Lecture 3

Outline

- formulating linear regression as optimization
- three approaches to solving opt problem
  - batch gradient descent
  - stochastic gradient descent
  - closed form solution
- probabilistic interpretation of linear regression
- expanding basis functions
- Regularization : bias vs variance tradeoff
- choosing regularization parameters
- uncertainty estimates on parameters

Given $D = \{(x^{(i)}, y^{(i)}) \mid 1 \leq i \leq m;\ x^{(i)} \in \mathbb{R}^d;\ y^{(i)} \in \mathbb{R}\}$

and class of linear functions $h : \mathbb{R}^{d+1} \mapsto \mathbb{R}$

$$h_\theta(x) = \theta^T x$$

where $x = [1, x_1, .., x_d)^T$; $\theta = (\theta_0, \theta_1, \ldots, \theta_d)^T$

Find h_θ that minimizes the empirical loss function

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

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \; J(\theta)$$

Assumptions

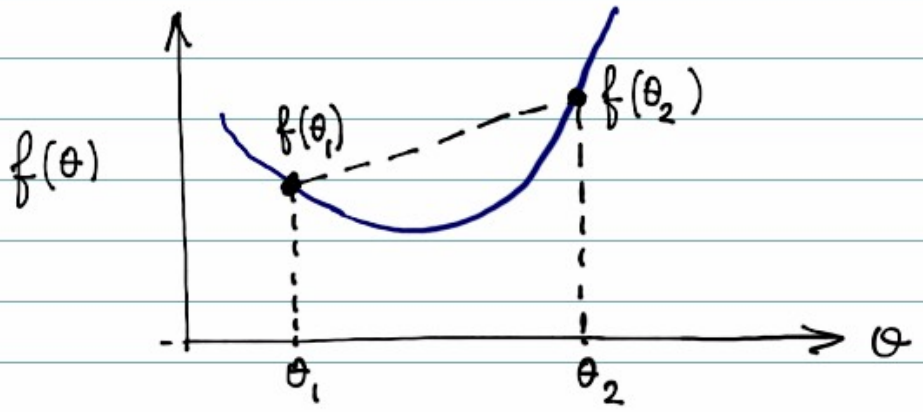
(1) h is a linear function of x parameterized by θ

(2) $\mathcal{D}$ is drawn iid and is a representative sample

How to find θ^*?

Approach #1 : batch gradient descent

1. $J(\theta)$ is convex (PROVE IT!)
2. $J(\theta)$ has a unique global minimum (why?)



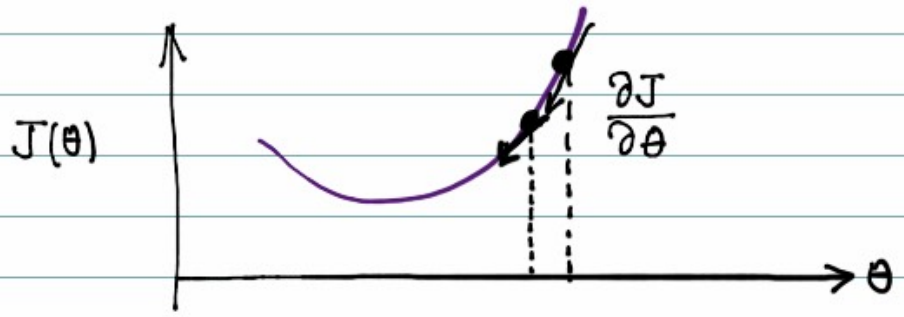
$f(\theta)$ is convex if

$$f(\lambda\theta_1 + (1-\lambda)\theta_2) \leq \lambda f(\theta_1) + (1-\lambda) f(\theta_2)$$

Gradient descent can be used to find minima of convex functions.

Gradient descent

- start with a random guess for θ
- repeatedly update θ by going in the direction of the steepest descent of $J(\theta)$



$$\theta \leftarrow \theta - \alpha . \frac{\partial J}{\partial \theta} \qquad \theta \in \mathbb{R}$$

For $\theta \in \mathbb{R}^{d+1}$

$$\theta_j \leftarrow \theta_j - \alpha . \frac{\partial J}{\partial \theta_j} \qquad j = 0, 1, \ldots, d$$

$\alpha > 0$ is called the step size or learning rate

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

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

So θ_j update rule is

$$\theta_j \leftarrow \theta_j + \frac{\alpha}{m} \sum_{i=1}^{m} \underbrace{\left(y^{(i)} - \theta^T x^{(i)}\right)}_{\text{error on example } i} \underbrace{\left(x_j^{(i)}\right)}_{j^{th} \text{ component of example } i}$$

Until θ converges

$$\theta_j \leftarrow \theta_j + \frac{\alpha}{m} \sum_{i=1}^{m} \left(y^{(i)} - \theta^T x^{(i)}\right) x_j^{(i)} \qquad j = 0, \ldots, d$$

Called batch gradient descent because updating θ requires summing over all m examples.

