CSC 4020 Fundamentals of Machine Learning: Linear Regression

Baoyuan Wu
School of Data Science, CUHK-SZ

February 22/24, 2021

Outline

Linear regression: deterministic perspective

Linear regression: deterministic perspective

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

- **Linear hypothesis function**: $h_{\boldsymbol{\theta}} = \phi(\boldsymbol{x})^\top \boldsymbol{\theta}$

$$\phi(x) = \begin{bmatrix} x_1 \\ x_2 \\ x_1 x_2 \\ x_1^2 \end{bmatrix}$$

Linear regression: deterministic perspective

- **Linear hypothesis function**: $h_{\boldsymbol{\theta}} = \phi(\boldsymbol{x})^\top \boldsymbol{\theta}$
- **Linear regression** by minimizing residual sum of squares (RSS):

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^{m} (\phi(\boldsymbol{x})_i^\top \boldsymbol{\theta} - y_i)^2$$

Linear regression: deterministic perspective

- **Linear hypothesis function**: $h_{\boldsymbol{\theta}} = \phi(\boldsymbol{x})^\top \boldsymbol{\theta}$
- **Linear regression** by minimizing residual sum of squares (RSS):

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^{m} (\phi(\boldsymbol{x})_i^\top \boldsymbol{\theta} - y_i)^2$$

- **Two solutions**: gradient descent and close-form solution (called normal equation or ordinary least squares solution)

$$(\phi(X)^\top \theta - y) \cdot \phi(X) = 0$$

$$\phi^\top(X)\phi(X)\,\theta = y\,\phi(X)$$

$$y$$

$$\begin{bmatrix} \vdots \\ \phi(X)_m \theta \end{bmatrix} - \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$$

Linear regression: probabilistic perspective

Linear regression: probabilistic perspective

- We assume that: $y = \boldsymbol{\theta}^\top \boldsymbol{x} + e$, where $e \sim \mathcal{N}(0, \sigma^2)$ is called **observation noise** or **residual error**

Linear regression: probabilistic perspective

- We assume that: $y = \boldsymbol{\theta}^\top \boldsymbol{x} + e$, where $e \sim \mathcal{N}(0, \sigma^2)$ is called **observation noise** or **residual error**
- y is also a random variable, and its conditional probability is

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}^\top \boldsymbol{x}, \sigma^2)$$

Linear regression: probabilistic perspective

- We assume that: $y = \boldsymbol{\theta}^\top \boldsymbol{x} + e$, where $e \sim \mathcal{N}(0, \sigma^2)$ is called **observation noise** or **residual error**
- y is also a random variable, and its conditional probability is

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}^\top \boldsymbol{x}, \sigma^2)$$

- **Maximum log-likelihood estimation**:

$$\begin{aligned}
\boldsymbol{\theta}_{MLE} &= \arg\max_{\boldsymbol{\theta}} \log \mathcal{L}(\boldsymbol{\theta}|D) && (1)\\
&= \sum_i^m \log p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \sum_i^m \log \mathcal{N}(\boldsymbol{\theta}^\top \boldsymbol{x}, \sigma^2) && (2)\\
&= -\log(\sigma^m (2\pi)^{\frac{m}{2}}) - \frac{1}{2\sigma^2} \sum_i^m (y_i - \boldsymbol{\theta}^\top \boldsymbol{x}_i) && (3)\\
&= \arg\min_{\boldsymbol{\theta}} \frac{1}{2} \sum_i^m (y_i - \boldsymbol{\theta}^\top \boldsymbol{x}_i)^2, && (4)
\end{aligned}$$

Variants of linear regression

- **Robust regression** for data with outliers: $\boldsymbol{\theta}_{MLE} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{m} |\bar{\boldsymbol{x}}_i^\top \boldsymbol{\theta} - y_i|$
- **Ridge regression** to avoid over-fitting, through MAP estimation:

$$\boldsymbol{\theta}_{MAP} = \arg\max_{\boldsymbol{\theta}} \sum_{i}^{m} \log p(y|\boldsymbol{x}, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) \tag{5}$$

$$= \sum_{i}^{m} \log \mathcal{N}(\boldsymbol{\theta}^\top \boldsymbol{x}, \sigma^2) + \mathcal{N}(\boldsymbol{\theta}|\mathbf{0}, \tau^2 \mathbf{I}) \tag{6}$$

$$\equiv \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{m} (\bar{\boldsymbol{x}}_i^\top \boldsymbol{\theta} - y_i)^2 + \lambda \|\boldsymbol{\theta}\|_2^2. \tag{7}$$

- **Lasso regression** to obtain sparse model,

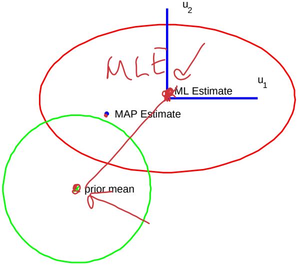
$$\boldsymbol{\theta}_{MAP} = \arg\max_{\boldsymbol{\theta}} \sum_{i}^{m} \log \mathcal{N}(\boldsymbol{\theta}^\top \boldsymbol{x}, \sigma^2) + \text{Lap}(\boldsymbol{\theta}|\mathbf{0}, b) \tag{8}$$

$$= \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{m} (\bar{\boldsymbol{x}}_i^\top \boldsymbol{\theta} - y_i)^2 + \lambda |\boldsymbol{\theta}|. \tag{9}$$

Summary of different linear regressions

Note that the uniform distribution will not change the mode of the likelihood. Thus, MAP estimation with a uniform prior corresponds to MLE.

$p(y\|\boldsymbol{x}, \boldsymbol{\theta})$	$p(\boldsymbol{\theta})$	regression method
Gaussian	Uniform	Least squares
Gaussian	Gaussian	Ridge regression
Gaussian	Laplace	Lasso regression
Laplace	Uniform	Robust regression
Student	Uniform	Robust regression



Generalized linear regression

- **Generalized linear model (GLM)**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = g^{-1}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \; y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), \tag{10}$$

  where g is called **link function**.

Generalized linear regression

- **Generalized linear model (GLM)**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = g^{-1}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})),\ y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), \tag{10}$$

  where g is called **link function**.
- We assume that the conditional probability follows

$$P(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}, N) = \text{Bin}(y_i|N, \mu_i) = \binom{N}{y_i} \mu_i^{y_i} (1-\mu_i)^{N-y_i}, \tag{11}$$

- Where $\mu_i = \frac{1}{1+e^{-\boldsymbol{\theta}^\top \boldsymbol{x}_i}}, g(\mu_i) = \ln \frac{\mu_i}{1-\mu_i}$.

Generalized linear regression

- **Generalized linear model (GLM)**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = g^{-1}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \; y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), \tag{10}$$

  where g is called **link function**.
- We assume that the conditional probability follows

$$P(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}, N) = \text{Bin}(y_i|N, \mu_i) = \binom{N}{y_i} \mu_i^{y_i} (1-\mu_i)^{N-y_i}, \tag{11}$$

- Where $\mu_i = \frac{1}{1+e^{-\boldsymbol{\theta}^\top \boldsymbol{x}_i}}, g(\mu_i) = \ln \frac{\mu_i}{1-\mu_i}$.
- The log-likelihood function is formulated as follows

$$\log \mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{m} \log P(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}) = y_i \log \mu_i + (N - y_i) \log(1-\mu_i) \tag{12}$$

Generalized linear regression

- **Generalized linear model (GLM)**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = g^{-1}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \ y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), \tag{10}$$

  where g is called **link function**.
- We assume that the conditional probability follows

$$P(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}, N) = \text{Bin}(y_i|N, \mu_i) = \binom{N}{y_i} \mu_i^{y_i} (1-\mu_i)^{N-y_i}, \tag{11}$$

- Where $\mu_i = \frac{1}{1+e^{-\boldsymbol{\theta}^\top \boldsymbol{x}_i}}, g(\mu_i) = \ln \frac{\mu_i}{1-\mu_i}$.
- The log-likelihood function is formulated as follows

$$\log \mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{m} \log P(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}) = y_i \log \mu_i + (N - y_i) \log(1 - \mu_i) \tag{12}$$

- We have

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{i=1}^{m} (y_i - N\mu_i)\boldsymbol{x}_i = 0 \quad \Rightarrow \frac{y_i}{N} = \mu_i = \boxed{\frac{1}{1+e^{-\boldsymbol{\theta}^\top \boldsymbol{x}_i}}}. \tag{13}$$

