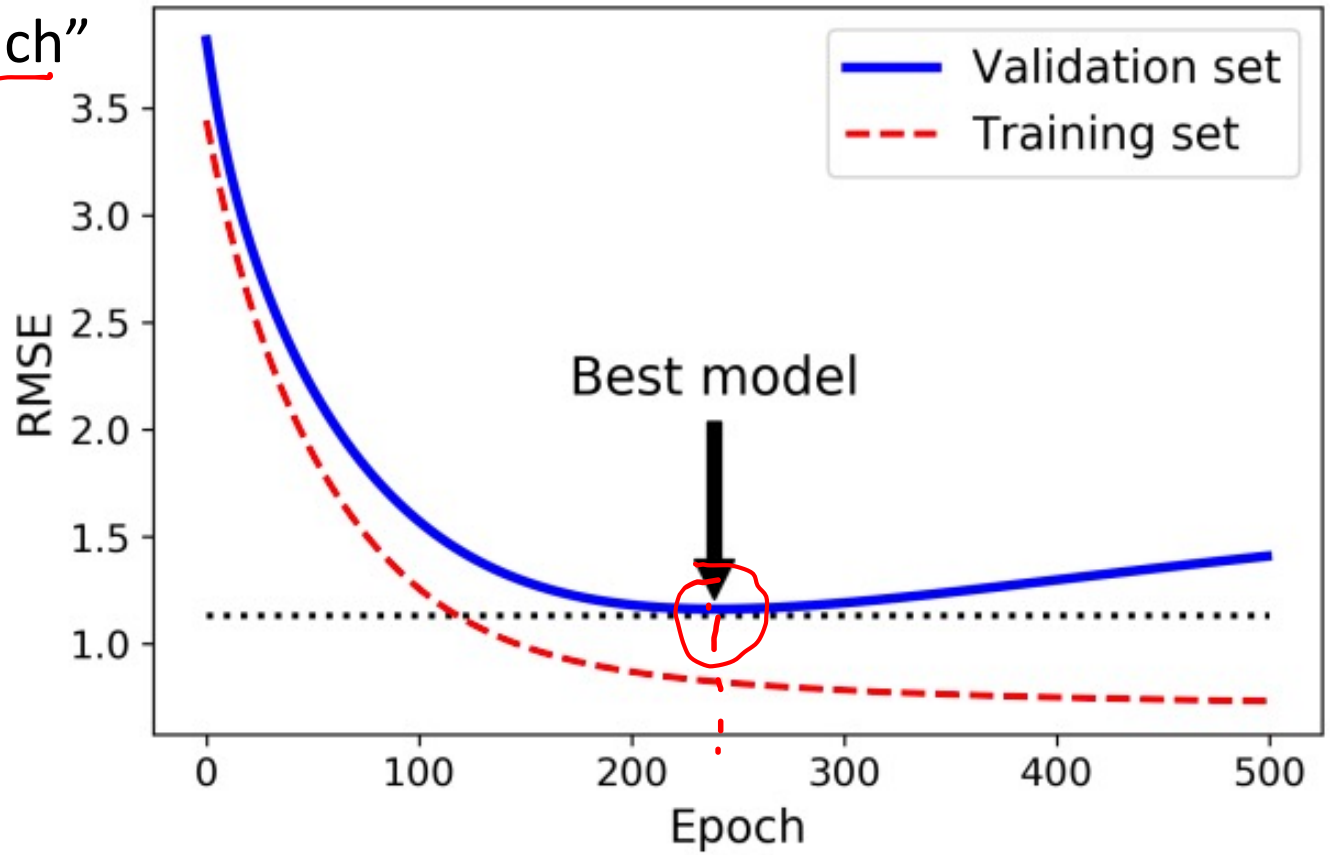
$$J(\boldsymbol{\theta}) = \frac{1}{2m}\sum_{i=1}^{m}\left(h\left(\boldsymbol{x}^{(i)}\right) - y^{(i)}\right)^2 + \frac{(1-r)}{2}\alpha\sum_{j=1}^{n}\theta_j^2 + r\alpha\sum_{j=1}^{n}|\theta_j|$$

- Dimana $r = 0 \rightarrow$ridge dan $r = 1 \rightarrow$lasso
- Rule of thumb:
  - Ridge: dapat digunakan untuk data apapun
  - Lasso: jika ingin membuat beberapa fitur irelevan
  - Elastic net lebih stabil daripada lassi Ketika $n > m$