Stochastic gradient descent

- Update on the basis of a single example.
- Converges faster than batch descent when m is large.

Until θ converges:

for $i = 1$ to m:

$$\theta_j \leftarrow \theta_j + \frac{\alpha}{m}\left(y^{(i)} - \theta^T x^{(i)}\right) x_j^{(i)}$$

end

To make gradient descent work

- choose α carefully
- scale columns in x so that coefficients θ_j are more or less of the same order

Minibatch gradient descent: choose a small subset of examples instead of a single one

Closed form solution

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

can be rewritten in matrix notation

$$X = \begin{bmatrix} 1 \leftarrow & x^{(1)^T} & \rightarrow \\ \vdots & & \\ 1 \leftarrow & x^{(m)^T} & \rightarrow \end{bmatrix} \qquad y = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(m)} \end{bmatrix}$$

$m \times (d+1)$ $\qquad$ $m \times 1$

$\theta \in \mathbb{R}^{d+1}$

$\hat{y} = X\theta$ predictions

error $= X\theta - y$ ($m \times 1$ vector)

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

L2 norm of error vector!

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\ J(\theta)$$

$\nabla_\theta J(\theta) = 0$ set first derivative of J to 0 and solve for θ

$$\nabla_\theta J(\theta) = \frac{1}{2m} \frac{\partial}{\partial \theta} \left[\theta^T X^T X \theta + y^T y - \theta^T X^T y - y^T X \theta \right] = 0$$

$$2(X^T X)\theta - X^T y - X^T y = 0$$

$$2(X^T X)\theta = 2X^T y$$

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

This expression is called the normal equation and can be used to calculate $E[\theta]$ and $Var[\theta]$ under some assumptions (Problem 2, Homework 1)

Probabilistic interpretation of linear regression

Suppose $y^{(i)}$ comes from an underlying linear model

$$y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)} \; ; \; \varepsilon^{(i)} \sim N(0, \sigma^2)$$

Let $x^{(i)}, y^{(i)} \in \mathbb{R}$ for simplicity.

Then

$$P(\varepsilon^{(i)}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\varepsilon^{(i)})^2}{2\sigma^2}\right)$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}\right)$$

$$P((x^{(i)}, y^{(i)}) \mid \theta, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}\right)$$

This is the probability of observing $(x^{(i)}, y^{(i)})$ given generative model above.

Given $\mathcal{D} = \left\{ (x^{(i)}, y^{(i)}) \mid 1 \le i \le m ;\ x^{(i)} \in \mathbb{R} ;\ y^{(i)} \in \mathbb{R} \right\}$ and assuming elements of $\mathcal{D}$ are drawn i.i.d

$$P(\mathcal{D} \mid \theta ; \sigma^2) = \prod_{i=1}^{m} P((x^{(i)}, y^{(i)}) \mid \theta ; \sigma^2)$$

$$= \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(- \frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2} \right)$$

We use the principle of maximum likelihood to infer θ from $\mathcal{D}$

$$\theta^*_{MLE} = \underset{\theta}{\operatorname{argmax}}\ P(\mathcal{D} \mid \theta)$$

$$= \underset{\theta}{\operatorname{argmax}}\ \log P(\mathcal{D} \mid \theta)$$

$$= \underset{\theta}{\operatorname{argmin}}\ \underbrace{- \log P(\mathcal{D} \mid \theta)}_{\text{negative log likelihood function}}$$

$$\log P(\mathcal{D} \mid \theta; \sigma^2) = \sum_{i=1}^{m} \left[\log \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2} \right]$$

The θ containing terms in $-\log P(\mathcal{D} \mid \theta; \sigma^2)$

$$NLL(\theta) = \sum_{i=1}^{m} (y^{(i)} - \theta^T x^{(i)})^2$$

Now we see how $J(\theta)$ arises in the context of estimating θ from $\mathcal{D}$ assuming a linear generative model.

Basis function expansion

Consider $x, y \in \mathbb{R}$

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

$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$

$h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_p x^p$

} models are linear in θ.