Generalized linear regression

- **Generalized linear model (GLM)**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = g^{-1}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})),\ y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), \tag{10}$$

  where g is called **link function**.
- We assume that the conditional probability follows

$$P(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}, N) = \text{Bin}(y_i|N, \mu_i) = \binom{N}{y_i} \mu_i^{y_i}(1-\mu_i)^{N-y_i}, \tag{11}$$

- Where $\mu_i = \frac{1}{1+e^{-\boldsymbol{\theta}^\top \boldsymbol{x}_i}}, g(\mu_i) = \ln \frac{\mu_i}{1-\mu_i}$.
- The log-likelihood function is formulated as follows

$$\log \mathcal{L}(\boldsymbol{\theta}) = \sum_{i=1}^{m} \log P(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}) = y_i \log \mu_i + (N - y_i) \log(1 - \mu_i) \tag{12}$$

- We have

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{i=1}^{m} (y_i - N\mu_i)\boldsymbol{x}_i = 0 \quad \Rightarrow \quad \frac{y_i}{N} = \mu_i = \frac{1}{1+e^{-\boldsymbol{\theta}^\top \boldsymbol{x}_i}}. \tag{13}$$

- Since the $\sigma(a) = \frac{1}{1+e^{-a}}$ is called **sigmoid function** or **logit function**, the above model is called **logit regression** or **logistic regression**.

Summary of last week

- Linear model is the linear function of the parameter $\boldsymbol{\theta}$, rather than the input feature

Summary of last week

- Linear model is the linear function of the parameter $\boldsymbol{\theta}$, rather than the input feature
- Linear model is a special case of generalized linear model, while generalized linear model is not always linear

Summary of last week

- Linear model is the linear function of the parameter $\boldsymbol{\theta}$, rather than the input feature
- Linear model is a special case of generalized linear model, while generalized linear model is not always linear
- Choosing different linear models is equivalent to choosing different distributions of $p(y|\boldsymbol{x}, \boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$, according to the task and the data

Classification

- Classification: classifying input data into discrete states

Classification

- Classification: classifying input data into discrete states
  - Email filtering: spam / not spam?

Classification

- Classification: classifying input data into discrete states
  - Email filtering: spam / not spam?
  - Whether forecast: sunny / not sunny?

Classification

- Classification: classifying input data into discrete states
  - Email filtering: spam / not spam? $\longrightarrow \{0, 1\}$
  - Whether forecast: sunny / not sunny?
  - Tumor: malignant / not malignant?

Classification

- Classification: classifying input data into discrete states
  - Email filtering: spam / not spam?
  - Whether forecast: sunny / not sunny?
  - Tumor: malignant / not malignant?
- The label $y \in \{0, 1\}$:

Classification

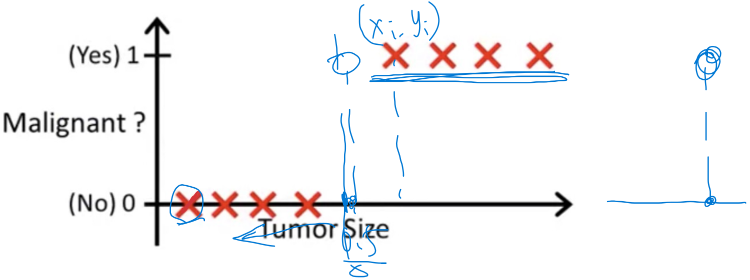
- Classification: classifying input data into discrete states
  - Email filtering: spam / not spam?
  - Whether forecast: sunny / not sunny?
  - Tumor: malignant / not malignant?
- The label $y \in \{0, 1\}$:
  - $y = 0$: negative class, *e.g.*, not spam, not sunny, not malignant

Classification

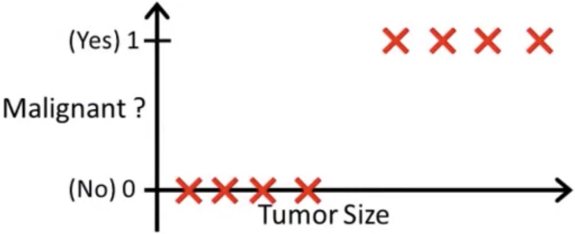
- Classification: classifying input data into discrete states
  - Email filtering: spam / not spam?
  - Whether forecast: sunny / not sunny?
  - Tumor: malignant / not malignant?
- The label $y \in \{0, 1\}$:
  - $y = 0$: negative class, *e.g.*, not spam, not sunny, not malignant
  - $y = 1$: positive class, *e.g.*, spam, sunny, malignant

Classification

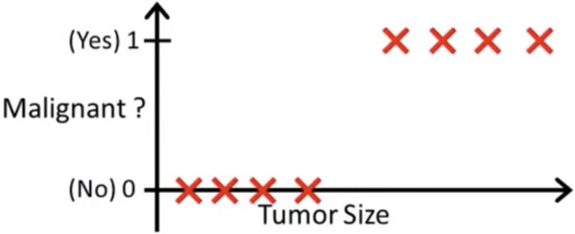
- Classification: classifying input data into discrete states
  - Email filtering: spam / not spam?
  - Whether forecast: sunny / not sunny?
  - Tumor: malignant / not malignant?
- The label $y \in \{0, 1\}$:
  - $y = 0$: negative class, *e.g.*, not spam, not sunny, not malignant
  - $y = 1$: positive class, *e.g.*, spam, sunny, malignant



Threshold classifier with linear regression

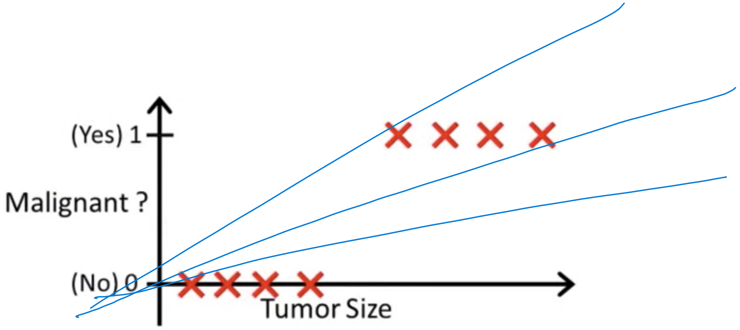


Threshold classifier with linear regression



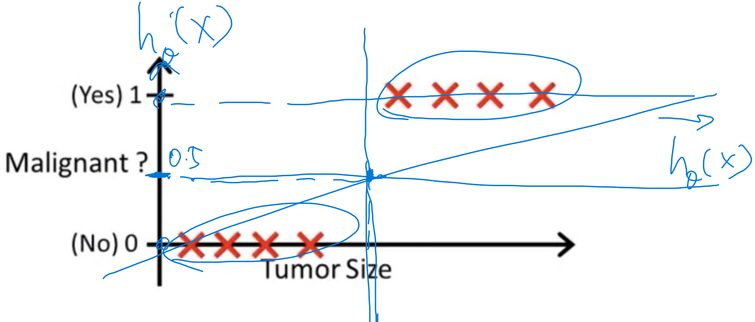
- We assume a linear hypothesis function $h_{\boldsymbol{\theta}} = \underbrace{\phi(\boldsymbol{x})^\top \boldsymbol{\theta}}_{X}$

Threshold classifier with linear regression



- We assume a linear hypothesis function $h_{\boldsymbol{\theta}} = \phi(\boldsymbol{x})^\top \boldsymbol{\theta}$
- A simple threshold classifier with this hypothesis function is

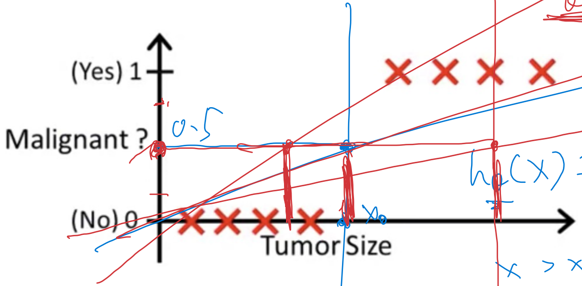
Threshold classifier with linear regression



- We assume a linear hypothesis function $h_{\boldsymbol{\theta}} = \phi(\boldsymbol{x})^\top \boldsymbol{\theta}$
- A simple threshold classifier with this hypothesis function is
  - If $h_{\boldsymbol{\theta}} > 0.5$, then $y = 1$, *i.e.*, malignant tumor

