Logistic Regression

Tao Tan

Logistic Regression

- Linear Regression
  - Housing price
  - Stock price
  - Exam score

Predicted value is continous

Logistic regression is a classification problem

Whether or not a person has Covid or regular pneomonia or non-penumonia

$$Y = \begin{cases} \text{Covid} \\ \text{Regular pnemonia}\mathit{monia} \\ \text{non - pnemonia} \end{cases}$$

Success of a vaccination

$$Y = \begin{cases} \text{Yes} \\ \text{No} \end{cases}$$

Whats the gender of a person

$$Y = \begin{cases} \text{Male} \\ \text{Female} \end{cases}$$

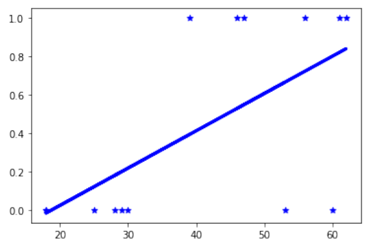
prediction is catergorical. For binary, 0 is negative and 1 is positive

Example: Age predicts car ownership

age	have_car
18	0
25	0
47	1
53	0
46	1
56	1
60	0
62	1
61	1
18	0
29	0
28	0
30	0
39	1

Linear regression

Have a car

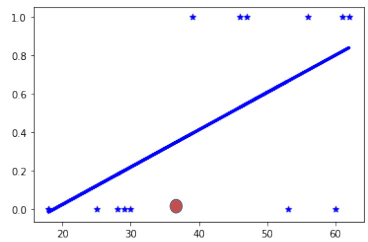


Age

Do threshold

Linear regression

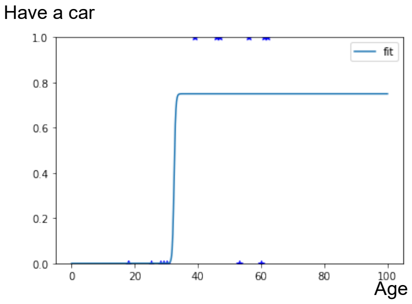
Have a car



Age

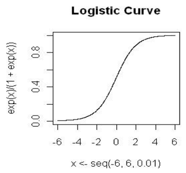
what if you have another sample

S-shape fitting is much better



Logistic regression is more stuiable as it is a classification problem. Logistic regression can even be negative or bigger than 1

Sigmoid/logsitic function



$$S(x) = \frac{1}{1+e^{-x}}$$

e = Euler's number ~ 2.71828

sigmoid functions converts input into range 0 to 1

Linear | Logistic Regression

Linear: $h = a * x + b$

Logistic Regression: $h = 1/(1+e^{-(a*x+b)})$

Parameters a and b
h = estimated probablity that y =1 on input x

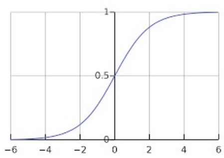
$$h = P(y=1 \mid x; a, b)$$

$$P(y=1 \mid x; a, b) + P(y=0 \mid x; a, b) = 1$$

Logistic Regression

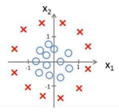
$h = 1/(1+e^{-(a*x+b)})$

if h > 0.5 , thats same that $a*x+b$ > 0



You can also have polymonia fuction

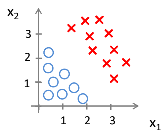
Non-linear decision boundaries



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

Logistic Regression

Decision Boundary



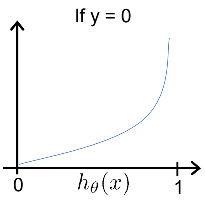
$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

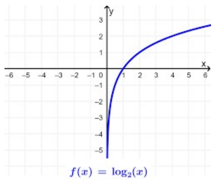
Predict "$y = 1$" if $-3 + x_1 + x_2 \geq 0$

Logistic Regression

Logistic regression cost function

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





Logistic Regression

Logistic regression cost function

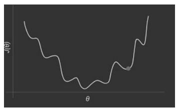
$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} \mathrm{Cost}(h_\theta(x^{(i)}), y^{(i)})$$
$$= -\frac{1}{m}\left[\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log\left(1 - h_\theta(x^{(i)})\right)\right]$$

To fit parameters θ :

$$\min_\theta J(\theta)$$

To make a prediction given new x :

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



non-convex

Logistic Regression

Gradient Descent

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

Want $\min_\theta J(\theta)$:

Repeat {

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

} (simultaneously update all θ)

Algorithm looks identical to linear regression!

