Machine Learning Crash Course

Week 3

I have always been convinced that the only way to get artificial intelligence to work is to do the computation in a way similar to the human brain. That is the goal I have been pursuing. We are making progress, though we still have lots to learn about how the brain actually works. ~ Geoffrey Hinton

Logistic Regression

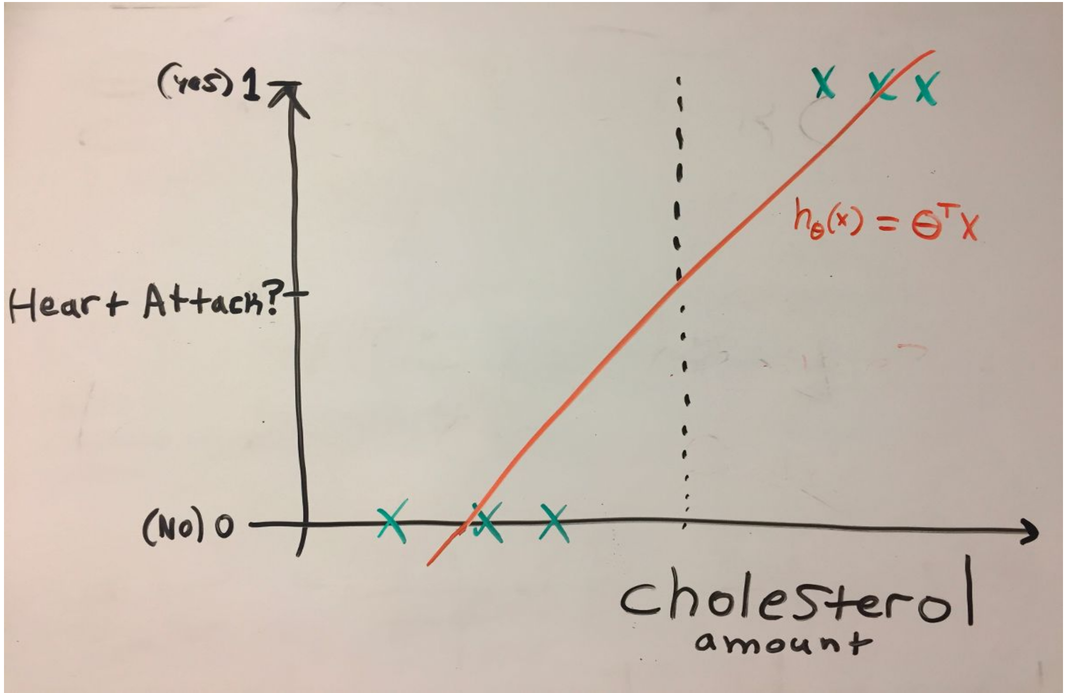
Today's Topics

- Classification
  - Binary Classification
  - Logistic Regression Hypothesis
  - Decision Boundary
- Logistic Regression Model
  - Cost Function
  - Gradient Descent
  - Optimization
- Multi-Class Classification
- Overfitting Problem
  - Definition
  - Adaptation of Cost Function
  - Application of Regularized Linear Regression
  - Application of Regularized Logistic Regression

1. Classification

Binary Classification

- Ex. Risk of Heart Attack



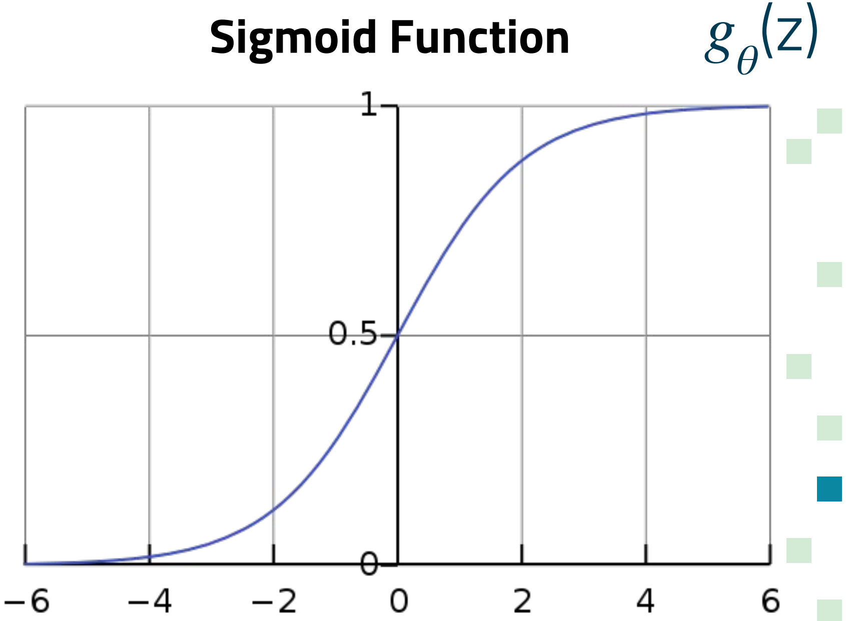
Binary Classification

Disadvantages of using Linear Regression:

- Prediction reliability dec. w/ the dec. of the gradient
- It could be that h_θ(x) can be larger than 1 or less than 0
  - Therefore it's best to use Logistic Regression

Logistic Regression Hypothesis

- Objective: $0 \le h_\theta(x) \le 1$
- $h_\theta(x) = g(\theta^T x) = g_\theta(z)$
- $h_\theta(x) = 1 \;/\; 1 + \exp(-\theta^T x)$



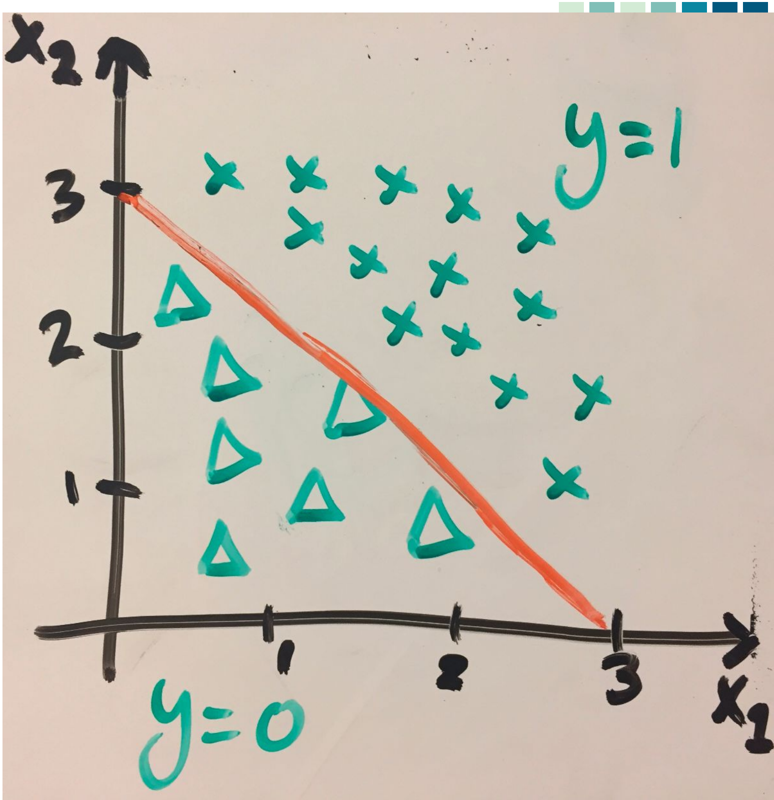
Logistic Regression Hypothesis

Interpretation of Output

- $h_\theta(x)$ is the probability between 0 and 1 for some predict / output being 1
  - We say that if $h_\theta(x)$ outputs some value ≥ 0.5 then we claim the output to be classified as 1 else we claim 0
- $\underline{x} = [x_0;x_1]=[1;\text{cholesterol }\#_s]$

Decision Boundary

- Goal is to properly bind your output's Max and Min
- Ex. $h_\theta(x)=g(\theta_0+\theta_1 x_1+\theta_2 x_2)$
- Let $\theta_0=-3$, $\theta_1=1$, $\theta_2=1$
- Predict y=1 for $-3+x_1+x_2\geq 0$