Threshold classifier with linear regression

$(h_\theta(x) - y)^2$



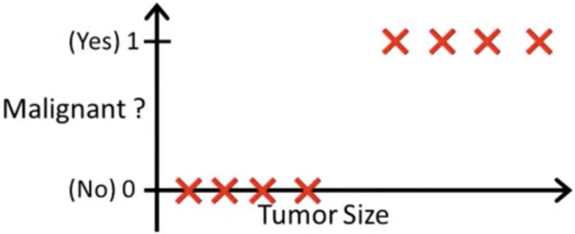
$h_\theta(x) = 0.5$

$x > x_0$, $y = 1$

$< x_0$, $y = 0$

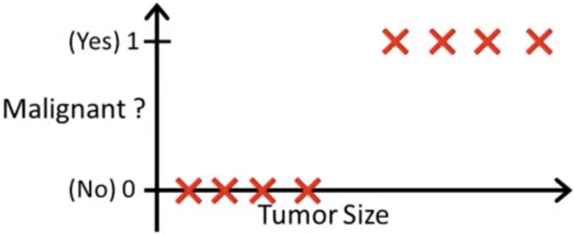
- We assume a linear hypothesis function $h_{\boldsymbol{\theta}} = \phi(\boldsymbol{x})^\top \boldsymbol{\theta}$
- A simple threshold classifier with this hypothesis function is
  - If $h_{\boldsymbol{\theta}} > 0.5$, then $y = 1$, *i.e.*, malignant tumor
  - If $h_{\boldsymbol{\theta}} < 0.5$, then $y = 0$, *i.e.*, benign tumor

Threshold classifier with linear regression



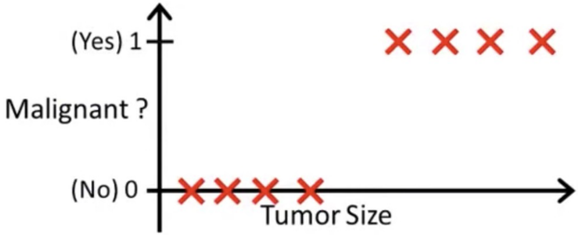
- It seems that the simple threshold classifier with linear regression works well on this classification task

Threshold classifier with linear regression



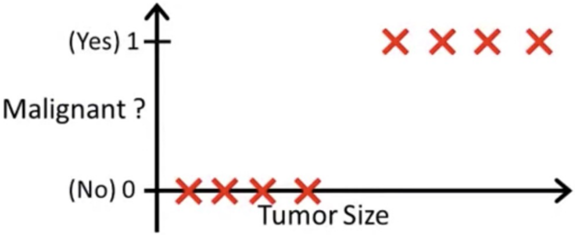
- It seems that the simple threshold classifier with linear regression works well on this classification task
- However, if there is a positive sample with very large tumor size (plot above), what will happen?

Threshold classifier with linear regression



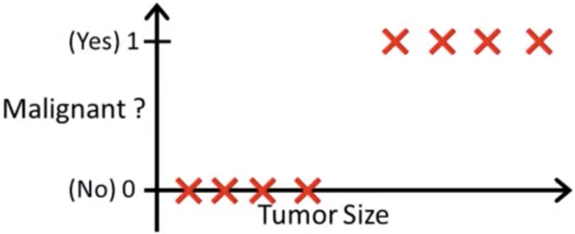
- It seems that the simple threshold classifier with linear regression works well on this classification task
- However, if there is a positive sample with very large tumor size (plot above), what will happen?
- The hypothesis function will be significantly changed, causing that some positive samples are mis-classified as negative (not malignant)?

Threshold classifier with linear regression



- It seems that the simple threshold classifier with linear regression works well on this classification task
- However, if there is a positive sample with very large tumor size (plot above), what will happen?
- The hypothesis function will be significantly changed, causing that some positive samples are mis-classified as negative (not malignant)? How to handle it?

Threshold classifier with linear regression



- It seems that the simple threshold classifier with linear regression works well on this classification task
- However, if there is a positive sample with very large tumor size (plot above), what will happen?
- The hypothesis function will be significantly changed, causing that some positive samples are mis-classified as negative (not malignant)? How to handle it? Adjusting the threshold value, or adopting robust linear regression.

Threshold classifier with linear regression

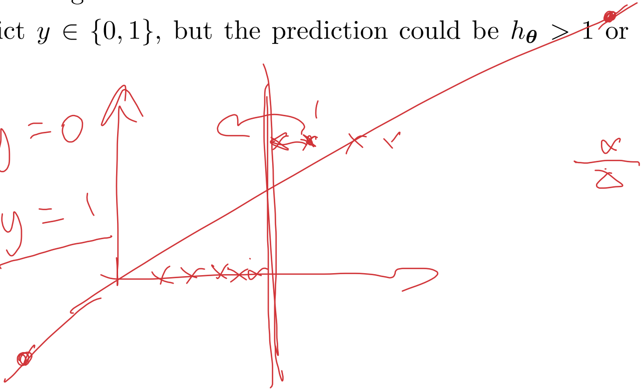
- But there is still something wired.

Threshold classifier with linear regression

- But there is still something wired.
- Our goal is to predict $y \in \{0, 1\}$, but the prediction could be $h_{\boldsymbol{\theta}} > 1$ or $h_{\boldsymbol{\theta}} < 0$

$h_\theta < t, \; y = 0$

$h_\theta > t, \; y = 1$



Threshold classifier with linear regression

- But there is still something wired.
- Our goal is to predict $y \in \{0, 1\}$, but the prediction could be $h_{\boldsymbol{\theta}} > 1$ or $h_{\boldsymbol{\theta}} < 0$
- It cannot reflect the difference among samples within the same class
- An expected hypothesis function for this task should be $h_{\boldsymbol{\theta}} \in [0, 1]$

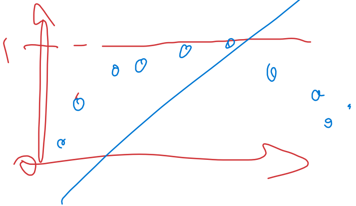
Threshold classifier with linear regression

Which statements are true?

Threshold classifier with linear regression

Which statements are true?

- If linear regression doesn't work well like the above example, feature scaling may help
- If the training set satisfies that all $y^{(i)} \in [0, 1]$ for all points $(\boldsymbol{x}, y^{(i)})$, then the linear hypothesis function $h_{\boldsymbol{\theta}} \in [0, 1]$ for all values of $\boldsymbol{x}$
- None of above two states are true



Hypothesis representation

- An expected hypothesis function for this task should be $h_{\boldsymbol{\theta}} \in [0, 1]$

- An expected hypothesis function for this task should be $h_{\boldsymbol{\theta}} \in [0, 1]$
- Recall the generalized linear regression that

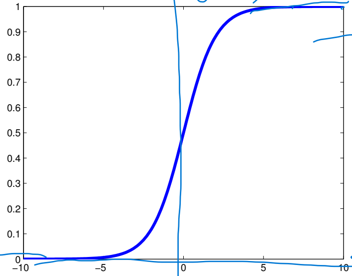
$g^{-1}(\quad)$ $\to 0$

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\boldsymbol{\theta}^\top \boldsymbol{x}) \in [0, 1], \quad g(z) = \frac{1}{1 + \exp(-z)},$$

u $\quad \ln \frac{u}{1-u}$

where $g(\cdot)$ is called **sigmoid function** or **logistic regression**. (Plot below)

$g(z)$



$-\inf$ $\quad z$

Hypothesis representation

- Interpretation of logistic function:

Hypothesis representation

- Interpretation of logistic function:
- $h_{\boldsymbol{\theta}}(\boldsymbol{x})$ = estimated probability that $y = 1$ of input $\boldsymbol{x}$

Hypothesis representation

- Interpretation of logistic function:
- $h_{\boldsymbol{\theta}}(\boldsymbol{x})$ = estimated probability that $y = 1$ of input $\boldsymbol{x}$
- For example (plot below), if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0.8$, then it means that an patient with tumor size x has 80% chance of tumor being malignant

$$h_\theta(x) = \frac{1}{1+e(-\theta^T x)} = p(y=1 \mid x;\theta)$$

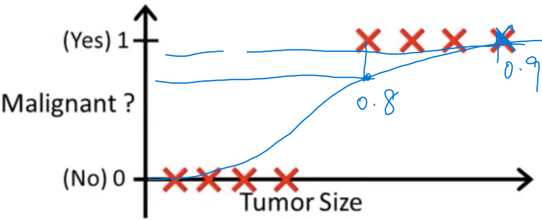
Hypothesis representation

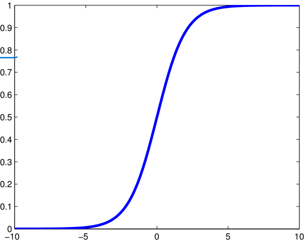
- Interpretation of logistic function:
- $h_{\boldsymbol{\theta}}(\boldsymbol{x})$ = estimated probability that $y = 1$ of input $\boldsymbol{x}$
- For example (plot below), if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0.8$, then it means that an patient with tumor size x has 80% chance of tumor being malignant
- In this task, larger tumor size has larger chance/probability of being malignant tumor.

