Logistic Regression

© Malay K. Das, 210 Southern Lab, ph-7359, mkdas@iitk.ac.in

Office hours: W 1030-1130, SL-210

Previously:

Optimization

Today:

Use of linear regression technique for classification

HW due:

September 03, 2024

Computing quiz:

September 04, 2024

So far in this course

1. Regression
   - Linear regression
     - Simple linear regression
     - Multiple linear regression
   - Nonlinear regression

2. Classification
   - K-nearest neighbor

3. Mathematics for machine learning
   - Linear algebra
   - Optimization

4. Applications problems

Logistic regression: Classification using regression techniques

Predicts probabilities of certain output

Simple Linear Regression

For training data $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{n}$ estimate $\widehat{y}_\star(x_\star)$

in simple linear regression $x_i \in \mathbb{R}, y_i \in \mathbb{R}$

residual (not the noise from experiments)

we assume a hypothesis $\widehat{y} = w_0 + w_1 x$ $\quad y = \widehat{y} + e$

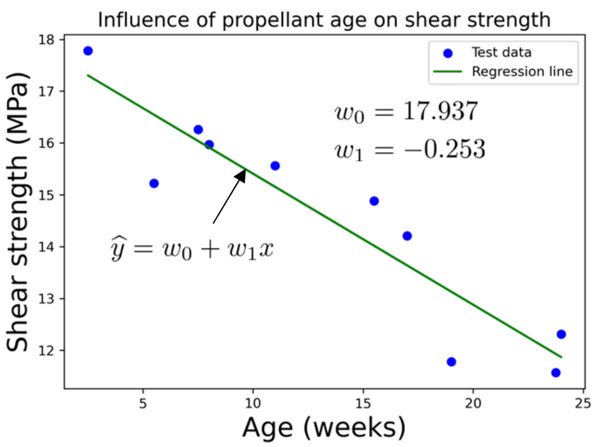
Loss Function $L_i = e_i^2 = (w_0 + w_1 x_i - y_i)^2$ **Cost Function** $E(w_0, w_1) = \frac{1}{n}\sum_{i=1}^{n} L_i$

Least square estimate minimizes **Cost Function**

in simple **Multiple Linear Regression** training data $\mathcal{T} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ $\quad \mathbf{x}_i \in \mathbb{R}, y_i \in \mathbb{R}$

Linear regression (simple or multiple) fits **smooth** curves/surfaces

Simple linear regression: application



Shear strength of rocket propellant

Test	Propellant age (Weeks) x_i	Shear strength (MPa) y_i
1	15.5	14.88
2	23.75	11.57
3	8	15.97
4	17	14.21
5	5.5	15.22
6	19	11.78
7	24	12.31
8	2.5	17.78
9	7.5	16.26
10	11	15.56

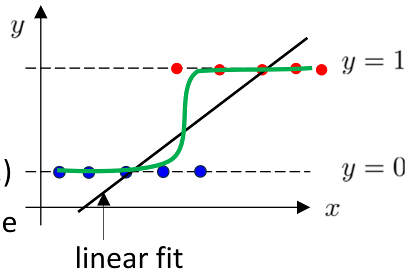
The same problem can also be modeled as a **classification** problem

Logistic regression uses regression technique for classification

Regression uses training dataset $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{n}$ to estimate $\widehat{y}_\star(x_\star)$

Logistic regression is used when $y_i = 0, 1$

Linear regression cannot produce discrete output; not a good option here



$y_i = 0, 1$ are **categorical** variables (yes/no, pass/fail etc.)

Logistic regression is thus a **Classification** technique

Therefore, for the test input $x_\star$

the model should give an output of $y_\star = 0$ or $y_\star = 1$

We can solve the problem using regression, if we accept that the model will give us the **probabilities** of being in a class!

Example: Propellant strength degradation with age

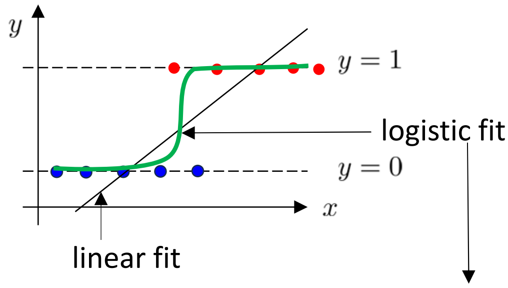
Below certain strength (above certain age) , we rate the propellant as unusable (marked as fail)

Test	Propellant age (Weeks) x	Shear strength test results
1	15.5	fail = 1
2	23.75	fail = 1
3	8	pass = 0
4	17	fail = 1
5	5.5	pass = 0
6	19	fail = 1
7	24	fail = 1
8	2.5	pass = 0
9	7.5	pass = 0
10	11	pass = 0

