Machine Learning

Amin Golzari Oskouei

a.golzari@azaruniv.ac.ir

a.golzari@tabrizu.ac.ir

https://github.com/Amin-Golzari-Oskouei

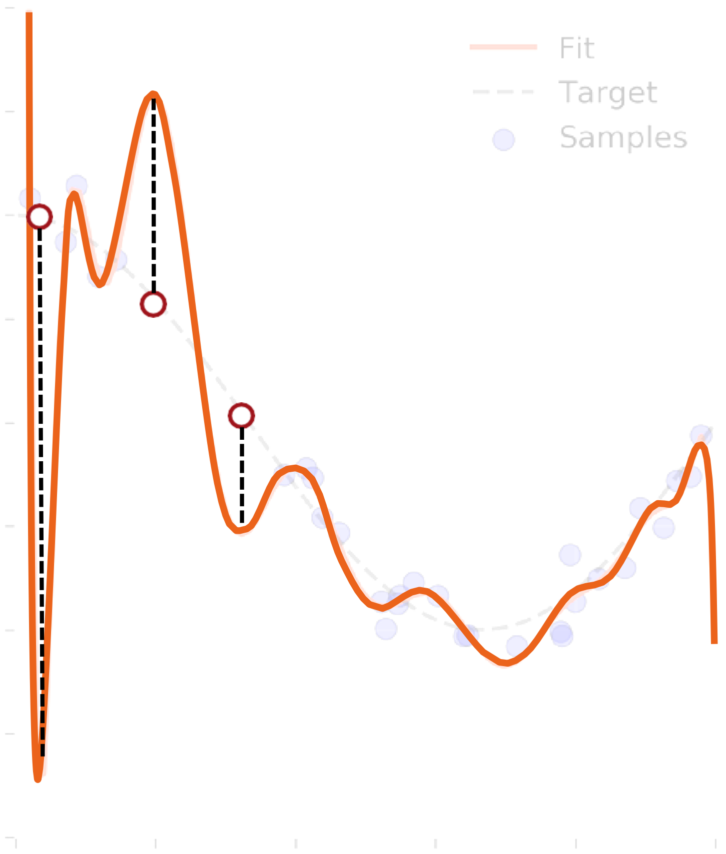
Azarbaijan Shahid Madani University

2023

Regularization: Dealing (Facing) With Overfitting

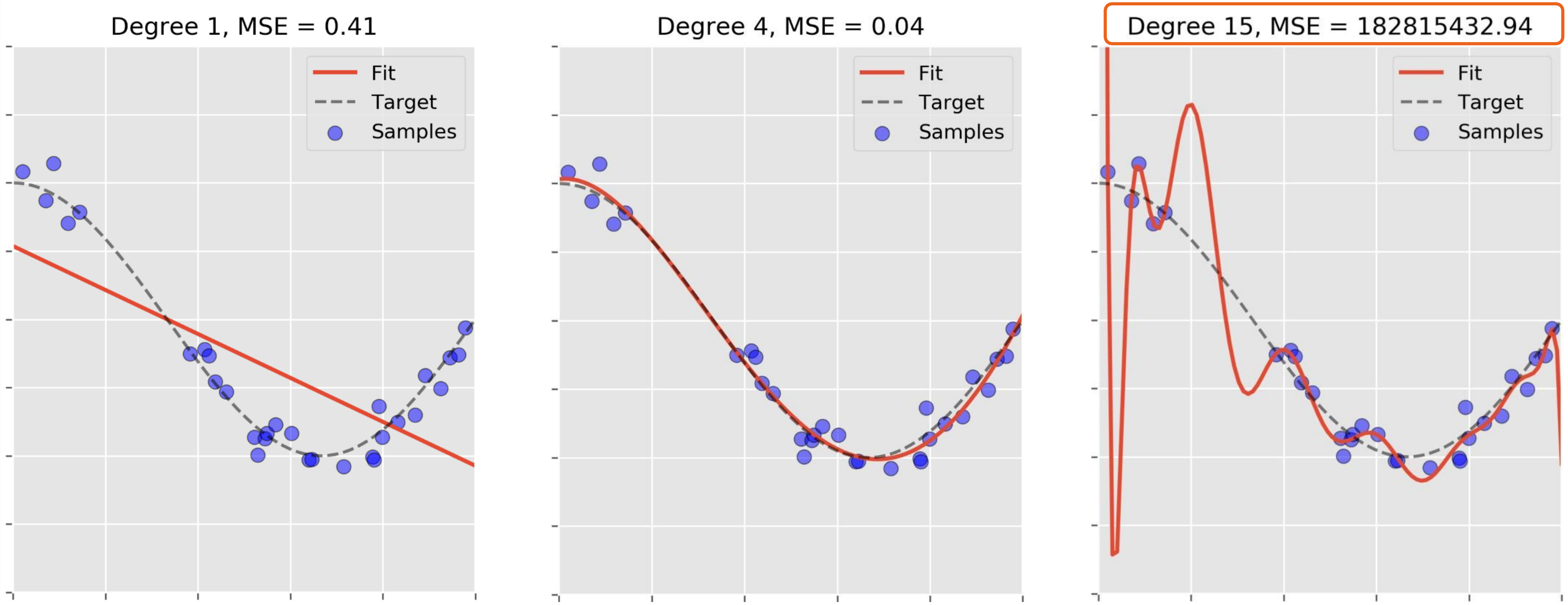
Regularization & Overfitting

- Overfitting. A common problem in machine learning
  - The model that is overly complicated
    - For example, due to the large number of features
  - Excellent performance of model on the training data
  - Awful performance of the model on the new data
    - Unable to generalize to new data!



- Regularization. An effective way to reduce or remove overfitting.