Early Stopping

- Salah satu cara simpel untuk regularisasi adalah menghentikan iterasi begitu validation error mencapai minimum
- Ketika error naik→mulai overfit→stop
- Hinton menyebutnya “beautiful free lunch”



CALVIN INSTITUTE OF TECHNOLOGY

Uji Pemahaman

- Sebuah regresi nonlinear memiliki bentuk $\hat{y} = \theta_0 + \theta_1 x + \theta_2 x^2$. Berapakah selisih hasil prediksi regresi dengan nilai aslinya jika:
  - $x = 0, y = 0, \theta_0 = 0, \theta_1 = 1, \theta_2 = 2$
  - $x = 2, y = 10, \theta_0 = 0, \theta_1 = 1, \theta_2 = 2$

$h(0) = \theta_0 + \theta_1 0 + \theta_2 0^2 = 0$

$y = 0$

$h(0) - y = 0$

$h(2) = 0 + 1 \times 2 + 2 \times 2^2 = 10$

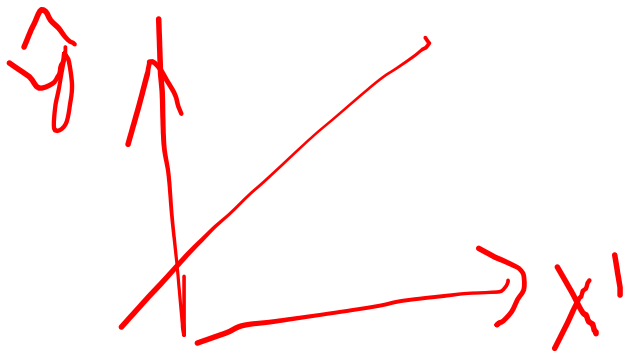
$y = 10 \quad h(2) - y = 0$

Uji Pemahaman

- Bagaimana merubah regresi nonlinear berikut menjadi regresi linear?
  - $\hat{y} = \theta_1 \log(x) + \theta_0$
  - $\hat{y} = \theta_1 x_1^2 + \theta_1 x_2^2 + \theta_1 x_1 x_2 + \theta_0$

$x' = \log x$

$\hat{y} = \theta_1 x' + \theta_0$



$\theta(x_1 + x_2)^2 = \theta x_1^2 + \theta x_2^2 + 2\theta x_1 x_2$

$\theta_1 = \theta$

$\theta_2 = \theta$

$\theta_{12} = 2\theta$

$\theta(x_1 + x_2)^2 + \theta_0$

$(x_1 + x_2)^2 = x'$

$\theta x' + \theta_0$

Regresi Logistik

Basis Function

- Kombinasi linear dari variabel input $\boldsymbol{x}$:

$$h(x) = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n$$

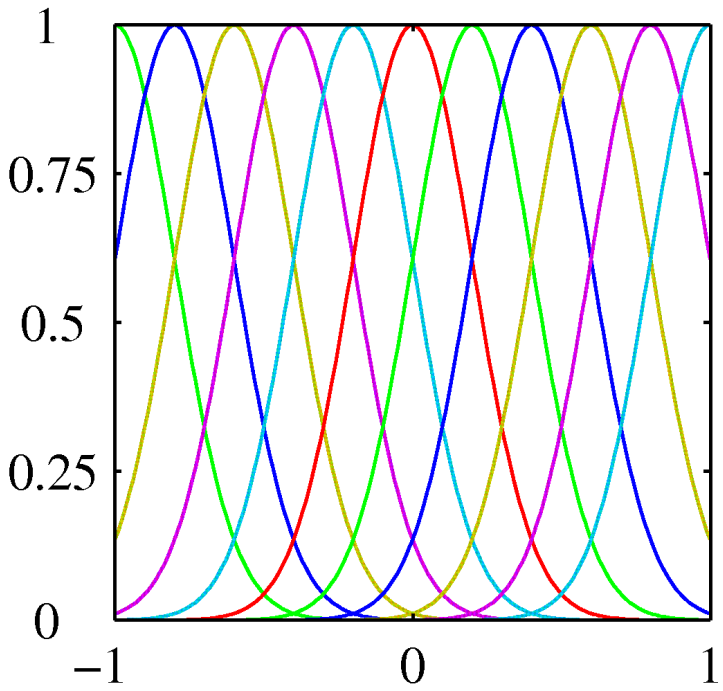
MV regression

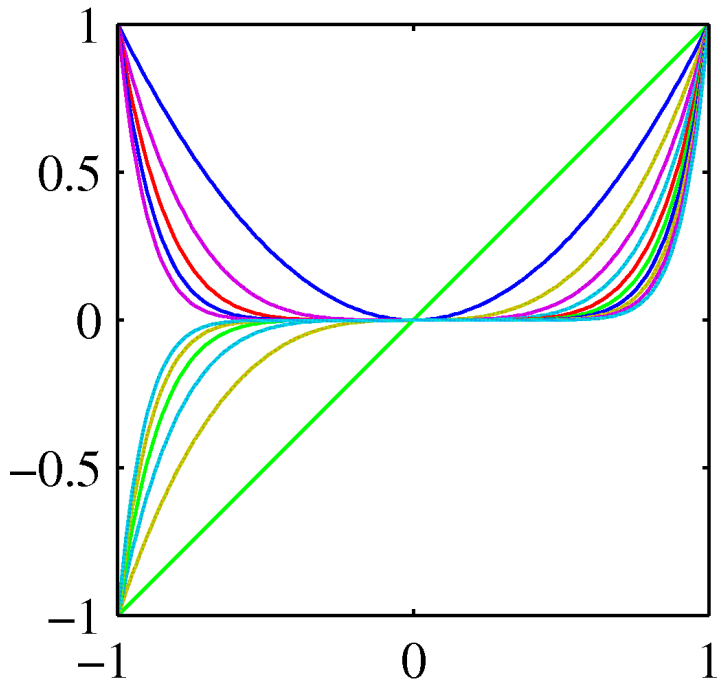
- Kombinasi linear dari (nonlinear) basis function ϕ dari variabel input $\boldsymbol{x}$:

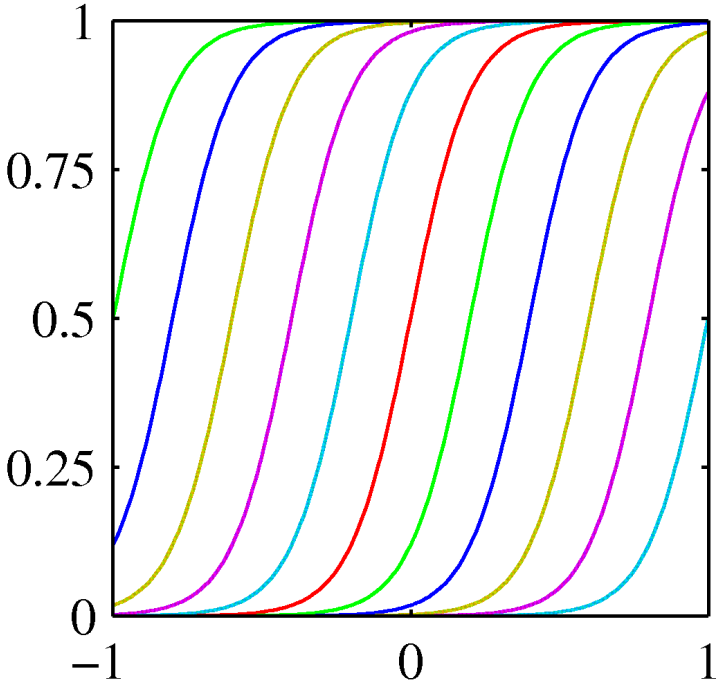
$$h(x) = \theta_0 + \theta_1 \phi_1(x_1) + \cdots + \theta_n \phi_n(x_n)$$

Contoh basis functions

- Polinomial:
  - $\phi_j(x) = x^j$
  - global
- Gaussian:
  - $\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$
  - lokal
- Sigmoidal:
  - $\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right)$
  - lokal







CALVIN INSTITUTE OF TECHNOLOGY

Gradient Descent

basis function

- Model regresi:

$$h(x) = \theta_0 + \theta_1\phi_1(x_1) + \cdots + \theta_n\phi_n(x_n)$$

- Cost function:

$$J(\boldsymbol{\theta}) = \frac{1}{2m}\sum_{i=1}^{m}\left(h\left(\boldsymbol{x}^{(i)}\right) - y^{(i)}\right)^2 = \frac{1}{2m}\sum_{i=1}^{m}\left(\sum_{j=0}^{n}\theta_j\phi_j(\boldsymbol{x}^{(i)}) - y^{(i)}\right)^2$$

- Gradient descent:

$$\boldsymbol{\theta} := \boldsymbol{\theta} - \alpha\nabla_{\boldsymbol{\theta}}J(\boldsymbol{\theta})$$

$$\theta_j = \theta_j - \alpha\frac{1}{m}\sum_{i=1}^{m}\left(h\left(\boldsymbol{x}^{(i)}\right) - y^{(i)}\right)\phi_j(\boldsymbol{x}^{(i)})$$