Hypothesis representation

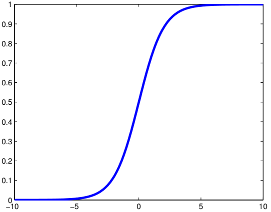
- Interpretation of logistic function:
- $h_{\boldsymbol{\theta}}(\boldsymbol{x})$ = estimated probability that $y = 1$ of input $\boldsymbol{x}$
- For example (plot below), if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0.8$, then it means that an patient with tumor size x has 80% chance of tumor being malignant
- In this task, larger tumor size has larger chance/probability of being malignant tumor.
- Thus, we can say that

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}).$$





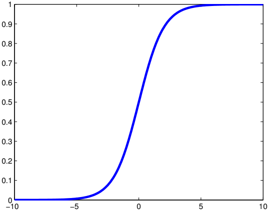
Decision boundary



- In logistic regression, we have

$$
\begin{cases}
h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\boldsymbol{\theta}^\top \boldsymbol{x}) = P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) \in [0, 1], & (14) \\
g(z) = \dfrac{1}{1 + \exp(-z)}, & (15)
\end{cases}
$$

Decision boundary



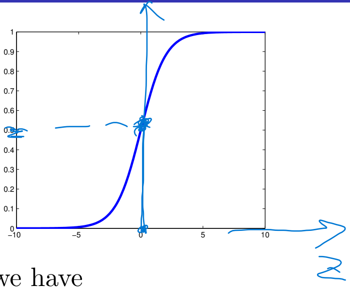
- In logistic regression, we have

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\boldsymbol{\theta}^\top \boldsymbol{x}) = P(y = 1|\boldsymbol{x};\boldsymbol{\theta}) \in [0, 1], \tag{14}$$

$$g(z) = \frac{1}{1 + \exp(-z)}, \tag{15}$$

- Suppose that if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq 0.5$, then we predict $y = 1$; if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) < 0.5$, then we predict $y = 0$

Decision boundary



- In logistic regression, we have

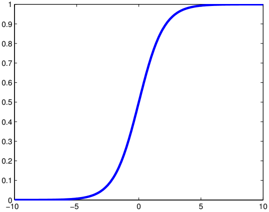
$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\boldsymbol{\theta}^\top \boldsymbol{x}) = P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) \in [0, 1], \tag{14}$$

$$g(z) = \frac{1}{1 + \exp(-z)}, \tag{15}$$

≥ 0.5

- Suppose that if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq 0.5$, then we predict $y = 1$; if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) < 0.5$, then we predict $y = 0$
- Correspondingly, if $\boldsymbol{\theta}^\top \boldsymbol{x} \geq 0$, we predict $y = 1$; if $\boldsymbol{\theta}^\top \boldsymbol{x} < 0$, then we predict $y = 0$.

Decision boundary



$\theta^T x \geq 0$

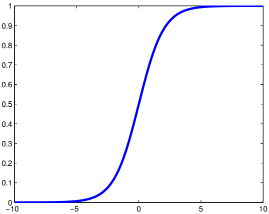
- In logistic regression, we have

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\boldsymbol{\theta}^\top \boldsymbol{x}) = P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) \in [0, 1], \tag{14}$$

$$g(z) = \frac{1}{1 + \exp(-z)}, \tag{15}$$

- Suppose that if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq 0.5$, then we predict $y = 1$; if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) < 0.5$, then we predict $y = 0$
- Correspondingly, if $\boldsymbol{\theta}^\top \boldsymbol{x} \geq 0$, we predict $y = 1$; if $\boldsymbol{\theta}^\top \boldsymbol{x} < 0$, then we predict $y = 0$. It determines the **decision boundary**

Decision boundary



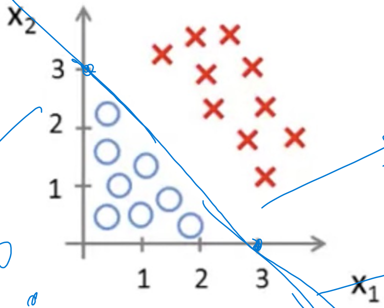
- In logistic regression, we have

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\boldsymbol{\theta}^\top \boldsymbol{x}) = P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) \in [0, 1], \tag{14}$$

$$g(z) = \frac{1}{1 + \exp(-z)}, \tag{15}$$

- Suppose that if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq 0.5$, then we predict $y = 1$; if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) < 0.5$, then we predict $y = 0$
- Correspondingly, if $\boldsymbol{\theta}^\top \boldsymbol{x} \geq 0$, we predict $y = 1$; if $\boldsymbol{\theta}^\top \boldsymbol{x} < 0$, then we predict $y = 0$. It determines the **decision boundary**
- **Decision boundary** is the curve/hyper-plane corresponding to $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0.5$, $\boldsymbol{\theta}^\top \boldsymbol{x} = 0$

Decision boundary



$\theta^T x \geq 0$

$\theta^T x < 0$

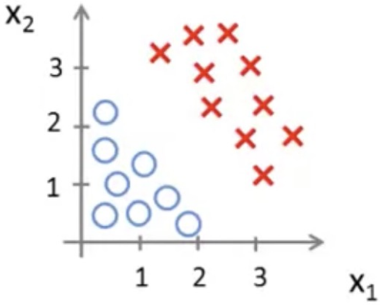
$\theta^T x$

- $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2) = g(-3 + x_1 + x_2)$

$$\begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix} \quad \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix}$$

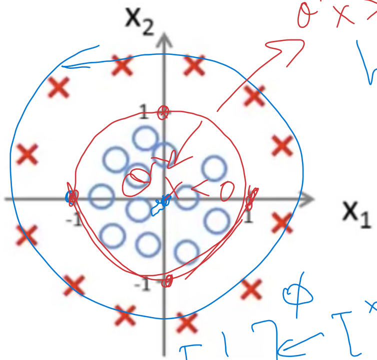
$\theta^T x = 0$

Decision boundary



- $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2) = g(-3 + x_1 + x_2)$
- Predict $y = 1$ if $-3 + x_1 + x_2 \geq 0$ (plot above)

Decision boundary



$\theta^T x > 0$

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

$z = \theta^T x$

ϕ

$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

Figure: Non-linear decision boundary

$\theta^T \phi(x) =$ $\quad$ $\theta^T x = 0$

$\begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1^2 \\ x_2^2 \end{bmatrix}$

- $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_1 x_1^2 + \theta_2 x_2^2) = g(-1 + x_1^2 + x_2^2)$

Decision boundary

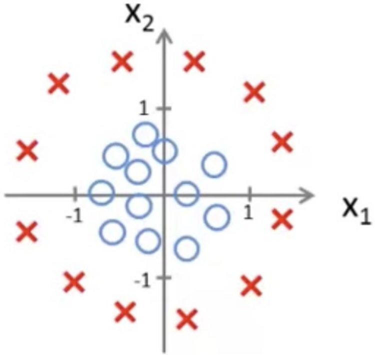


Figure: Non-linear decision boundary

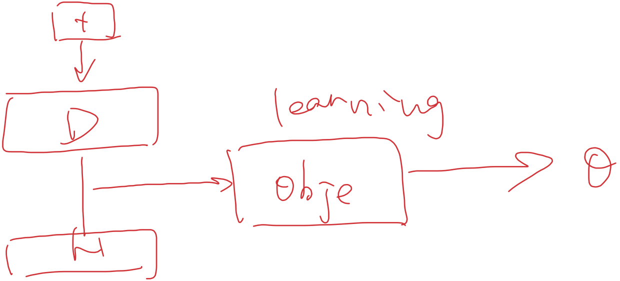
- $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_1 x_1^2 + \theta_2 x_2^2) = g(-1 + x_1^2 + x_2^2)$
- Predict $y = 1$ if $-1 + x_1^2 + x_2^2 \geq 0$ (plot above)

Cost function

- **Training set**: m training examples $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m}$

Cost function

- **Training set**: m training examples $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m}$
- **Hypothesis function**: $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{1}{1+\exp(-\boldsymbol{\theta}^\top \boldsymbol{x})}$



Cost function

- **Training set**: m training examples $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m}$
- **Hypothesis function**: $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{1}{1+\exp(-\boldsymbol{\theta}^\top \boldsymbol{x})}$
- How to learn the model parameter $\boldsymbol{\theta}$?