Overfitting & Regression

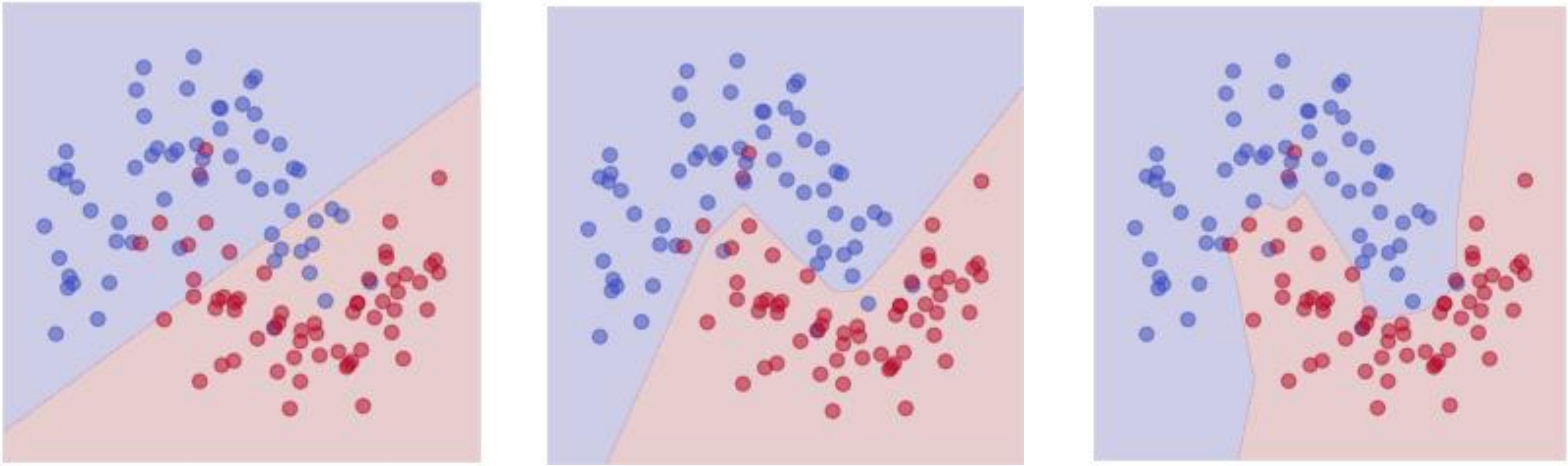


Underfitting

Correct Model

Overfitting

Overfitting & Classification



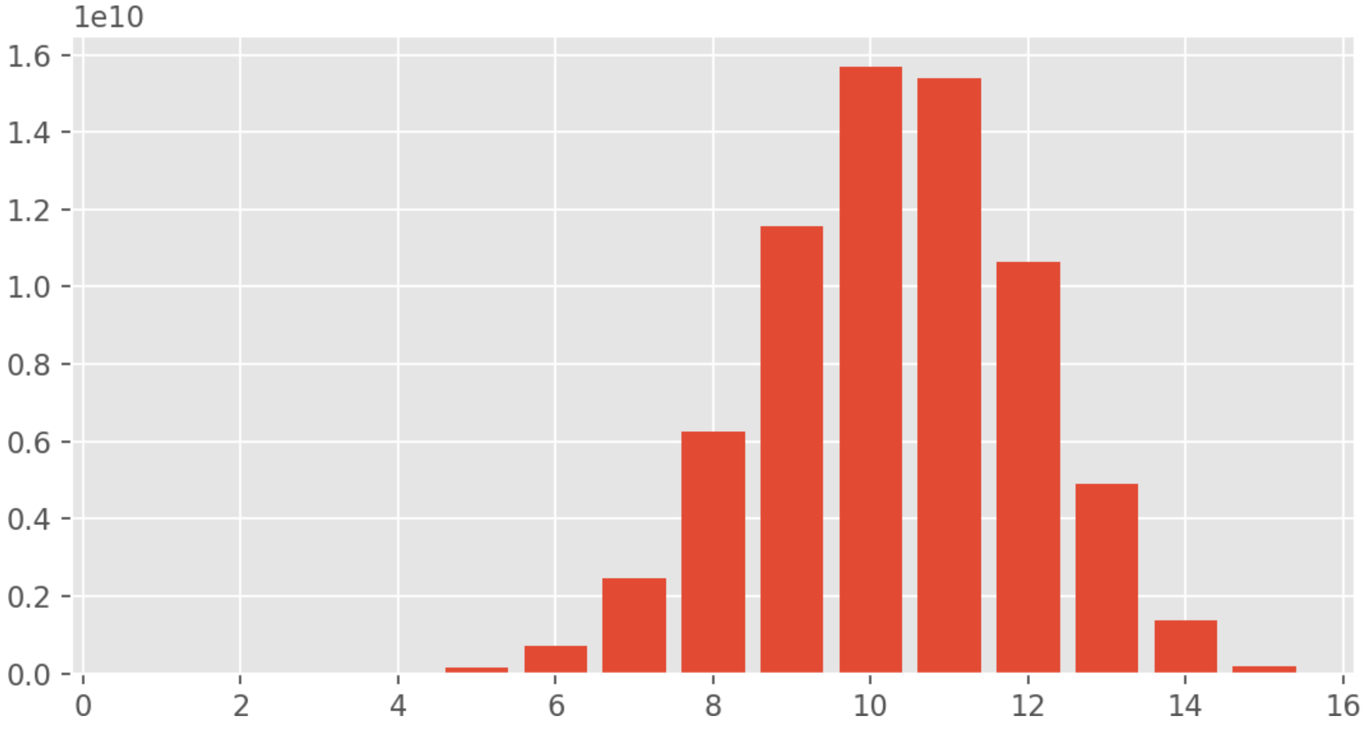
Underfitting

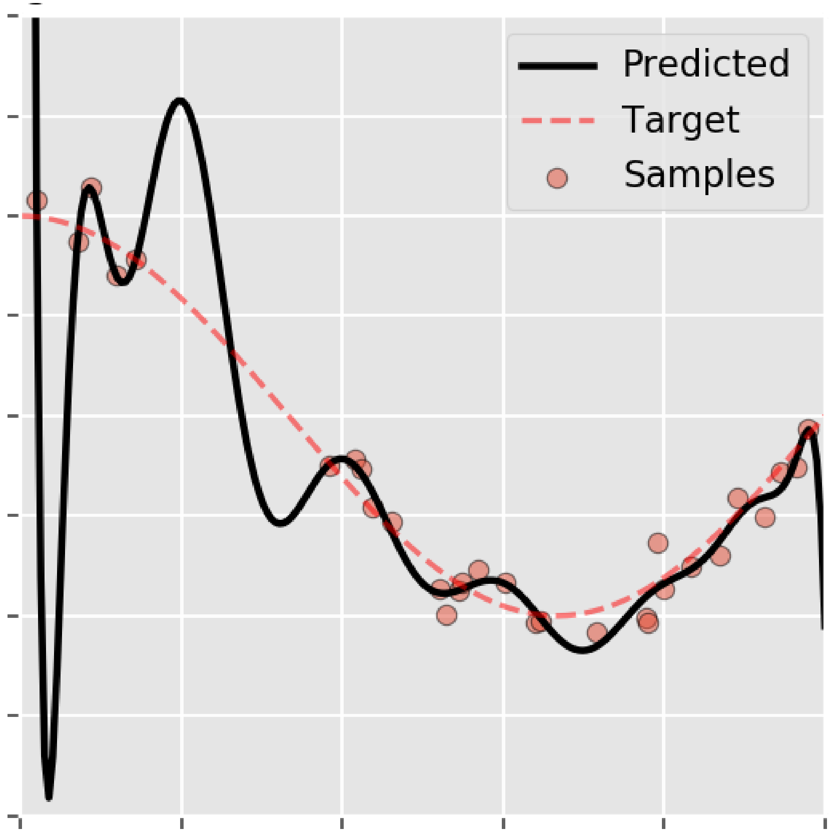
Correct Model

Overfitting

Regularization

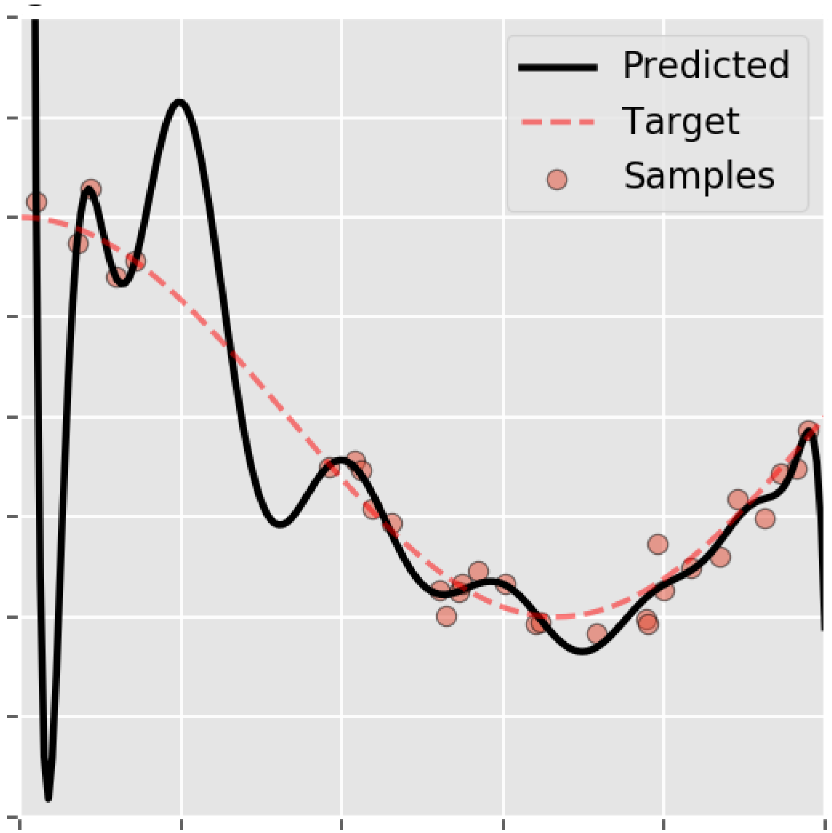
- **Idea.** Prevent parameters from getting too large by adding a term to the cost function due to penalizing large parameter values.





Regularization