We model this by mapping each x to $(x, x^2, .., x^p)$

$$x \longmapsto (x, x^2, \ldots, x^p)$$

$$\Phi : \mathbb{R} \to \mathbb{R}^p$$

Given $x \in \mathbb{R}$, construct $\Phi(x) \in \mathbb{R}^p$

$$X = \begin{bmatrix} 1 & x^{(1)} \\ \vdots & \vdots \\ 1 & x^{(m)} \end{bmatrix} \qquad \Phi(X) = \begin{bmatrix} 1 & x^{(1)} & (x^{(1)})^2 & \cdots & (x^{(1)})^p \\ \vdots & & & & \\ 1 & x^{(m)} & (x^{(m)})^2 & \cdots & (x^{(m)})^p \end{bmatrix}$$

$m \times 2$ $\qquad$ $m \times (p+1)$

Now we can find θ to minimize $J(\theta)$ by replacing X by $\Phi(X)$.

We can learn higher order models in x, which are linear in θ.

Φ is not limited to 1-dim x's.

$$\Phi : \mathbb{R}^d \to \mathbb{R}^p \quad \text{where } p > d$$

For example : $x \in \mathbb{R}^2$ can be mapped to $\mathbb{R}^6$

$$(x_1, x_2) \mapsto (1, x_1, x_2, x_1 x_2, x_1^2, x_2^2)$$

Models

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 + \theta_5 x_2^2$$

Batch gradient descent

$$\theta_j \leftarrow \theta_j + \frac{\alpha}{m} \sum_{i=1}^{m} \left(y^{(i)} - \theta^T \phi(x^{(i)})\right) \left[\phi(x^{(i)})\right]_j$$

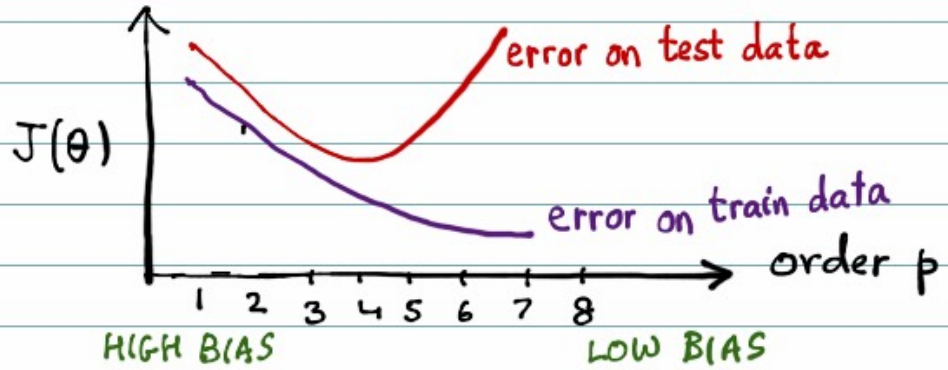
Stochastic gradient descent

$$\theta_j \leftarrow \theta_j + \frac{\alpha}{m} \left(y^{(i)} - \theta^T \phi(x^{(i)})\right) \left[\phi(x^{(i)})\right]_j$$

Normal equations

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

Controlling model complexity



Overfitted model : J(training data) < J(test data)

J = average mean squared error

If you examine θ for higher order models

- they have large absolute values
- unstable - small changes to D cause large changes to θ estimates (high variance)

Model selection : trading off bias & variance

HIGH BIAS : simpler model - lower variance in θ

LOW BIAS : complex model - higher variance in θ

One approach to automatically control model complexity is to penalize large θ_j

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left(y^{(i)} - \theta^T \phi(x^{(i)})\right)^2 + \underbrace{\frac{\lambda}{2m} \sum_{j=1}^{d} \theta_j^2}$$

We do not penalize θ_0

L2 regularization

Constrained optimization / Ridge regression

Solving $\operatorname{argmin}_{\theta} J(\theta)$

- Gradient descent $\theta_j \leftarrow \theta_j - \alpha \cdot \frac{\partial J(\theta)}{\partial \theta_j}$

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

$$+ \frac{\lambda}{m} \theta_j \qquad \text{for } j = 1, \ldots, d$$

Regular update for $\partial J(\theta)/\partial \theta_0$

Normal equation can be derived from

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

Now solve $\nabla_\theta J(\theta) = 0$

$$\frac{\partial}{\partial\theta}\left[\theta^T\phi^T\phi\theta + y^Ty - \theta^T\phi^Ty - y^T\phi\theta + \lambda\theta^T\theta\right]$$