Cost function

- **Training set**: m training examples $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{m}$
- **Hypothesis function**: $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{1}{1+\exp(-\boldsymbol{\theta}^\top \boldsymbol{x})}$
- How to learn the model parameter $\boldsymbol{\theta}$?
- We need to design a cost function/objective function

Cost function

Cost function

- **Linear regression**: $J(\boldsymbol{\theta}) = \frac{1}{m}\sum_i^m \left(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}\right)^2$

Cost function

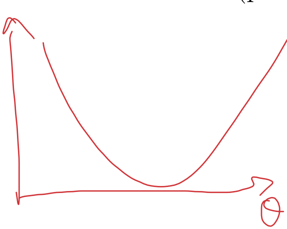
- **Linear regression**: $J(\boldsymbol{\theta}) = \frac{1}{m}\sum_i^m (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)})^2$
- $\text{cost}(h_{\boldsymbol{\theta}}(\boldsymbol{x}), y) = (h_{\boldsymbol{\theta}}(\boldsymbol{x}) - y)^2$, which is called ℓ_2 **loss** or **residual sum of squares**

Cost function

Cost function

$= \theta^T x$

- **Linear regression**: $J(\boldsymbol{\theta}) = \frac{1}{m}\sum_i^m (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)})^2$
- $\text{cost}(h_{\boldsymbol{\theta}}(\boldsymbol{x}), y) = (h_{\boldsymbol{\theta}}(\boldsymbol{x}) - y)^2$, which is called ℓ_2 **loss** or **residual sum of squares**
- It is convex function (plot below) w.r.t. $\boldsymbol{\theta}$ for linear regression



$(\theta^T x - y)^2$

$2(\theta^T x - y) \cdot x$

$2 x^T x \geq 0$

Cost function

$$\frac{\partial h_\theta(z)}{\partial z} = h \cdot (1-h) \qquad = \frac{1}{1+e(-\theta^T x)}$$

Cost function

- **Linear regression**: $\boxed{J(\boldsymbol{\theta}) = \frac{1}{m}\sum_i^m (h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)})^2}$
- $\text{cost}(h_{\boldsymbol{\theta}}(\boldsymbol{x}), y) = (h_{\boldsymbol{\theta}}(\boldsymbol{x}) - y)^2$, which is called ℓ_2 **loss** or **residual sum of squares**
- It is convex function (plot below) w.r.t. $\boldsymbol{\theta}$ for linear regression
- However, it is non-convex (plot below) w.r.t. $\boldsymbol{\theta}$ for logistic regression. Why? (please derive it)

$$\frac{\nabla J(\theta)}{\nabla \theta} = 2(h-y)\cdot h \cdot (1-h)\cdot(-x) = 2(h^2-hy)(1-h)(-x) = 2(h^2-h^3-hy-h^2y)(-x)$$

$$x \cdot x$$

$$\frac{\nabla^2 J(\theta)}{\nabla \theta} = 2[(2h-3h^2-y-2hy)(-x)\cdot(1-h)\cdot h\cdot(-x)]$$

Cost function

$P \longrightarrow -\log p \longrightarrow \sum p(-\log p)$

- **Cross-entropy**:

$$H(p,q) = -\int_x p(x)\log(q(x))d(x) \text{ or } \left[-\sum_x p(x)\log(q(x))\right]$$

- We set

$$y(\boldsymbol{x}) = P(y=1|\boldsymbol{x}), \;\; h_{\boldsymbol{\theta}}(\boldsymbol{x})) = P(y=1|\boldsymbol{x};\boldsymbol{\theta})$$

Cost function

- **Cross-entropy**: $H(p,q) \neq H(q,p)$

$$H(p,q) = -\int_x p(x)\log(q(x))d(x) \text{ or } -\sum_x p(x)\log(q(x))$$

- We set $= \{0,1\}$

$$y(\boldsymbol{x}) = P(y=1|\boldsymbol{x}),\ h_{\boldsymbol{\theta}}(\boldsymbol{x})) = P(y=1|\boldsymbol{x};\boldsymbol{\theta})$$

- **Cross-entropy loss**:

$$\begin{aligned}
\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) &= H(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) && (16)\\
&= -\sum_{\boldsymbol{x}} y(\boldsymbol{x})\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})) && (17)\\
&= \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1\\ -\log(1-h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases} && (18)
\end{aligned}$$

Cost function

Cost function

- **Cross-entropy loss**:

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases}$$

Cost function

- **Cross-entropy loss**:

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases}$$

- For $y = 1$, if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 1$, *i.e.*, $P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) = 1$, then the prediction equals to the ground-truth label, the cost is 0

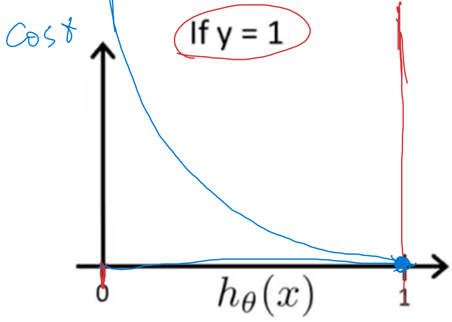
Cost function

- **Cross-entropy loss:**

log(0)

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases}$$

- For $y = 1$, if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 1$, *i.e.*, $P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) = 1$, then the prediction equals to the ground-truth label, the cost is 0
- For $y = 1$, if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \to 0$, *i.e.*, $P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) \to 0$, then it should be penalized with a very large cost. Here we have $\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) \to \infty$



Cost function

Cost function

- **Cross-entropy loss**:

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases}$$

Cost function

- **Cross-entropy loss**:

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases}$$

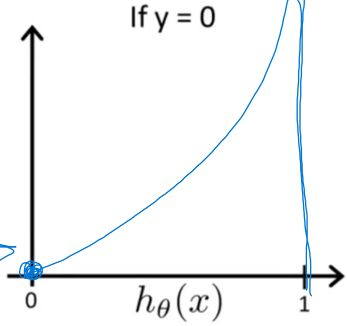
- For $y = 0$, if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0$, *i.e.*, $P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) = 0$, then the prediction equals to the ground-truth label, the cost is 0

Cost function

- **Cross-entropy loss**:

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases}$$

- For $y = 0$, if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0$, *i.e.*, $P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) = 0$, then the prediction equals to the ground-truth label, the cost is 0
- For $y = 0$, if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \to 1$, *i.e.*, $P(y = 1|\boldsymbol{x}; \boldsymbol{\theta}) \to 0$, then it should be penalized with a very large cost. Here we have $\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) \to \infty$



Cost function

Cost function

- **Cross-entropy loss**:

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases}$$

Cost function

- **Cross-entropy loss**:

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases}$$

- Which states are true?
  - If $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = y$, then $\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = 0$ for both $y = 0$ and $y = 1$ ✓
  - If $y = 0$, then $\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) \to \infty$ as $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \to 1$ ✓
  - If $y = 0$, then $\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) \to \infty$ as $h_{\boldsymbol{\theta}}(\boldsymbol{x}) \to 0$ ✗
  - Regardless whether $y = 0$ or $y = 1$, if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0.5$, then $\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) > 0$ ✓

Cost function of logistic regression

- **Cost function of logistic regression**

$$J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i}^{m} \text{cost}(y^{(i)}, h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})), \tag{19}$$