- **Idea.** Prevent parameters from getting too large by adding a term to the cost function due to penalizing large parameter values.



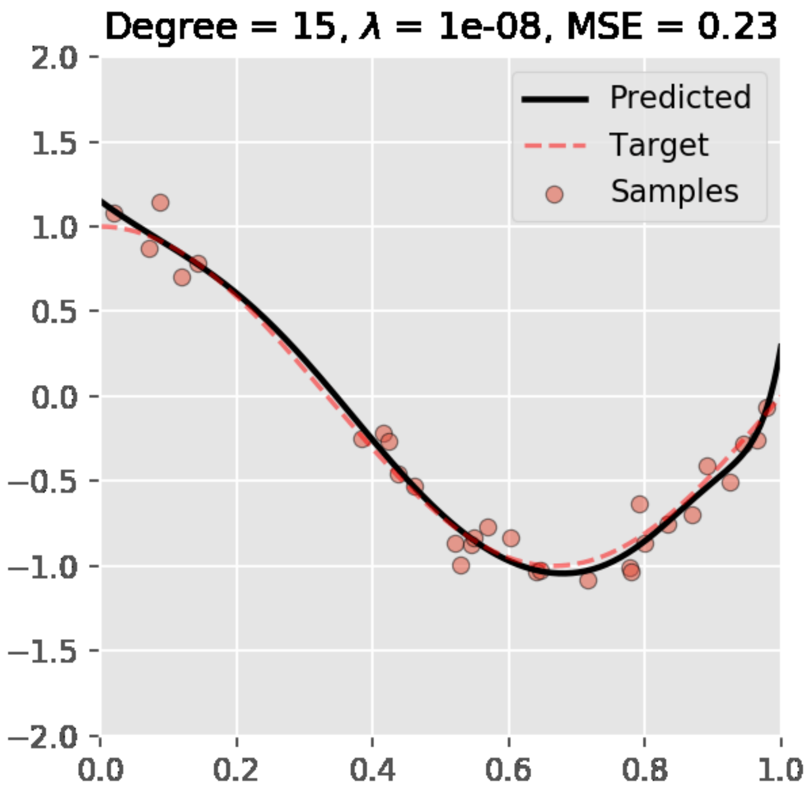
$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} cost(x^{(i)}, y^{(i)}) \quad + \quad \lambda R(\theta)$$

$$R(\theta) = \sum_{j=1}^{n} \theta_j^2 = \|\theta\|_2^2$$ L2 Regularization

$$R(\theta) = \sum_{j=1}^{n} |\theta_j| = \|\theta\|_1$$ L1 Regularization

Regularization

- **Idea.** Prevent parameters from getting too large by adding a term to the cost function due to penalizing large parameter values.