CALVIN INSTITUTE OF TECHNOLOGY

Solusi Analitik

- Mirip dengan solusi analitik untuk regresi linear, solusi analitik untuk basis function adalah:

$$\boldsymbol{\theta} = \left((\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\right)\boldsymbol{y}$$

- Dimana:

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_0\left(x^{(1)}\right) & \phi_1\left(x^{(1)}\right) & \cdots & \phi_n\left(x^{(1)}\right) \\ \phi_0\left(x^{(2)}\right) & \phi_1\left(x^{(2)}\right) & \cdots & \phi_n\left(x^{(2)}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0\left(x^{(m)}\right) & \phi_1\left(x^{(m)}\right) & \cdots & \phi_n\left(x^{(m)}\right) \end{pmatrix}$$

Regresi Logistik

- Regresi dapat digunakan untuk proses klasifikasi
- Regresi logistik menggunakan sigmoidal basis function dimana fungsi ini merepresentasikan suatu peluang dari kombinasi parameter:

$$\hat{p} = h(\boldsymbol{x}) = \sigma(\boldsymbol{x}^T \boldsymbol{\theta})$$

- Regresi logistik adalah klasifikasi biner dimana jika peluang diatas 50% maka model akan memprediksi positif (1), jika peluang dibawah 50% maka diprediksi negative (0):

$$\hat{y} = \begin{cases} 0 \; jika \; \hat{p} < 0.5 \\ 1 \; jika \; \hat{p} \geq 0.5 \end{cases}$$

- Cost function menggunakan cross entropy:

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log\left(\hat{p}^{(i)}\right) + (1 - y^{(i)}) \log\left(1 - \hat{p}^{(i)}\right) \right]$$

- Turunan parsial:

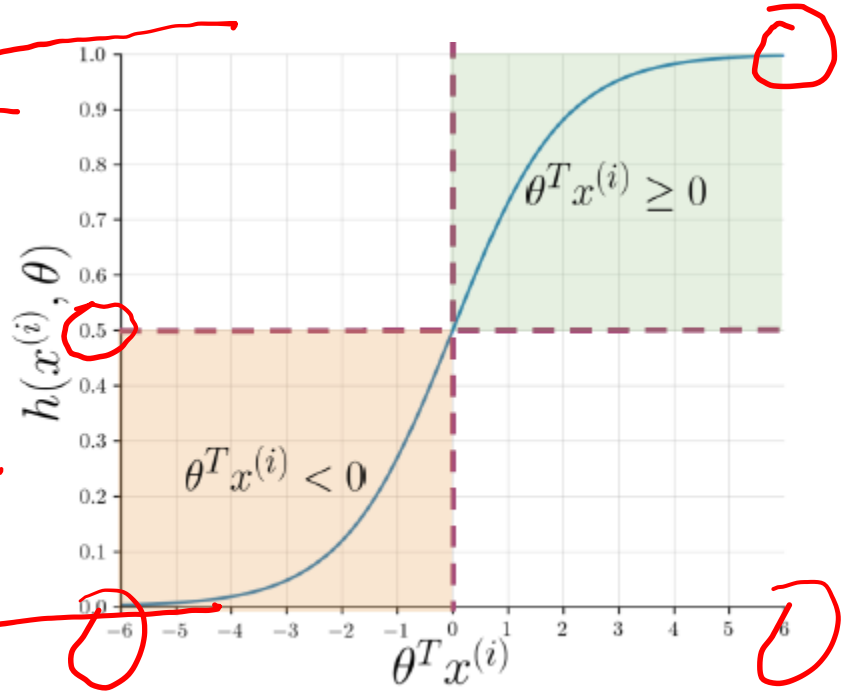
$$\frac{\partial}{\partial \theta_j} J(\boldsymbol{\theta}) = \frac{1}{m} \sum_{i=1}^{m} \left(\sigma\left(\boldsymbol{\theta}^T \boldsymbol{x}^{(i)}\right) - y^{(i)} \right) x_j^{(i)}$$