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases} \tag{20}$$

Cost function of logistic regression

- **Cost function of logistic regression**

$$J(\boldsymbol{\theta}) = \frac{1}{m} \sum_{i}^{m} \text{cost}(y^{(i)}, h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})), \tag{19}$$

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases} \tag{20}$$

- The above cost function can be simplified as follows

$y^{(i)} \in \{0, 1\}$

$$J(\boldsymbol{\theta}) = -\frac{1}{m} \sum_{i}^{m} \left[y^{(i)} \log(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) + (1 - y^{(i)}) \log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) \right]. \tag{21}$$

Cost function of logistic regression

- **Cost function of logistic regression**

$$J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i}^{m} \text{cost}(y^{(i)}, h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})), \tag{19}$$

$$\text{cost}(y(\boldsymbol{x}), h_{\boldsymbol{\theta}}(\boldsymbol{x})) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x})), & \text{if } y(\boldsymbol{x}) = 0 \end{cases} \tag{20}$$

- The above cost function can be simplified as follows

$$\frac{\partial h(z)}{\partial z} = h \cdot (1-h)$$

$$J(\boldsymbol{\theta}) = -\frac{1}{m}\sum_{i}^{m} \left[y^{(i)} \log(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) + (1 - y^{(i)}) \log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) \right]. \tag{21}$$

- It is convex function (plot below) w.r.t. $\boldsymbol{\theta}$. (Please derive it)

Gradient descent

$$J(\boldsymbol{\theta}) = -\frac{1}{m}\sum_{i}^{m}\left[y^{(i)}\log(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) + (1-y^{(i)})\log(1-h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}))\right]. \tag{22}$$

- Learning $\boldsymbol{\theta}$ by $\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$
- **Gradient descent**: repeat the following update until convergence

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \tag{23}$$

θ^{t+1}

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i}^{m}[h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}]\boldsymbol{x}^{(i)} \tag{24}$$

- How to define convergence? Calculating the changes of $J(\boldsymbol{\theta})$ or $\boldsymbol{\theta}$ in the last K steps, if the change is lower than a threshold, than it can be seen as convergence. Remember that choosing suitable learning rate η is important to achieve a good converged solution.

10^{-4}

Gradient descent

$$J(\boldsymbol{\theta}) = -\frac{1}{m}\sum_{i}^{m}\Big[y^{(i)}\log(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) + (1-y^{(i)})\log(1-h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}))\Big]. \tag{22}$$

- Learning $\boldsymbol{\theta}$ by $\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$
- **Gradient descent**: repeat the following update until convergence

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \tag{23}$$

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i}^{m}[h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}]\boldsymbol{x}^{(i)} \tag{24}$$

Gradient descent

$$J(\boldsymbol{\theta}) = -\frac{1}{m}\sum_{i}^{m}\left[y^{(i)}\log(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) + (1-y^{(i)})\log(1-h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}))\right]. \tag{22}$$

- Learning $\boldsymbol{\theta}$ by $\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$
- **Gradient descent**: repeat the following update until convergence

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta\nabla_{\boldsymbol{\theta}}J(\boldsymbol{\theta}) \tag{23}$$

$$\nabla_{\boldsymbol{\theta}}J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i}^{m}[h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}]\boldsymbol{x}^{(i)} \tag{24}$$

- How to define convergence?

Gradient descent

$$J(\boldsymbol{\theta}) = -\frac{1}{m}\sum_{i}^{m}\left[y^{(i)}\log(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) + (1-y^{(i)})\log(1-h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}))\right]. \tag{22}$$

- Learning $\boldsymbol{\theta}$ by $\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$
- **Gradient descent**: repeat the following update until convergence

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \tag{23}$$

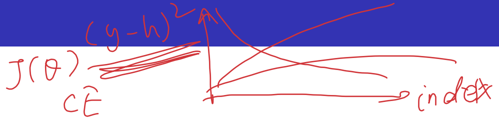
$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i}^{m}[h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) - y^{(i)}]\boldsymbol{x}^{(i)} \tag{24}$$

- How to define convergence? Calculating the changes of $J(\boldsymbol{\theta})$ or $\boldsymbol{\theta}$ in the last K steps, if the change is lower than a threshold, than it can be seen as convergence. Remember that choosing suitable learning rate η is important to achieve a good converged solution.

Gradient descent

Suppose you are running a logistic regression model, and you should observe the learning procedure to find a suitable learning rate η. Which of the following is reasonable to make sure η is set properly and that the gradient descent is running correctly?

Gradient descent



Suppose you are running a logistic regression model, and you should observe the learning procedure to find a suitable learning rate η. Which of the following is reasonable to make sure η is set properly and that the gradient descent is running correctly?

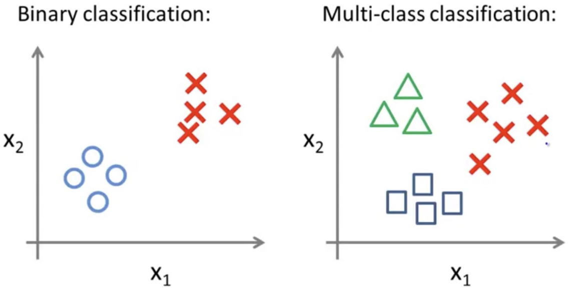
- Plot $J(\boldsymbol{\theta}) = -\frac{1}{m}\sum_i^m (y^{(i)} - h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}))^2$ as a function of the number of iterations (*i.e.*, the horizontal axis is the iteration number) and make sure $J(\boldsymbol{\theta})$ is decreasing on every iteration.
- Plot $J(\boldsymbol{\theta}) = -\frac{1}{m}\sum_i^m \left[y^{(i)}\log(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) + (1 - y^{(i)})\log(1 - h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}))\right]$ as a function of the number of iterations (*i.e.*, the horizontal axis is the iteration number) and make sure $J(\boldsymbol{\theta})$ is decreasing on every iteration.
- Plot $J(\boldsymbol{\theta})$ as a function of $\boldsymbol{\theta}$ and make sure it is decreasing on every iteration.
- Plot $J(\boldsymbol{\theta})$ as a function of $\boldsymbol{\theta}$ and make sure it is convex.

Multi-class classification

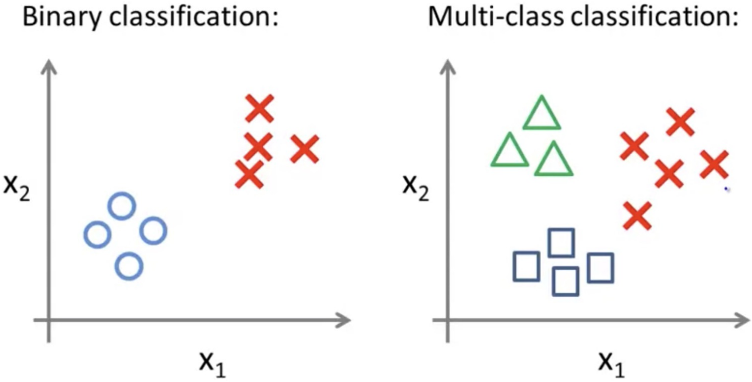
- In above examples and derivations, we only consider the binary classification problem, *i.e.*, $y \in \{0, 1\}$.

Multi-class classification

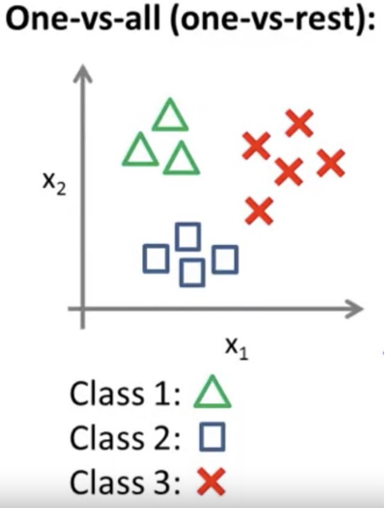
- In above examples and derivations, we only consider the binary classification problem, *i.e.*, $y \in \{0, 1\}$.
- However, many practical problems involve with multi-class classification:
  - Whether forecast: sunny, cloudy, rain, snow
  - Email tagging: work, friends, families, hobby

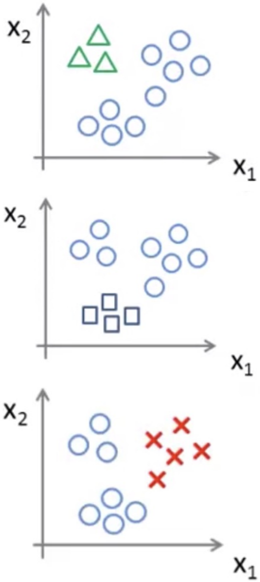


Multi-class classification



Multi-class classification: one-vs-all





Multi-class classification: one-vs-all

One-vs-all logistic regression:

- Train a binary logistic regression $h_{\boldsymbol{\theta}_i}(\cdot)$ for each class i, by setting all samples of other classes as negative class
- For a new testing sample $\boldsymbol{x}$, predict its class as $\arg\max_i h_{\boldsymbol{\theta}_i}(\boldsymbol{x})$.