Test	Propellant age (Weeks) x	Class c	$y = \mathrm{P}(c = 1 \mid x)$
1	15.5	1	1
2	23.75	1	1
3	8	0	0
4	17	1	1
5	5.5	0	0
6	19	1	1
7	24	1	1
8	2.5	0	0
9	7.5	0	0
10	11	0	0

probability that $c = 1$ for a given x (can take values 0 or 1 only)

$\widehat{y} =$ **estimated** probability that $c = 1$ for a given x (takes any value in $[0, 1]$)



We will now create model to predict $\widehat{y} = \mathrm{P}\left(c=1 \mid x\right)$

Estimated probability that $c=1$ for a given value of x

$$\mathrm{P}\left(c=0 \mid x\right) = 1 - \mathrm{P}\left(c=1 \mid x\right) = 1 - \widehat{y}$$

$$y = 0, 1 \qquad 0 \le \widehat{y} \le 1$$

continuous function, regression could be useful now

Test	Propellant age (Weeks) x	Class c	$y = \mathrm{P}\left(c=1 \mid x\right)$ $y = c$
1	15.5	1	1
2	23.75	1	1
3	8	0	0
4	17	1	1
5	5.5	0	0
6	19	1	1
7	24	1	1
8	2.5	0	0
9	7.5	0	0
10	11	0	0

We wish to predict $\widehat{y} = \mathrm{P}\left(c=1 \mid x\right)$

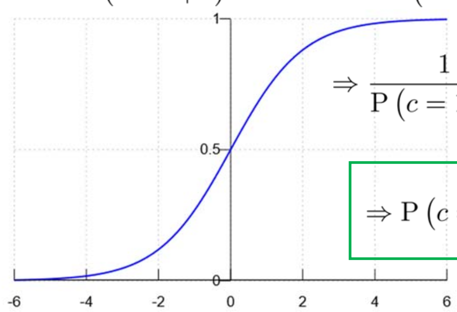
We cannot fit simple linear regression $\widehat{y} = w_0 + w_1 x$

since, output from such model may not be in $[0,1]$

$$\boxed{\textbf{Logistic regression} \quad z = w_0 + w_1 x = \ln\left[\frac{\mathrm{P}\left(c=1 \mid x\right)}{\mathrm{P}\left(c=0 \mid x\right)}\right]}$$

$$\frac{\mathrm{P}\left(c=1 \mid x\right)}{1-\mathrm{P}\left(c=1 \mid x\right)} = e^{z} \quad \Rightarrow \frac{1-\mathrm{P}\left(c=1 \mid x\right)}{\mathrm{P}\left(c=1 \mid x\right)} = e^{-z}$$

$$\Rightarrow \frac{1}{\mathrm{P}\left(c=1 \mid x\right)} = 1 + e^{-z}$$



$$\boxed{\Rightarrow \mathrm{P}\left(c=1 \mid x\right) = \frac{1}{1+e^{-z}} = \operatorname{logit}\left(z\right)}$$

logistic function

or **sigmoid** function

Test	Propellant age (Weeks) x	$y = c$
1	15.5	1
2	23.75	1
3	8	0
4	17	1
5	5.5	0
6	19	1
7	24	1
8	2.5	0
9	7.5	0
10	11	0

We wish to predict $\widehat{y} = \mathrm{P}\left(c=1 \mid x\right) = \dfrac{1}{1+e^{-z}}$ $\quad z = w_0 + w_1 x$

Estimated probabilities

since either $c=0$ or $c=1$

$\mathrm{P}\left(c=1 \mid x\right) = \widehat{y}\left(x\right)$ $\quad \mathrm{P}\left(c=0 \mid x\right) = 1 - \widehat{y}\left(x\right)$ $\quad \mathrm{P}\left(c=0 \mid x\right) = 1 - \dfrac{1}{1+e^{-z}}$

Combining $\mathrm{P}\left(c \mid x\right) = \left(\widehat{y}\right)^{c}\left(1-\widehat{y}\right)^{1-c} = \left(\widehat{y}\right)^{y}\left(1-\widehat{y}\right)^{1-y}$ since $c = y$

for the training data (x_i, y_i) estimated probability $\mathrm{P_i} = \mathrm{P}\left(c_i \mid x_i\right) = \left(\widehat{y}_i\right)^{y_i}\left(1-\widehat{y}_i\right)^{1-y_i}$

for the training dataset $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{n}$ where $\widehat{y}_i = \dfrac{1}{1+\exp\left(-w_0 - w_1 x_i\right)}$