CALVIN INSTITUTE OF TECHNOLOGY

Regresi logistik

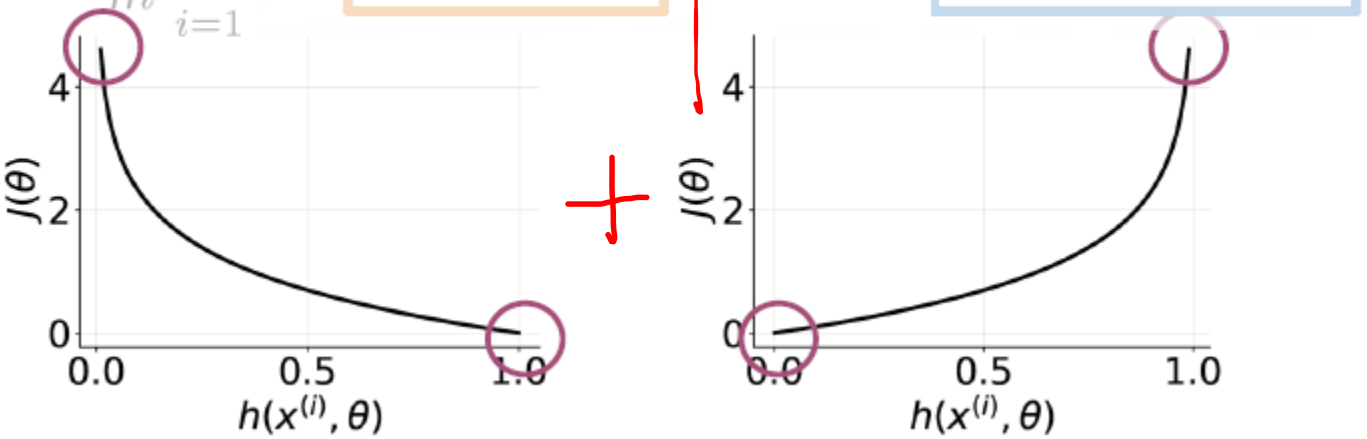
$h(x) \in [0, 1]$

$$h(x^{(i)}, \theta) = \frac{1}{1 + e^{-\theta^T x^{(i)}}}$$

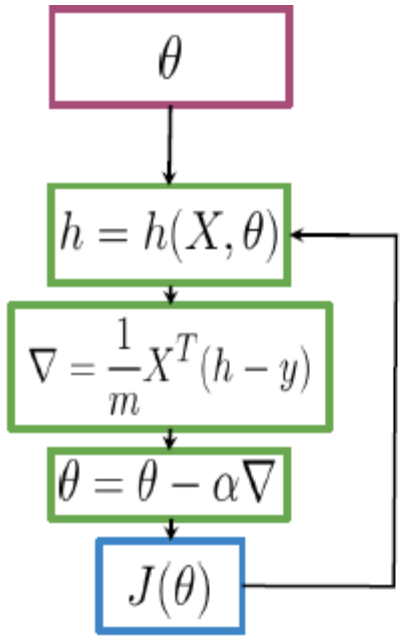


Cost function dari fungsi regresi logistik

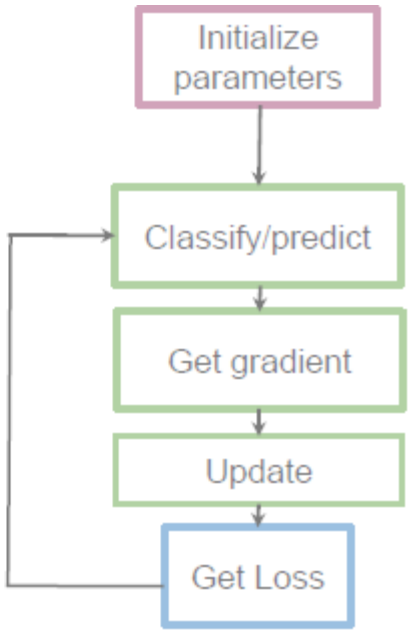
$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)}\log h(x^{(i)},\theta) + (1-y^{(i)})\log(1-h(x^{(i)},\theta))]$$