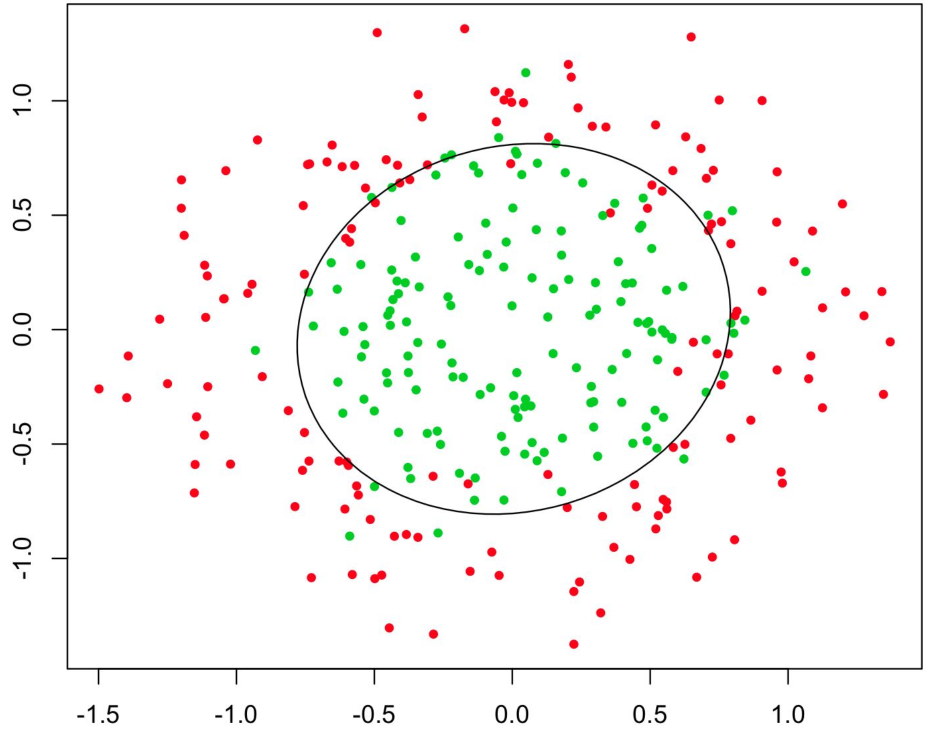


Decision Boundary

Non-Linear Decision Boundaries

- Ex. $h_\theta(x)=g(\theta_0+\theta_1 x_1+\theta_2 x_2 +\theta_3 x_3^2+\theta_4 x_4^2)$
- Let $\theta_0=-1$, $\theta_1=0$, $\theta_2=0$, $\theta_3=1$, $\theta_4=1$
- Predict y=1 for