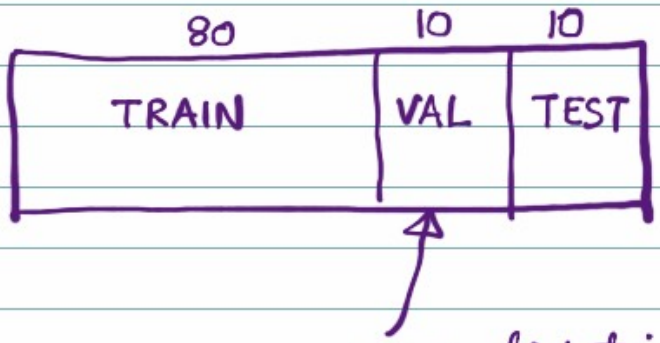
$$= 2\phi^T\phi\theta - \phi^Ty - \phi^Ty + 2\lambda I\theta = 0$$

$$\text{So} \quad \theta = (\phi^T\phi + \lambda I)^{-1}\phi^Ty$$

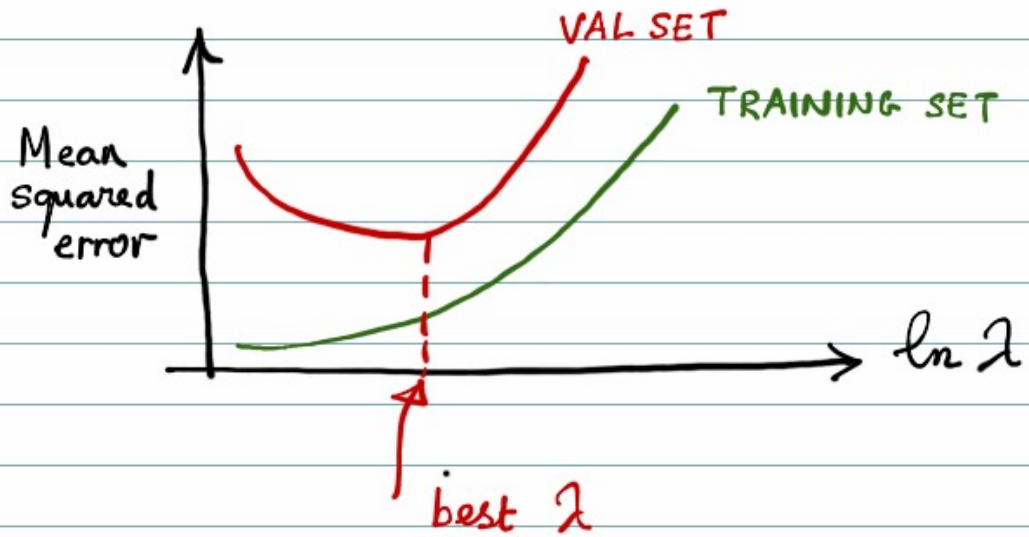
Regularization allows complex models to be trained on data sets of limited size without overfitting by limiting effective model complexity.

How to choose regularization parameter λ ?

— an empirical approach



use validation set to choose λ



Maximum aposteriori estimates of θ

$$p(\theta|\mathcal{D}) \propto \underbrace{p(\mathcal{D}|\theta)}_{\text{likelihood}} \underbrace{p(\theta)}_{\text{prior}}$$

We perform a Bayesian update on θ after observing the data $\mathcal{D}$.
To calculate $p(\mathcal{D}|\theta)$, we assume that the data is drawn from

$$y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)} \; ; \; \varepsilon^{(i)} \sim N(0, \sigma^2)$$

We assume a conjugate prior on θ

$$p(\theta) = N(\theta | 0, \alpha^2 I)$$

which is a multinomial Gaussian

$$p(\theta) = \left(\frac{1}{2\pi\alpha^2}\right)^{\frac{d+1}{2}} \exp\left(-\frac{\theta^T\theta}{2\alpha^2}\right)$$

We already know that

$$p(\mathcal{D}|\theta) = \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\left(y^{(i)} - \theta^T x^{(i)}\right)^2}{2\sigma^2}\right)$$

Now, $p(\theta|\mathcal{D}) \propto p(\mathcal{D}|\theta)\, p(\theta)$

which is

$$\left(\frac{1}{2\pi\alpha^2}\right)^{\frac{d+1}{2}} \exp\left(-\frac{\theta^T\theta}{2\alpha^2}\right) \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\left(y^{(i)} - \theta^T x^{(i)}\right)^2}{2\sigma^2}\right)$$