$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} cost(x^{(i)}, y^{(i)}) + \lambda R(\theta)$$

- **coefficient of regularization.** Establishing a balance between the above targets.

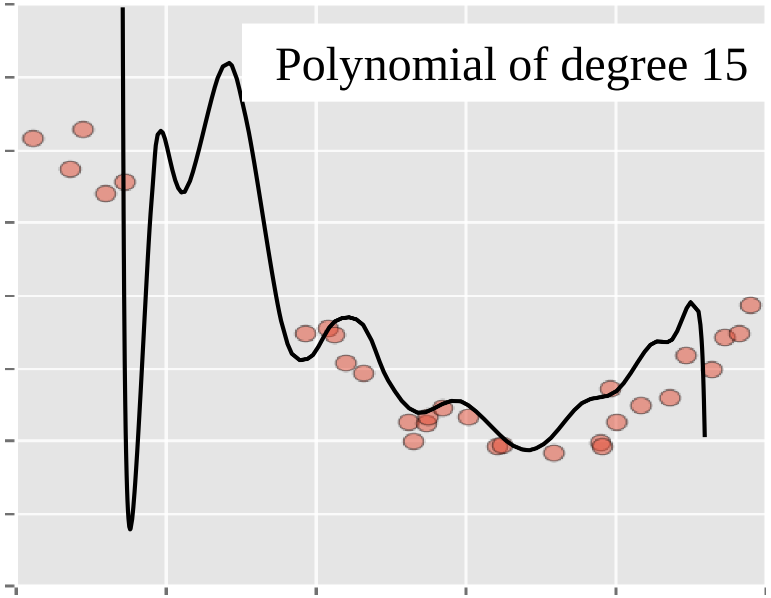
$\lambda \rightarrow 0$ ⇒ paying more attention to the error of the training collection(set)

$\lambda \rightarrow \infty$ ⇒ paying more attention to the error of generalization