Multi-class classification: one-vs-all

One-vs-all logistic regression:

- Train a binary logistic regression $h_{\boldsymbol{\theta}_i}(\cdot)$ for each class i, by setting all samples of other classes as negative class
- For a new testing sample $\boldsymbol{x}$, predict its class as $\arg\max_i h_{\boldsymbol{\theta}_i}(\boldsymbol{x})$.

Pros: Easy to implement

Multi-class classification: one-vs-all

One-vs-all logistic regression:

- Train a binary logistic regression $h_{\boldsymbol{\theta}_i}(\cdot)$ for each class i, by setting all samples of other classes as negative class
- For a new testing sample $\boldsymbol{x}$, predict its class as $\arg\max_i h_{\boldsymbol{\theta}_i}(\boldsymbol{x})$.

Pros: Easy to implement
Cons: The training cost is too high, and is difficult to scale to tasks with large number of classes.

Multi-class classification: logistic regression with softmax function

- Softmax function:

$$h_{\boldsymbol{\theta}_i}(\boldsymbol{x}) = \frac{\exp(\boldsymbol{\theta}_i^\top \boldsymbol{x})}{\sum_i \exp(\boldsymbol{\theta}_i^\top \boldsymbol{x})} = P(y = i | \boldsymbol{x}; \boldsymbol{\Theta}), \tag{25}$$

where $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_i\}_i^C$ with C being the number of classes.

Multi-class classification: logistic regression with softmax function

- Softmax function:

$$h_{\boldsymbol{\theta}_i}(\boldsymbol{x}) = \frac{\exp(\boldsymbol{\theta}_i^\top \boldsymbol{x})}{\sum_i \exp(\boldsymbol{\theta}_i^\top \boldsymbol{x})} = P(y = i | \boldsymbol{x}; \boldsymbol{\Theta}), \tag{25}$$

where $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_i\}_i^C$ with C being the number of classes.

- Cost function:

$$J(\boldsymbol{\Theta}) = -\frac{1}{m} \sum_j^m \sum_i^C \left[\mathbb{I}(y^{(j)} = i) \log(h_{\boldsymbol{\theta}_i}(\boldsymbol{x}^{(j)})) \right], \tag{26}$$

where $\mathbb{I}(a) = 1$ if a is true, otherwise $\mathbb{I}(a) = 0$.

Multi-class classification: logistic regression with softmax function

- Softmax function:

$$h_{\boldsymbol{\theta}_i}(\boldsymbol{x}) = \frac{\exp(\boldsymbol{\theta}_i^\top \boldsymbol{x})}{\sum_i \exp(\boldsymbol{\theta}_i^\top \boldsymbol{x})} = P(y = i | \boldsymbol{x}; \boldsymbol{\Theta}), \tag{25}$$

  where $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_i\}_i^C$ with C being the number of classes.

- Cost function:

$$J(\boldsymbol{\Theta}) = -\frac{1}{m} \sum_j^m \sum_i^C \left[\mathbb{I}(y^{(j)} = i) \log(h_{\boldsymbol{\theta}_i}(\boldsymbol{x}^{(j)})) \right], \tag{26}$$

  where $\mathbb{I}(a) = 1$ if a is true, otherwise $\mathbb{I}(a) = 0$.

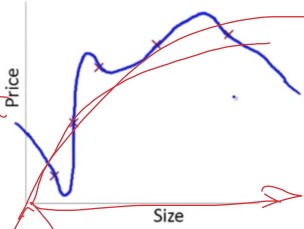
- It can also be optimized by gradient descent. (Please derive its gradient)

Overfitting in linear regression

Overfitting in linear regression

Addressing overfitting:

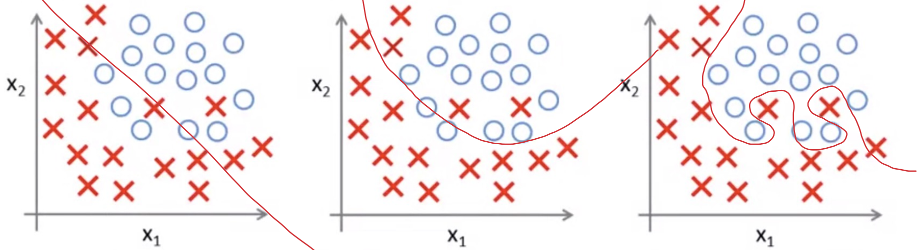
$x_1 =$ size of house
$x_2 =$ no. of bedrooms
$x_3 =$ no. of floors
$x_4 =$ age of house
$x_5 =$ average income in neighborhood
$x_6 =$ kitchen size
$\vdots$
x_{100}



Overfitting: If we have too many features, the learned hypothesis may fit the training data very well (low bias), but fail to generalize to new examples.

Overfitting in logistic regression

Example: Logistic regression



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

(g = sigmoid function)

$$g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2)$$

$$g(\theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2 + \theta_5 x_1^2 x_2^3 + \theta_6 x_1^3 x_2 + \dots$$

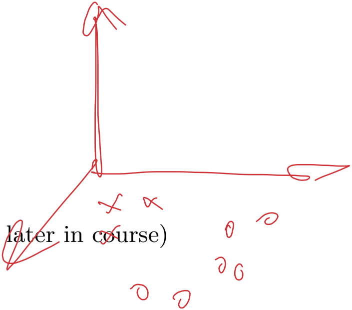
Addressing Overfitting

Two options:

Addressing Overfitting

Two options:

1. Reducing the number of features:
    - Feature selection
    - Dimensionality reduction (introduced later in course)



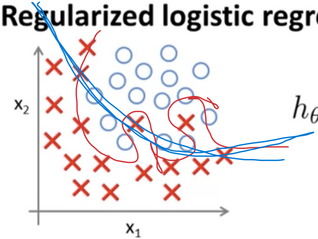
Addressing Overfitting

Two options:

1. Reducing the number of features:
   - Feature selection
   - Dimensionality reduction (introduced later in course)
2. Regularization:
   - Keep all features, but reduce magnitude/value of each parameter, such that each feature contributes a bit to predict y

Regularized logistic regression

Regularized logistic regression.



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2 + \theta_5 x_1^2 x_2^3 + \dots)$$

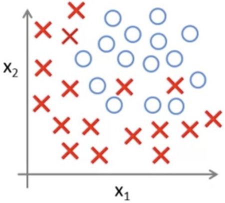
Cost function:

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

$$+ \frac{2\lambda}{m} \|\theta\|_2^2$$

$$\sum_i^n \theta_i^2$$

Regularized logistic regression.



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2 + \theta_5 x_1^2 x_2^3 + \dots)$$

Cost function:

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

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

$J(\theta) = CE(\theta)$

Gradient descent

Repeat $\{$

$$\theta_j := \theta_j - \alpha \quad \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}$$

$$(j = 0, 1, 2, 3, \ldots, n)$$

$\}$

θ_0

Regularized logistic regression

$J(\theta) = CE(\theta) + \frac{\lambda}{2}\|\theta\|^2$

Gradient descent

Repeat {

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

$$\theta_j := \theta_j - \alpha \quad \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)} \qquad (j = \cancel{0}, 1, 2, 3, \ldots, n)$$

}

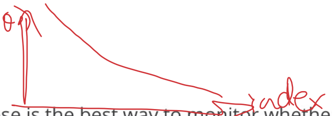
$- \lambda \theta_j$

$\sum_{j=1}^{n} \theta_j^2$

Note: the bias parameter θ_0 is not penalized.

Regularized logistic regression

$$J(\theta) = CE(\theta) + \frac{\lambda}{2}\|\theta\|^2$$



When using regularized logistic regression, which of these is the best way to monitor whether gradient descent is working correctly?