We now take log likelihoods and separate terms

$$\log p(\theta|\mathcal{D}) \propto \frac{d+1}{2}\log\frac{1}{2\pi\alpha^2} - \frac{1}{2\alpha^2}\theta^T\theta + \sum_{i=1}^{m}\log\frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2\sigma^2}\sum_{i=1}^{m}\left(y^{(i)} - \theta^T x^{(i)}\right)^2$$

The maximum aposteriori (MAP) estimate of θ

$$\begin{aligned}\theta_{MAP} &= \operatorname*{argmax}_{\theta} \log P(\theta|\mathcal{D}) \\ &= \operatorname*{argmax}_{\theta}\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{m}\left(y^{(i)} - \theta^T x^{(i)}\right)^2 - \frac{1}{2\alpha^2}\theta^T\theta\right] \\ &= \operatorname*{argmin}_{\theta}\left[\frac{1}{2\sigma^2}\sum_{i=1}^{m}\left(y^{(i)} - \theta^T x^{(i)}\right)^2 + \frac{1}{2\alpha^2}\theta^T\theta\right] \\ &= \operatorname*{argmin}_{\theta}\left[\frac{1}{2m}\sum_{i=1}^{m}\left(y^{(i)} - \theta^T x^{(i)}\right)^2 + \frac{\lambda}{2m}\theta^T\theta\right]\end{aligned}$$

where $\lambda = \sigma^2/\alpha^2$. This is exactly ridge regression!

Assessing confidence bands on θ

Frequentist perspective

- θ is a fixed parameter whose value is determined by an estimator
- error bars on this estimate are obtained by considering the distribution of possible data sets $\mathcal{D}$.
- Theoretical confidence bars can be derived with assumptions on generative process for $\mathcal{D}$
- Empirical confidence bars can be derived by bootstrap sampling

Bayesian perspective

- there is a single data set $\mathcal{D}$
- θ is a random variable with a prior $p(\theta)$
- after observing $\mathcal{D}$, we perform a Bayesian update on it to get $p(\theta|\mathcal{D})$.
- Since $p(\theta|\mathcal{D})$ is a distribution, we can derive its mean and variance

Frequentist perspective

Assume

$$y^{(i)} = \theta^{*T} x^{(i)} + \varepsilon^{(i)}$$
$$\varepsilon^{(i)} \sim N(0, \sigma^2)$$

Our estimator

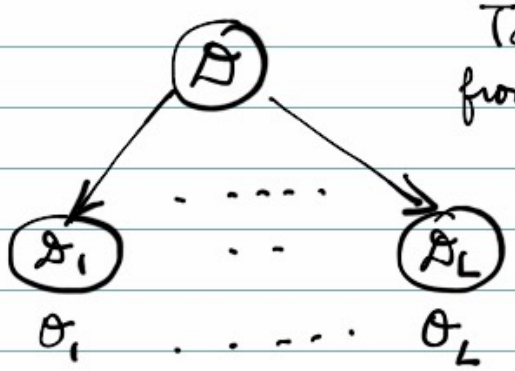
$$\theta = (X^T X)^{-1} X^T y$$

In HW1, we calculate

$$E(\theta) = \theta^*$$
$$Var(\theta) = (X^T X)^{-1} \sigma^2$$

Bootstrap sampling

$|\mathcal{D}| = m$



Take m samples from $\mathcal{D}$ with replacement

L bootstrap samples of $\mathcal{D}$

$E(\theta)$; $Var(\theta)$ from $\theta_1 \dots \theta_L$

Global vs local models of regression

$$\mathcal{D} = \left\{ (x^{(i)}, y^{(i)}) \mid 1 \leq i \leq m ;\ x^{(i)} \in \mathbb{R}^d ;\ y \in \mathbb{R} \right\}$$

all points in $\mathcal{D}$ have the same influence or weight on cost function $J(\theta)$

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

$$\theta_{MLE} = \operatorname{argmin}_{\theta} J(\theta)$$

This is a global model. We learn θ_{MLE} and we can throw $\mathcal{D}$ away. All future predictions are basically $\theta_{MLE}^T x$

In HW1, you see an interesting variation with locally weighted regression

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

where we have a weighting function

$$w^{(i)} = \exp\left(- \frac{(x - x^{(i)})^T (x - x^{(i)})}{2\sigma^2} \right)$$

Weight of $x^{(i)}$ is a function of point x where we seek new prediction. Parameter σ defines the sphere of influence around x

Demo here