Loss Function

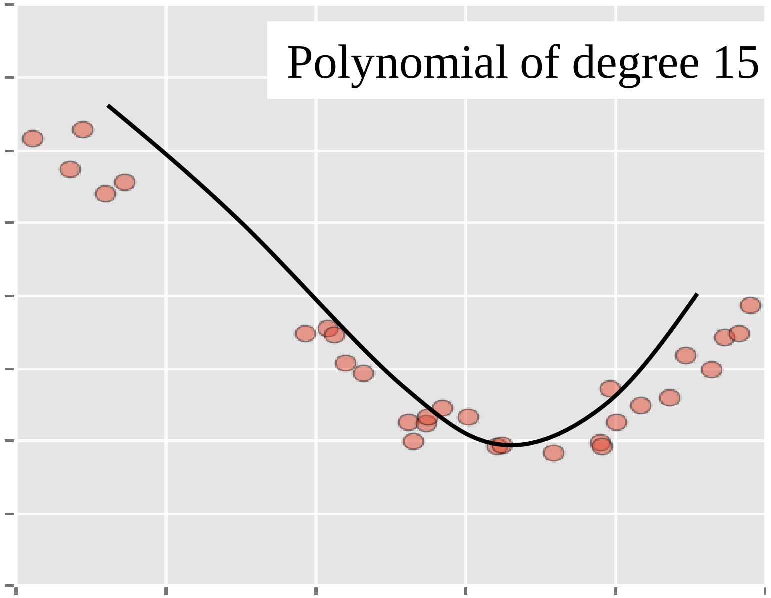
Regularization

Regression without Regularization



$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} cost(x^{(i)}, y^{(i)})$$

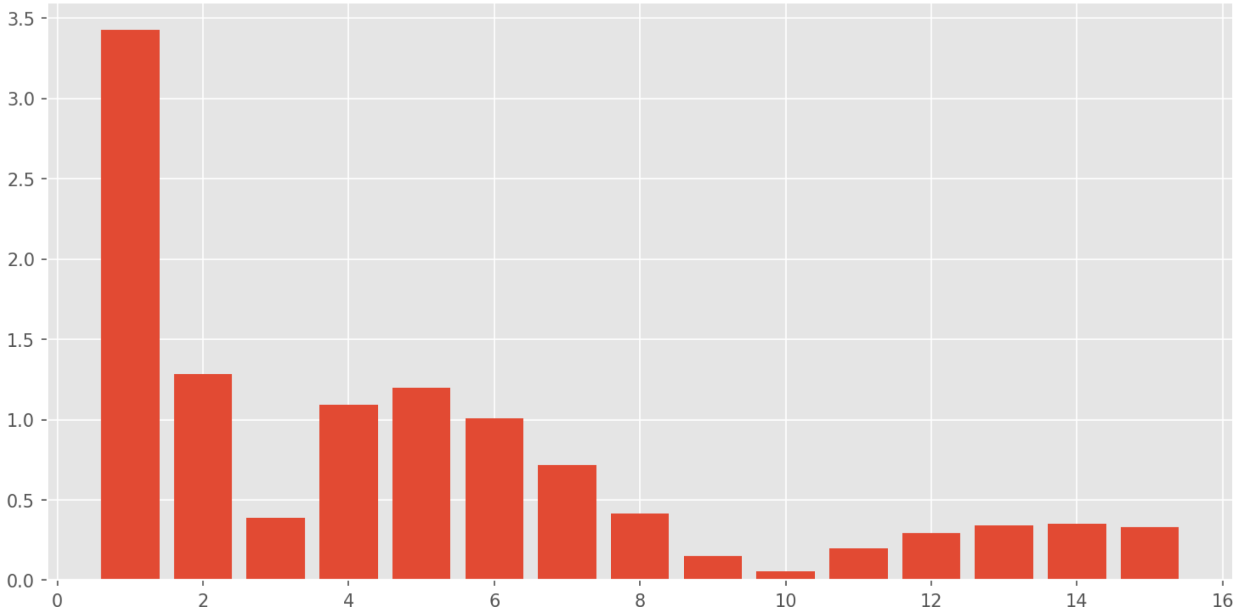
Regularized Regression



$$\frac{1}{m}\sum_{i=1}^{m} cost(x^{(i)}, y^{(i)}) + \lambda R(\theta)$$

Regularization

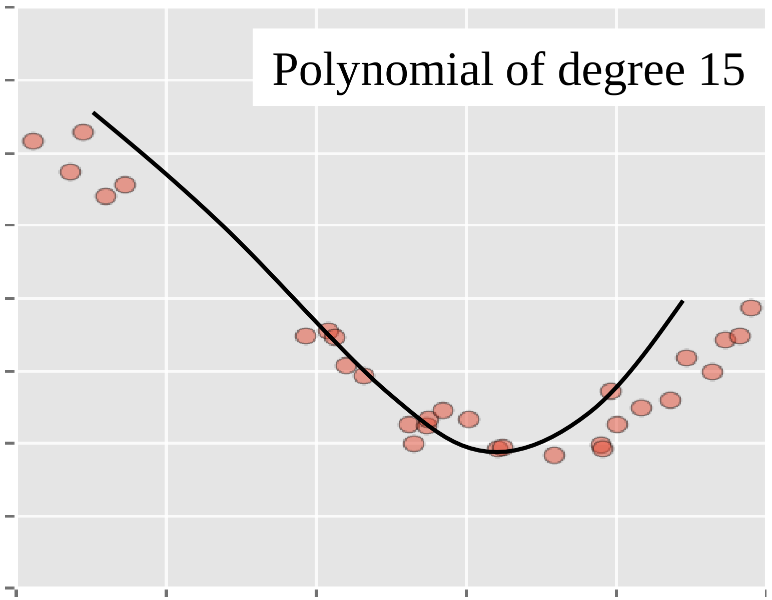
Value of parameters if regularization is used



$$R(\theta) = \sum_{j=1}^{n} \theta_j^2 = \|\theta\|_2^2$$

L2 Regularization

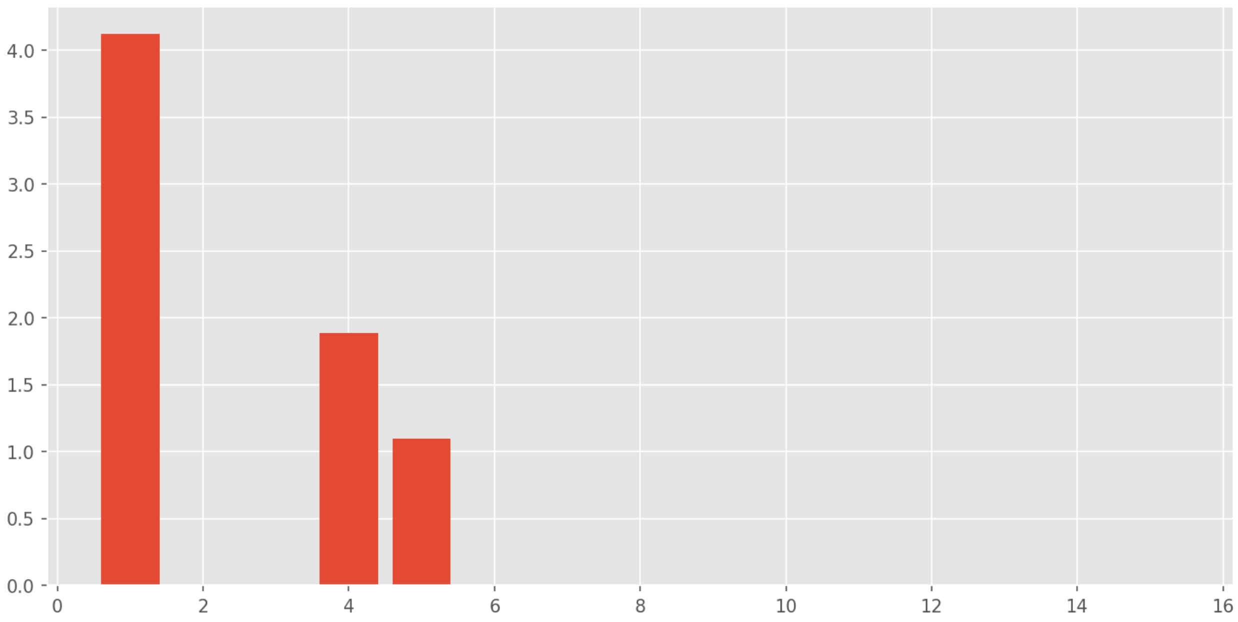
Regularized Regression



$$\frac{1}{m}\sum_{i=1}^{m} cost(x^{(i)}, y^{(i)}) + \lambda R(\theta)$$

Regularization

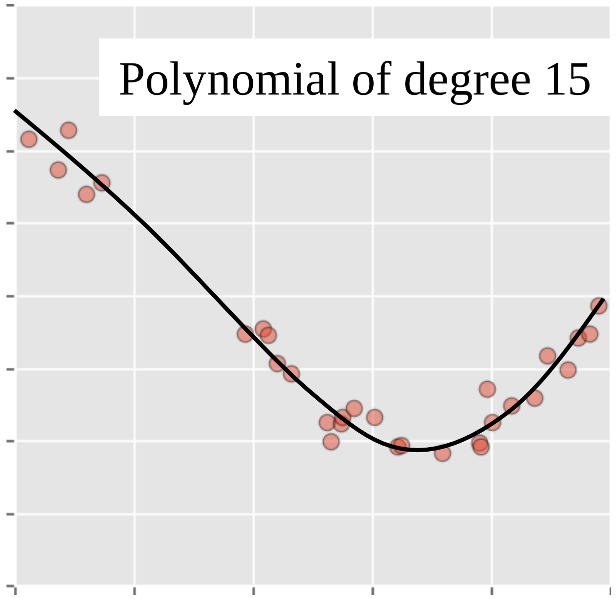
Value of parameters if regularization is used