We may minimize the least square cost function $\dfrac{1}{n}\sum_{n}\left[y_i - \left(\widehat{y}_i\right)^{y_i}\left(1-\widehat{y}_i\right)^{1-y_i}\right]^2$

Minimizing such cost function is difficult, we may define other kind of cost function

For the training data (x_i, y_i) estimated probability $\mathrm{P}\left(c_i \mid x_i\right) = \left(\widehat{y}_i\right)^{y_i}\left(1-\widehat{y}_i\right)^{1-y_i}$

If the estimation is correct $\widehat{y}_i = y_i \Rightarrow \mathrm{P}\left(c_i \mid x_i\right) = \left(y_i\right)^{y_i}\left(1-y_i\right)^{1-y_i}$ For $y_i = 0, 1$ $\mathrm{P}\left(c_i \mid x_i\right) = 1$

If the estimation is incorrect $\mathrm{P}\left(c_i \mid x_i\right) = 1 - \widehat{y}_i$ or $\mathrm{P}\left(c_i \mid x_i\right) = \widehat{y}_i$ $\Rightarrow \mathrm{P}\left(c_i \mid x_i\right) < 1$

for the training dataset $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{n}$ we maximize $\phi = \prod_{i=1}^{n}\left(\widehat{y}_i\right)^{y_i}\left(1-\widehat{y}_i\right)^{1-y_i}$

Ideally the maximum value of ϕ should be 1 $\quad \widehat{y}_i = \dfrac{1}{1+\exp\left(-w_0 - w_1 x_i\right)}$

We now define **cost function** $E = -\dfrac{1}{n}\ln\phi$

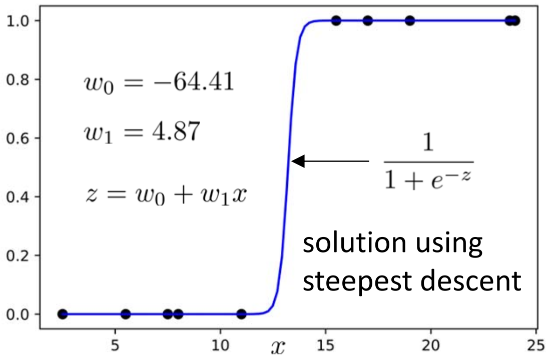
The above cost function, different from standard least square cost function, is known as **Cross Entropy cost function**

There is no closed form solution for the minimization problem, we may use, gradient descent, or any other suitable optimization methods

$$\widehat{y}_i = \frac{1}{1+\exp(-z_i)} \qquad 1-\widehat{y}_i = \frac{1}{1+\exp(z_i)} \qquad z_i = w_0 + w_1 x_i$$

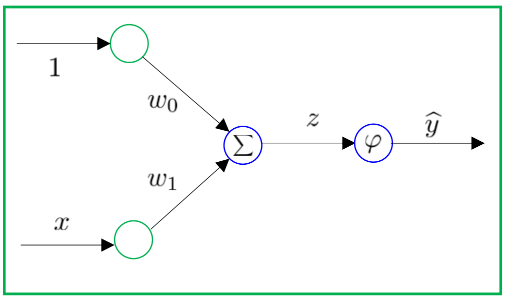
We wish to maximize $\phi = \prod_{i=1}^{n} (\widehat{y}_i)^{y_i} (1-\widehat{y}_i)^{1-y_i}$

Cost function $E = -\frac{1}{n}\ln\phi$ (to be minimized)



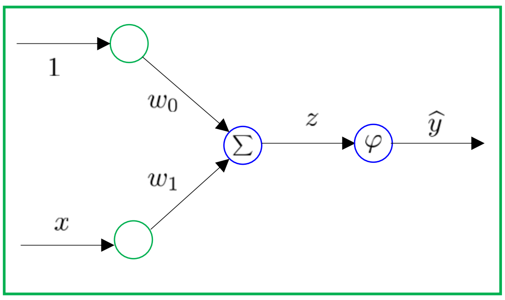
Test	Propellant age (Weeks) x	$y = c$
1	15.5	1
2	23.75	1
3	8	0
4	17	1
5	5.5	0
6	19	1
7	24	1
8	2.5	0
9	7.5	0
10	11	0

Linear regression



Activation function $\varphi(z) = z$

Logistic regression



Activation function $\varphi(z) = \frac{1}{1+e^{-z}}$

Both algorithms converge toward a general structure: artificial neural network

Perceptron is the simplest unit of artificial neural network, to be discussed later