- Plot $-\left[\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)}))\right]$ as a function of the number of iterations and make sure it's decreasing.
- Plot $-\left[\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)}))\right] - \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$ as a function of the number of iterations and make sure it's decreasing.
- Plot $-\left[\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)}))\right] + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$ as a function of the number of iterations and make sure it's decreasing.
- Plot $\sum_{j=1}^{n}\theta_j^2$ as a function of the number of iterations and make sure it's decreasing.

- **Linear model**:

$$\begin{cases} \mu(\boldsymbol{x}|\boldsymbol{\theta}) = \boldsymbol{\theta}^\top \phi(\boldsymbol{x}), & (27) \\ y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), & (28) \end{cases}$$

where f denotes a distribution function.

Generalized linear regression

- **Linear model**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = \boldsymbol{\theta}^\top \phi(\boldsymbol{x}), \tag{27}$$
$$y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), \tag{28}$$

where f denotes a distribution function.

- **Generalized linear model (GLM)**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = \boxed{g^{-1}}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \tag{29}$$
$$y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), \tag{30}$$

where g is called **link function**, which is required to be monotonically increasing differentiable.

Generalized linear regression

- **Linear model**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = \boldsymbol{\theta}^\top \phi(\boldsymbol{x}), \tag{27}$$
$$y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), \tag{28}$$

  where f denotes a distribution function.

- **Generalized linear model (GLM)**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = g^{-1}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \tag{29}$$
$$y(x|\boldsymbol{\theta}) \sim f(\mu(\boldsymbol{x}|\boldsymbol{\theta})), \tag{30}$$

  where g is called **link function**, which is required to be monotonically increasing differentiable.

- The standard linear model is a special case of GLM with $g(a) = a$.

Logistic regression: probabilistic modeling

- **Logistic regression**:

$g^{-1}(z) = \frac{1}{1+e(-z)}$ $\quad \ln \frac{u}{1-u}$

$$
\begin{aligned}
\mu(\boldsymbol{x}|\boldsymbol{\theta}) &= \text{Sigmoid}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), && (31)\\
y(\boldsymbol{x}|\boldsymbol{\theta}) &\sim \text{Bernoulli}(\mu(\boldsymbol{x}|\boldsymbol{\theta})). && (32)
\end{aligned}
$$

$$\underset{\theta}{\arg\max} \log\left[u^{y} \cdot (1-u)^{(1-y)}\right]$$

$$\Downarrow$$

$$\min CE(\theta) \quad y \log u + (1-y)\log(1-u)$$

$$-CE(\theta) \quad y \log \frac{1}{1+e(-\theta^\top x)} + (1-y)\log \frac{e(-\theta^\top x)}{1+e(-\theta^\top x)}$$

Logistic regression: probabilistic modeling

- **Logistic regression**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = \text{Sigmoid}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \tag{31}$$

$$y(\boldsymbol{x}|\boldsymbol{\theta}) \sim \text{Bernoulli}(\mu(\boldsymbol{x}|\boldsymbol{\theta})). \tag{32}$$

- We have

$$P(y|\boldsymbol{x};\boldsymbol{\theta}) = \begin{cases} \mu & \text{if } y = 1 \\ 1 - \mu & \text{if } y = 0 \end{cases} \tag{33}$$

Logistic regression: probabilistic modeling

- **Logistic regression**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = \text{Sigmoid}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \tag{31}$$
$$y(\boldsymbol{x}|\boldsymbol{\theta}) \sim \text{Bernoulli}(\mu(\boldsymbol{x}|\boldsymbol{\theta})). \tag{32}$$

- We have

$$P(y|\boldsymbol{x};\boldsymbol{\theta}) = \begin{cases} \mu & \text{if } y = 1 \\ 1 - \mu & \text{if } y = 0 \end{cases} \tag{33}$$

- The log-likelihood is

$$\mathcal{L}(\boldsymbol{\theta}) = y\log(\mu) + (1-y)\log(1-\mu) \tag{34}$$

Logistic regression: probabilistic modeling

- **Logistic regression**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = \text{Sigmoid}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \tag{31}$$
$$y(\boldsymbol{x}|\boldsymbol{\theta}) \sim \text{Bernoulli}(\mu(\boldsymbol{x}|\boldsymbol{\theta})). \tag{32}$$

- We have

$$P(y|\boldsymbol{x};\boldsymbol{\theta}) = \begin{cases} \mu & \text{if } y = 1 \\ 1-\mu & \text{if } y = 0 \end{cases} \tag{33}$$

- The log-likelihood is

$$\mathcal{L}(\boldsymbol{\theta}) = y\log(\mu) + (1-y)\log(1-\mu) \tag{34}$$

- Thus, we obtain

$$\underline{\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) \equiv \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})} \tag{35}$$

Logistic regression: probabilistic modeling

- **Logistic regression**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = \text{Sigmoid}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \tag{36}$$

$$y(\boldsymbol{x}|\boldsymbol{\theta}) \sim \text{Bernoulli}(\mu(\boldsymbol{x}|\boldsymbol{\theta})). \tag{37}$$

- **Logistic regression**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = \text{Sigmoid}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \tag{36}$$

$$y(\boldsymbol{x}|\boldsymbol{\theta}) \sim \text{Bernoulli}(\mu(\boldsymbol{x}|\boldsymbol{\theta})). \tag{37}$$

$\text{Lap}(\theta|b)$

- **Regularized Logistic regression**: we assume $\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta}|\mathbf{0}, \sigma^2\mathbf{I})$, then we have

$$\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) + \log \mathcal{N}(\boldsymbol{\theta}|\mathbf{0}, \sigma^2\mathbf{I}) \equiv \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) + \frac{\lambda}{2m}\sum_{j=1}^{n} \theta_j^2 \tag{38}$$

$\sum |\theta_j|$ $\exp\left(-\frac{|x-a|}{b}\right)$

Logistic regression: probabilistic modeling

- **Logistic regression**:

$$\mu(\boldsymbol{x}|\boldsymbol{\theta}) = \text{Sigmoid}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})), \tag{36}$$
$$y(\boldsymbol{x}|\boldsymbol{\theta}) \sim \text{Bernoulli}(\mu(\boldsymbol{x}|\boldsymbol{\theta})). \tag{37}$$

- **Regularized Logistic regression**: we assume $\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta}|\mathbf{0}, \sigma^2\mathbf{I})$, then we have

$$\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) + \log \mathcal{N}(\boldsymbol{\theta}|\mathbf{0}, \sigma^2\mathbf{I}) \equiv \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) + \frac{\lambda}{2m}\sum_{j=1}^{n} \theta_j^2 \tag{38}$$

- **Regularized Logistic regression**: if we assume $\boldsymbol{\theta} \sim \text{Laplace}(\boldsymbol{\theta}|\mathbf{0}, b)$, then we have

$$\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) + \log \text{Laplace}(\boldsymbol{\theta}|\mathbf{0}, b) \equiv \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) + \frac{\lambda}{2m}\sum_{j=1}^{n} |\theta_j| \tag{39}$$

Summary of linear models

- Linear model is the linear function of the parameter $\boldsymbol{\theta}$, rather than the input feature

Summary of linear models

- Linear model is the linear function of the parameter $\boldsymbol{\theta}$, rather than the input feature
- Linear model is a special case of generalized linear model, while generalized linear model is not always linear

Summary of linear models

- Linear model is the linear function of the parameter $\boldsymbol{\theta}$, rather than the input feature
- Linear model is a special case of generalized linear model, while generalized linear model is not always linear
- Logistic model is also a special case of generalized linear model. It is used to solve classification task because the output range of its hypothesis function (*i.e.*, sigmoid function) is $[0, 1]$, which can be seen as posterior probability

Summary of linear models

$\theta^T \phi(x)$

- Linear model is the linear function of the parameter $\boldsymbol{\theta}$, rather than the input feature
- Linear model is a special case of generalized linear model, while generalized linear model is not always linear
- Logistic model is also a special case of generalized linear model. It is used to solve classification task because the output range of its hypothesis function (*i.e.*, sigmoid function) is $[0, 1]$, which can be seen as posterior probability
- Different variants of linear models correspond to different distributions of $p(y|\boldsymbol{x}, \boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$, according to the task and the data, *i.e.*, handling outliers or alleviating overfitting

$p(y \mid x;\theta)$	$p(\theta)$
Gau	unit ( )
B	Gau →
La	La →