$$R(\theta) = \sum_{j=1}^{n} |\theta_j| = \|\theta\|_1$$

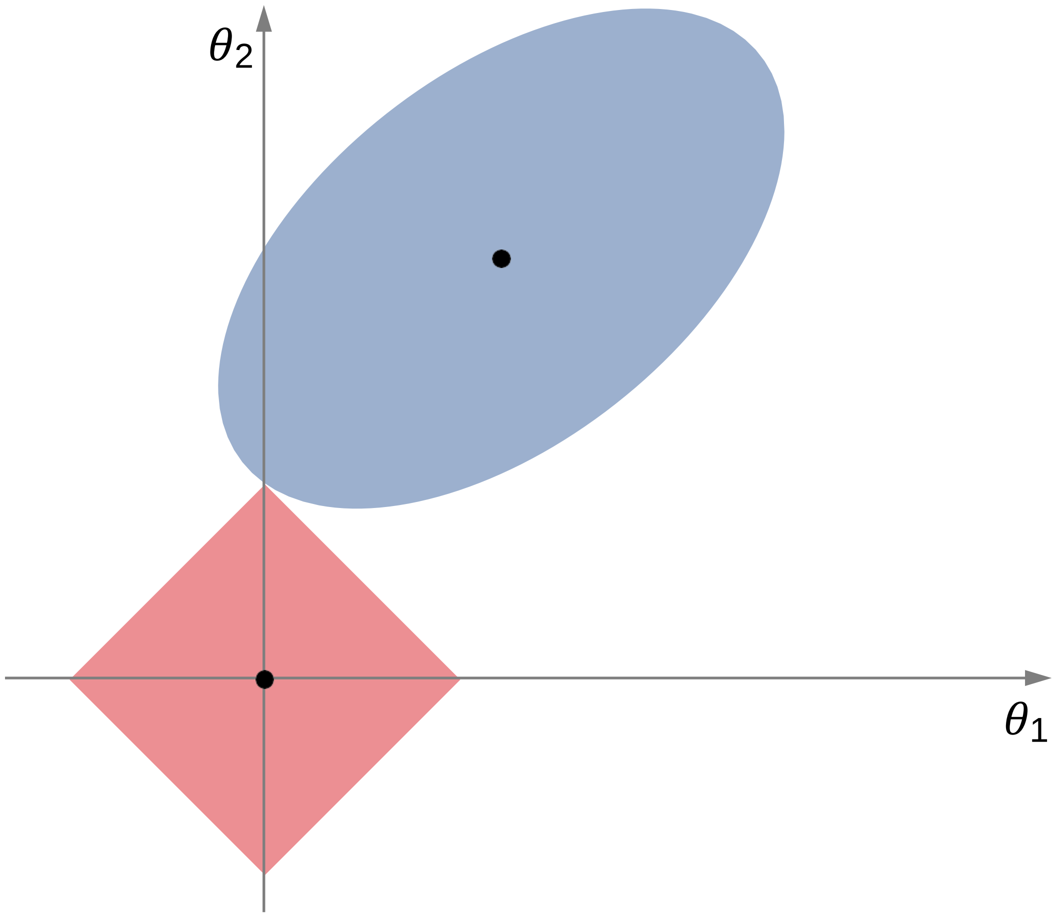
L1 Regularization

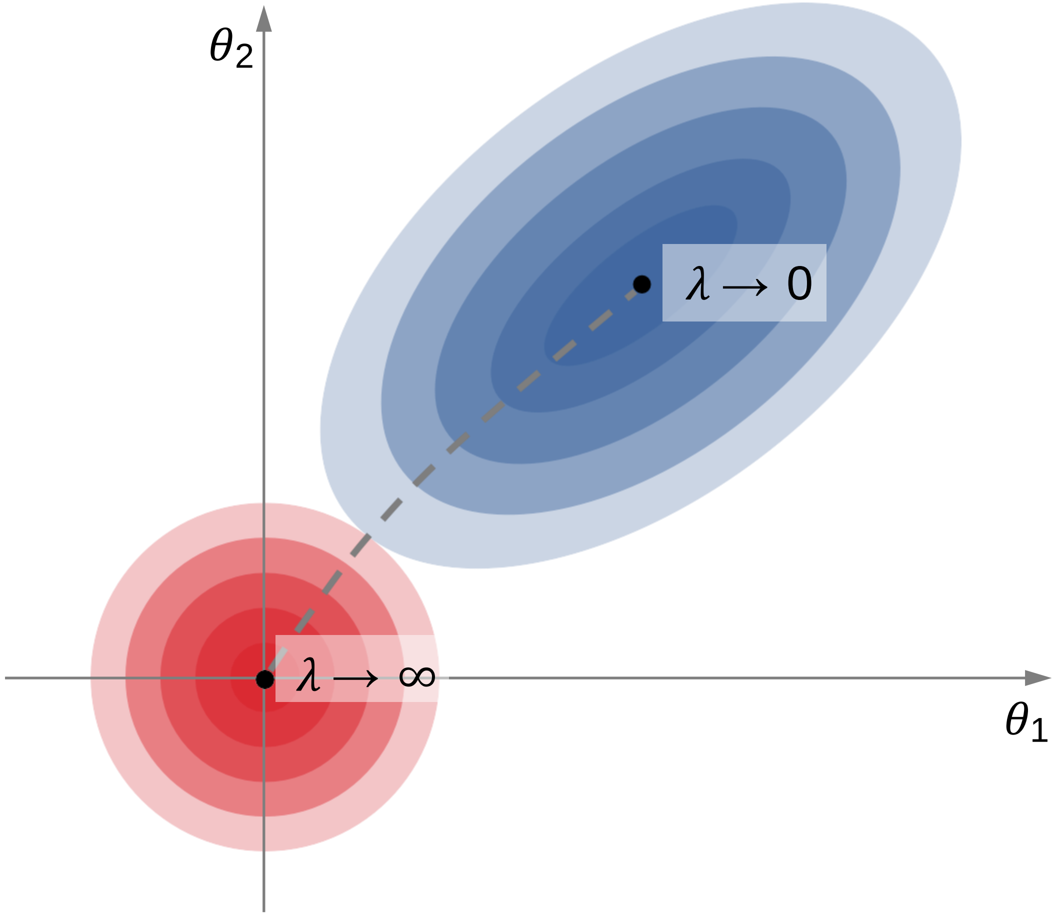
Regularized Regression



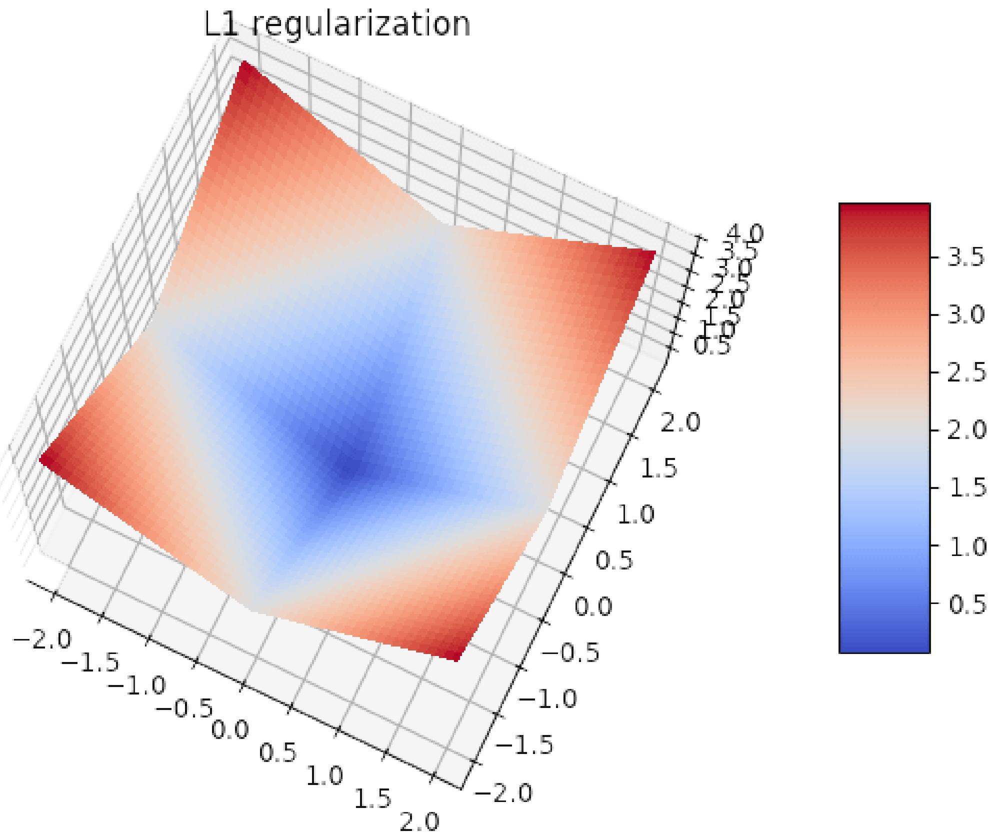
$$\frac{1}{m}\sum_{i=1}^{m} cost(x^{(i)}, y^{(i)}) + \lambda R(\theta)$$

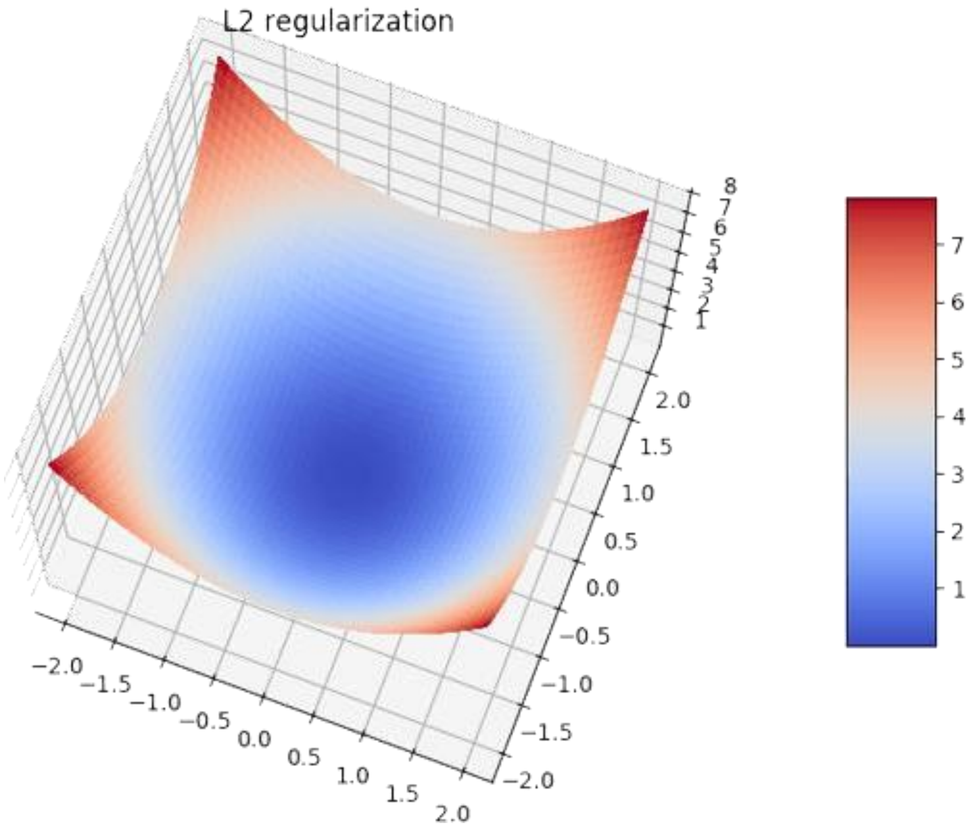
L1 & L2 Regularization





L1 & L2 Regularization





Regularized Linear Regression

Regularized Linear Regression

- Loss Function.

$$J(\theta) = \frac{1}{2}\left[\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 + \lambda\sum_{j=1}^{n}\theta_j^2\right] \quad = \frac{1}{2}(X\theta - y)^T(X\theta - y) + \frac{1}{2}\lambda\theta^T\theta$$

- Purpose. Minimization of the loss function in order to find the optimal value of the parameters

$$\min_{\theta} J(\theta)$$

Gradient Descent (Without Regularization)

- Without Regularization.