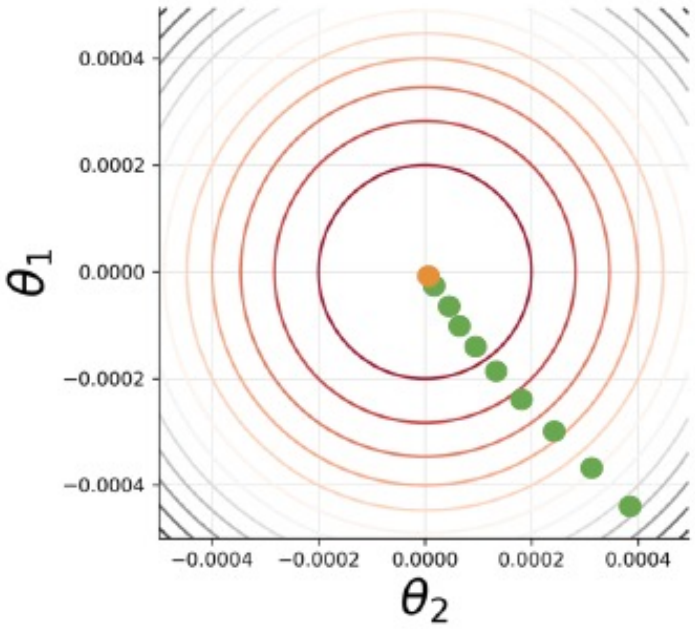


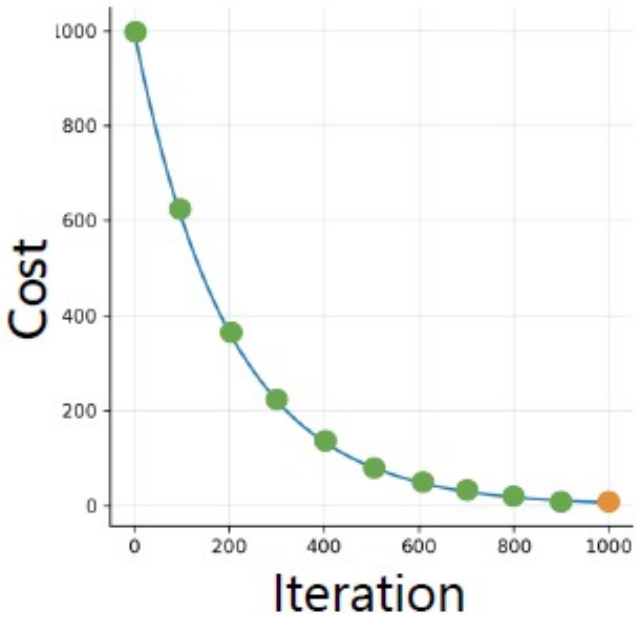
Pelatihan regresi logistik



Until good enough

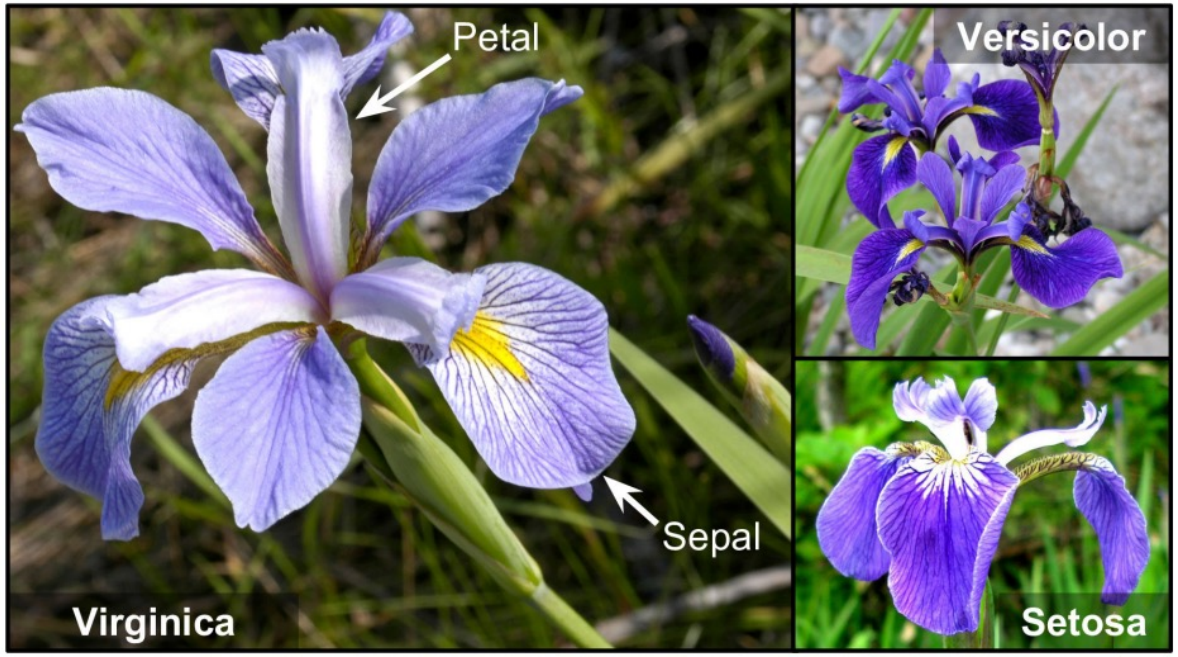


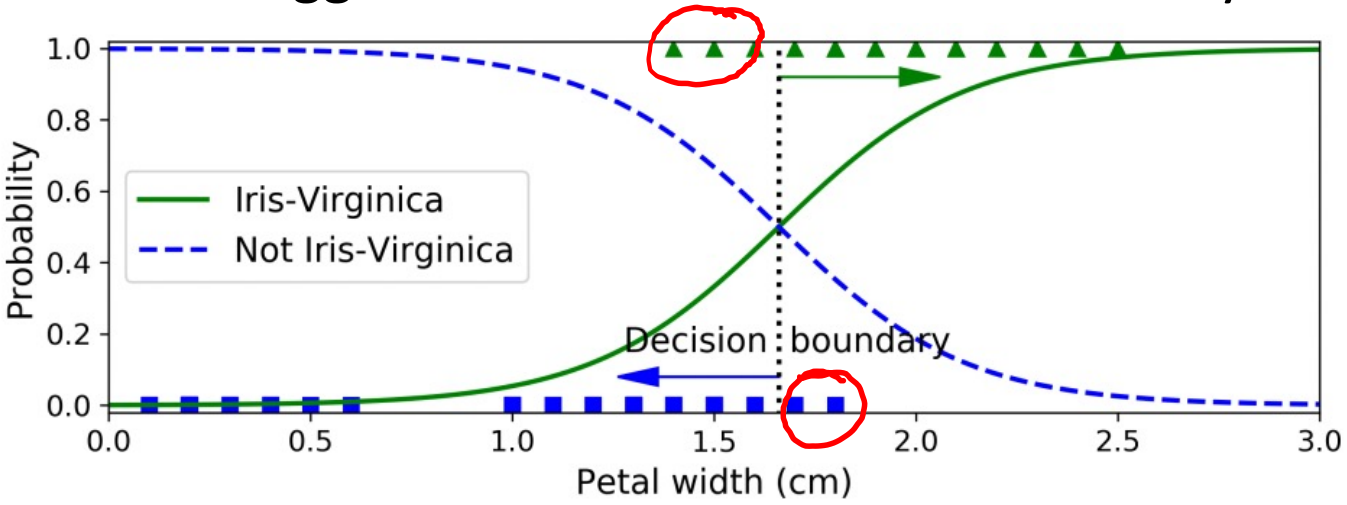


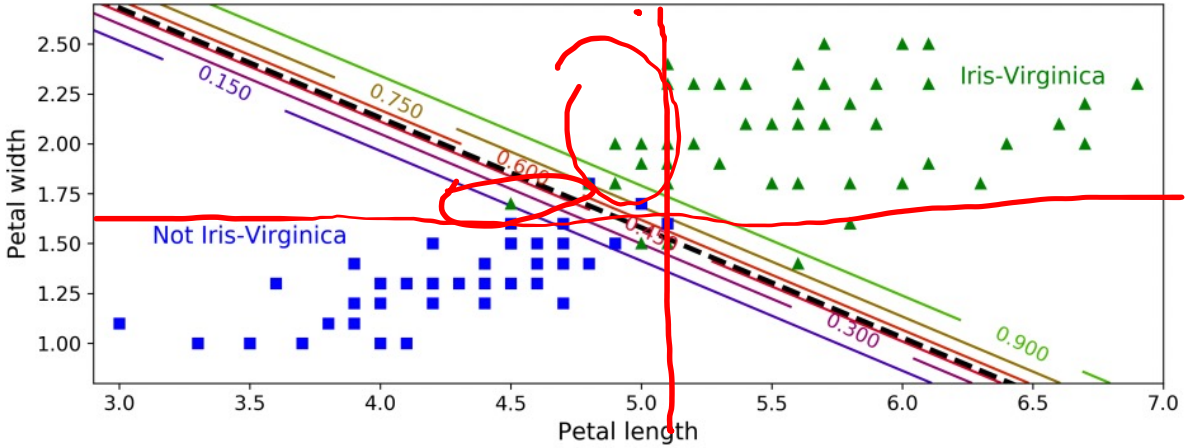


Decision Boundary

- Regresi logistik akan menciptakan decision boundary diantara 2 kelas
- Seringkali klasifikasi kelas sulit diputuskan dengan menggunakan 1 fitur saja karena bisa terdapat overlap antara 2 kelas
- Overlap ini dapat diperkecil dengan menggunakan kombinasi fitur-fitur lainnya