Jupyter Notebook practice

Linear Regression

- Used to make **predictions** about an unknown event from known evidence
- **Output continuous**
- Inputs can be any level of measurement
- **Assumes linear relationship**
- Uses **least squares estimation**

Logistic Regression LOGIT

- Used to determine which variables affect the **probability** of a particular outcome
- **Output categorical**
- Inputs may be any level of measurement
- **Doesn't assume linear relationship but rather a logit transformation**
- Uses **maximum likelihood estimation**

Examining likelihood of event

- **Likelihood conventionally expressed on a scale of 0 to 1**
  Many health outcomes are dichotomous:
  Breast cancer=1 (yes) vs Healthy=0 (no)
- **Can be used to compare likelihood in groups:**
  Case vs controls
  Males vs females
  Chemo vs no chemo

Non-binary variables?

- A lot of categorical variables are not binary though, what can we do with these?
  - Often we can recode them to a binary response.
  - *Multinomial logistic regression.*

Multiclass classification

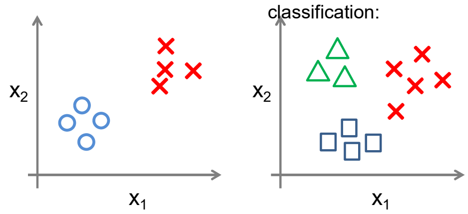
Email foldering/tagging: Work, Friends, Family, Hobby

Medical diagrams: Not ill, Cold, Flu

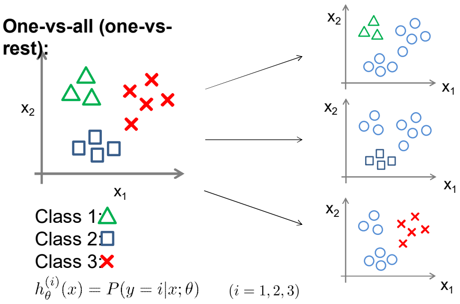
Weather: Sunny, Cloudy, Rain, Snow

Binary classification:

Multi-class classification:



One-vs-all (one-vs-rest):



$h_\theta^{(i)}(x) = P(y = i|x;\theta) \quad (i = 1, 2, 3)$

One-vs-all

Train a logistic regression classifier $h_\theta^{(i)}(x)$ for each class i to predict the probability that $y = i$.

On a new input x, to make a prediction, pick the class i that maximizes

$$\max_i h_\theta^{(i)}(x)$$

- https://www.w3schools.com/python/python_ml_logistic_regression.asp

Advanced optimization

Optimization algorithm

Cost function $J(\theta)$. Want $\min_\theta J(\theta)$.

Given θ, we have code that can compute $J(\theta)$
- $\frac{\partial}{\partial\theta_j} J(\theta)$ (for $j = 0, 1, \ldots, n$)
-

Gradient descent:

Repeat $\{$

$\theta_j := \theta_j - \alpha \frac{\partial}{\partial\theta_j} J(\theta)$

$\}$

Advanced optimization

Optimization algorithm

Given θ , we have code that can compute $J(\theta)$
- $\frac{\partial}{\partial\theta_j} J(\theta)$ (for $j = 0, 1, \ldots, n$)
-

Optimization algorithms:
- Gradient descent
- Conjugate gradient
- BFGS
- L-BFGS

Advantages:
- No need to manually pick α
- Often faster than gradient descent.

Disadvantages:
- More complex

Advanced optimization

Example:

$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$

$J(\theta) = (\theta_1 - 5)^2 + (\theta_2 - 5)^2$

$\frac{\partial}{\partial\theta_1} J(\theta) = 2(\theta_1 - 5)$

$\frac{\partial}{\partial\theta_2} J(\theta) = 2(\theta_2 - 5)$

Exercise

https://www.kaggle.com/datasets/giripujar/hr-analytics

1 Do analysis to figure out which factors have direct impact on employee rention
2 Plot bar charts showing impact of employee salaries on retention
3 Build logistic regression model using variables that were selected in step 1
4 Measure the accuracy