repeat until convergence {

$$\theta_j = \theta_j - \alpha \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right) \cdot x_j^{(i)} \qquad (\ j = 0,1,2,\ldots,n)$$

}

Gradient Descent (With Regularization)

- With Regularization

repeat until convergence {

$$\theta_0 = \theta_0 - \alpha \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_0^{(i)}$$

$$\theta_j = \theta_j - \alpha \left[\sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)} + \lambda\theta_j \right] \qquad (j = 1,2,\ldots,n)$$

}

$$\theta_j = \theta_j \underbrace{(1 - \alpha\lambda)}_{< 1} - \alpha \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)}$$

Normal Equation (With Regularization)

$$J(\theta) = \frac{1}{2}(X\theta - y)^T(X\theta - y) + \frac{1}{2}\lambda\theta^T\theta$$

$$\theta = (X^T X + \lambda I)^{-1} X^T y$$

($\lambda > 0$) reversible

$$\frac{\partial J}{\partial \theta} = X^T(X\theta - y) + \lambda\theta$$

$$= X^T X\theta - X^T y + \lambda\theta$$

$$= (X^T X + \lambda I)\theta - X^T = 0$$

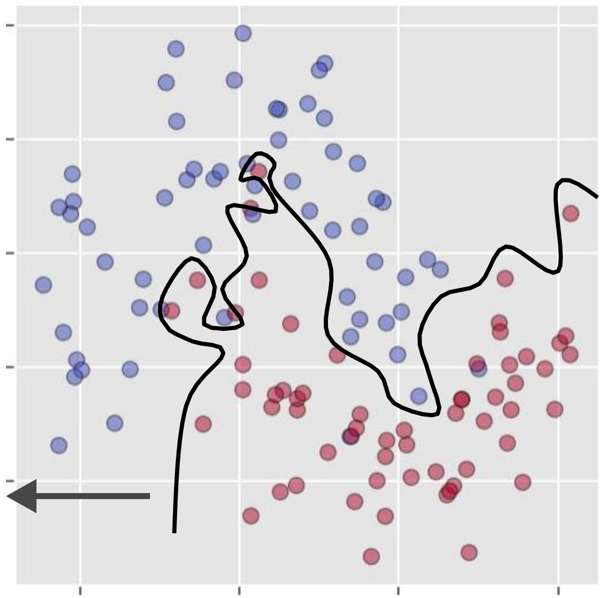
$$(X^T X + \lambda I)\theta = X^T y$$

$$\theta = \left(X^T X + \lambda \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \right)^{-1} X^T y$$

Regularized Logistic Regression

Logistic Regression (Without Regularization)

- Hypothesis(Supposition).



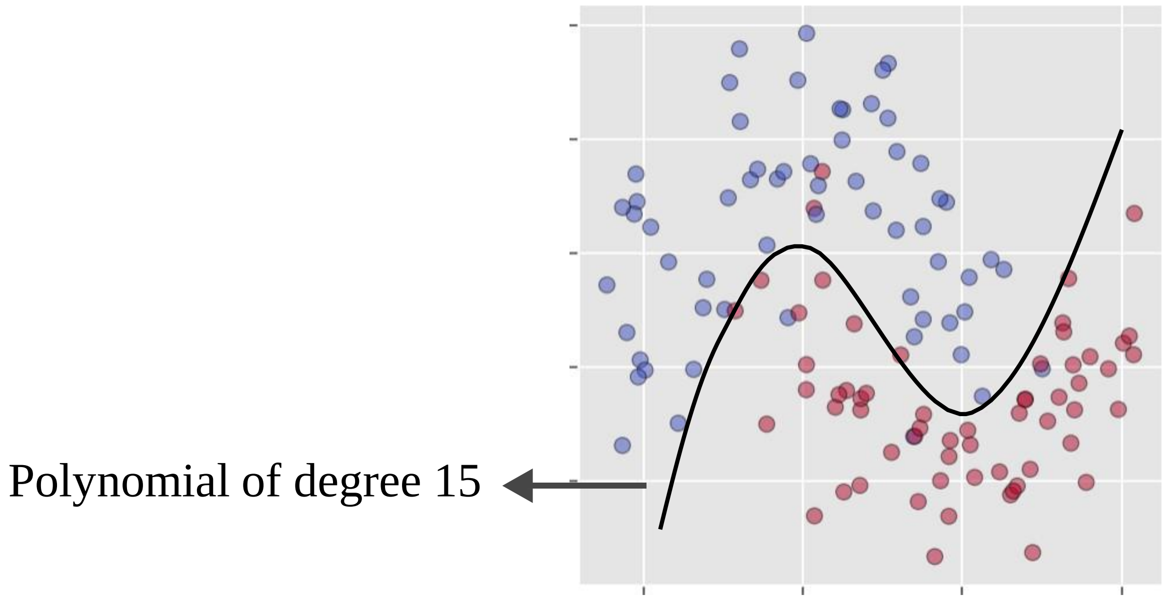
Polynomial of degree 15

- Loss Function.

$$J(\theta) = -\sum_{i=1}^{m} y^{(i)} \log h_\theta\left(x^{(i)}\right) + (1 - y^{(i)}) \log\left(1 - h_\theta\left(x^{(i)}\right)\right)$$

Logistic Regression (With Regularization)

- Hypothesis(Supposition).



- Loss Function.

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

Gradient Descent

- With Regularization.

repeat until convergence {

$$\theta_0 = \theta_0 - \alpha \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_0^{(i)}$$

$$\theta_j = \theta_j - \alpha \left[\sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)} + \lambda\theta_j \right] \qquad (j = 1,2,\ldots,n)$$

}

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

Advanced Optimization


```
from scipy.optimize import minimize


minimize(J, x0, method='CG', jac=grads)
```

Gradient

$$\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right) \cdot x_j^{(i)} + \lambda\theta_j$$

Implementation of the loss function

$$-\sum_{i=1}^{m} y^{(i)} \log h_\theta\left(x^{(i)}\right) + \left(1 - y^{(i)}\right) \log\left(1 - h_\theta\left(x^{(i)}\right)\right) + \frac{\lambda}{2}\sum_{j=1}^{n}\theta_j^2$$