Regresi Softmax

- Model regresi logistik dapat digeneralisasi untuk multi-kelas, tanpa menggabungkan berbagai klasifikasi biner→regresi softmax
- Ide: Ketika diberikan data $\boldsymbol{x}$, model regresi softmax akan menghitung skor $s_k(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{\theta}^{(k)}$ untuk setiap kelas k
- Setelah menghitung skor setiap kelas untuk data $\boldsymbol{x}$, peluang $\hat{p}_k$ data tersebut merupakan kelas k adalah:

$$\hat{p}_k = \frac{\exp\left(s_k(\boldsymbol{x})\right)}{\sum_{j=1}^{K} \exp\left(s_j(\boldsymbol{x})\right)}$$

- Cost function menggunakan cross entropy:

$$J(\boldsymbol{\Theta}) = -\frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{K} y_k^{(i)} \log\left(\hat{p}_k^{(i)}\right)$$

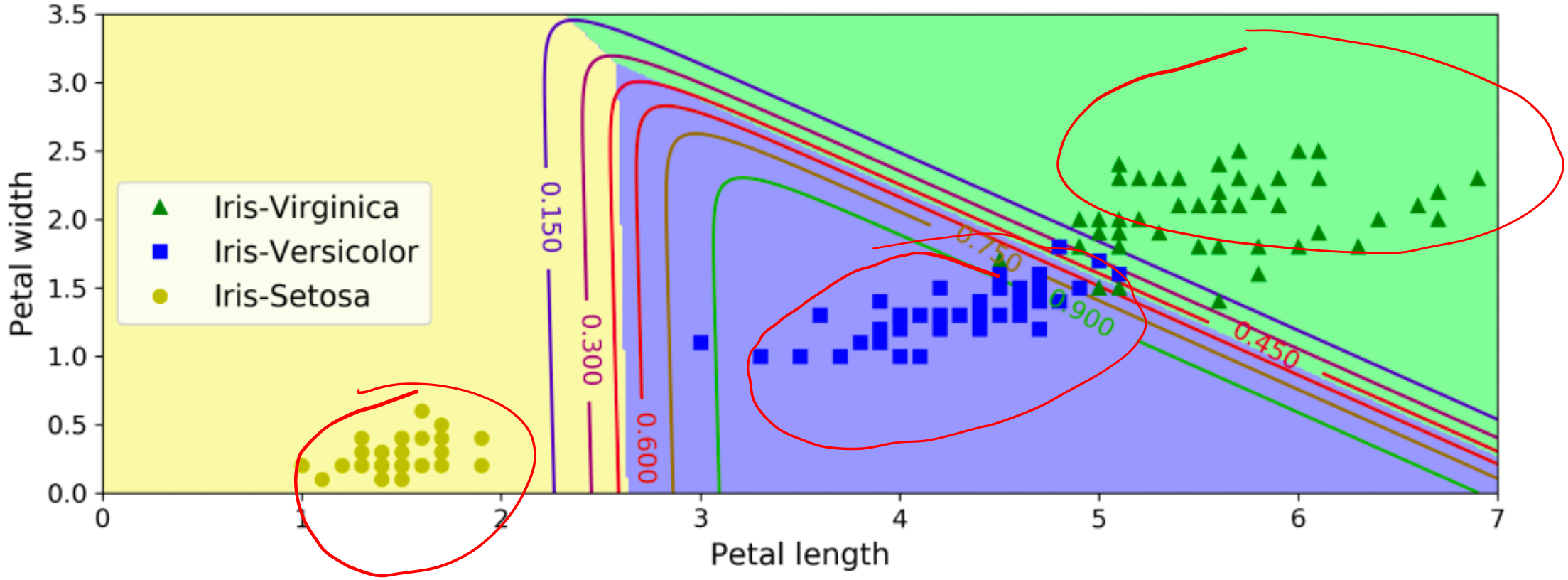
- Jika hanya 2 kelas (K=2), maka cost function akan tereduksi menjadi regresi logistik

CALVIN INSTITUTE OF TECHNOLOGY

Regresi Softmax

- Vektor gradien untuk tiap kelas k adalah:

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{m} \sum_{i=1}^{m} \left(\hat{p}_k^{(i)} - y_k^{(i)} \right) \boldsymbol{x}^{(i)}$$



CALVIN INSTITUTE OF TECHNOLOGY

Tuhan Memberkati