  $-1+x_1^2+x_2^2 \geq 0$



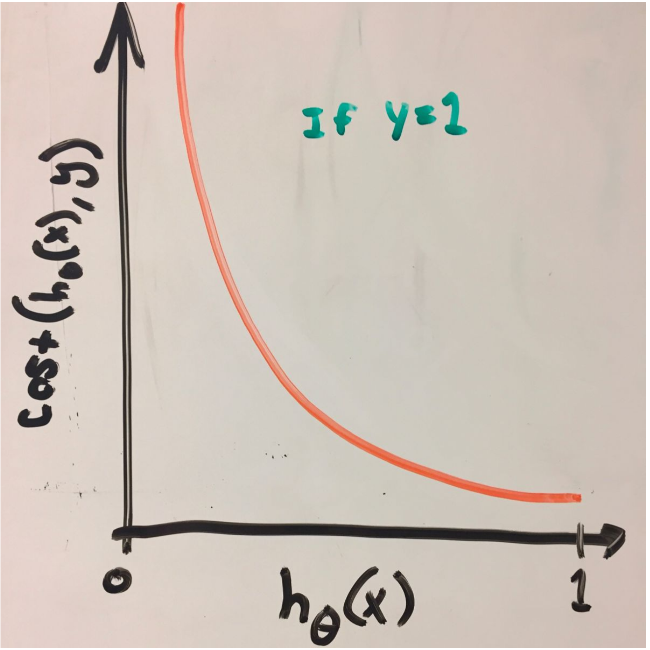
2. Logistic Regression Model

Cost Function

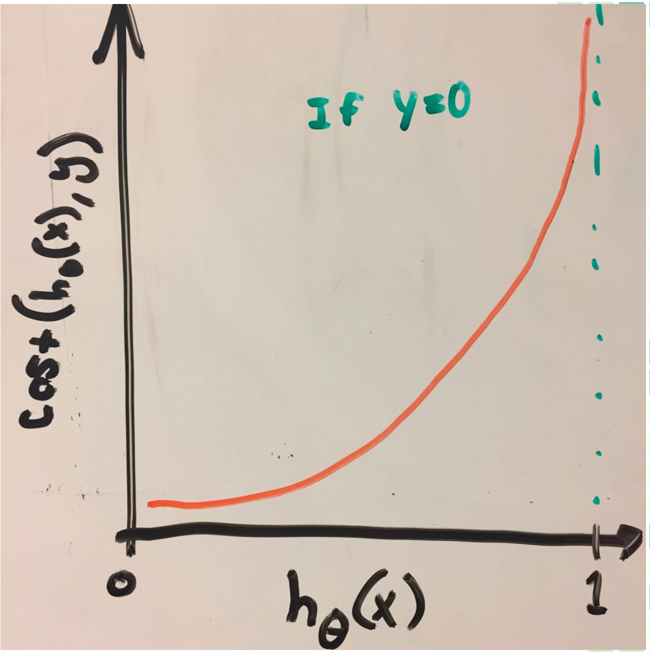
- Assume: $h_\theta(x) = 1 / 1 + e^{-\theta Tx}$
- Let the training set **S** equal the Cartesian Product of the set **X** and set **Y** whereas:
  - **X** represents the examples
  - **Y** represents what's being predicted e.g. {0,1}
- We'll Use:
  - $Cost(h_\theta(x), y) = \{-\log(h_\theta(x))$ if $y=1$ and $-\log(1-h_\theta(x))$ if $y=0$

Cost Function

If y=1



If y=0



Cost Function

- $J(\theta)=(1/m)\Sigma Cost(h_\theta(x^i),y^i)$ for i=1 to m
- Simplified to:
  - $J(\theta)=(1/m)(\Sigma y^i * log(h_\theta(x^i))+(1-y^i)*log(1-h_\theta(x^i))$ for i=1 to m)

Gradient Descent

- Now replace $h_\theta(x^i)$ with 1 / 1 + $e^{-\theta Tx}$

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

Advanced Optimization

- Suggestions:
  - Conjugate Gradient
  - BFGS
  - L-BFGS

- Advantages:
  - No picking alpha
  - Faster

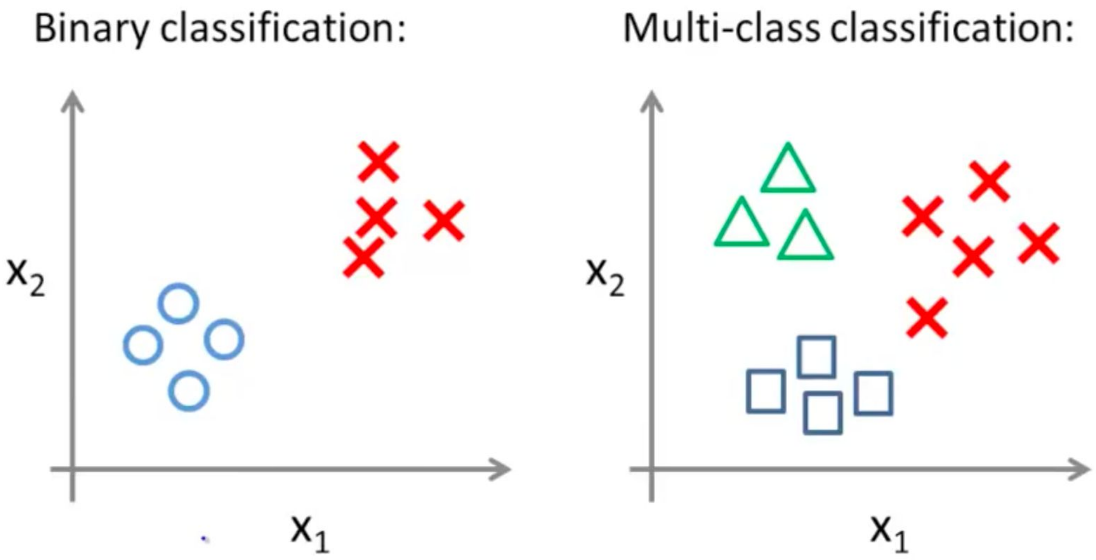
- Disadvantages:
  - More complex
  - Better to utilize a pre-built library

3. Multi-class Classification

Cost Function

Consider:

- Training a logistic regression classifier $h_\theta(x^i)$ for each class i
- Goal:
  - For every new input x, pick the class that maximizes $h_\theta(x^i)$ to make a prediction



4. Overfitting Problem

Definition

Overfitting is simply defined as the model fitting to the “training” data extremely well, but not being able to generalize to “new” data.

Potential Solutions

- Reduce the amount of features
  - Goal: To select the features to keep
- Regularization
  - Goal: To reduce the magnitude or values of θ_j

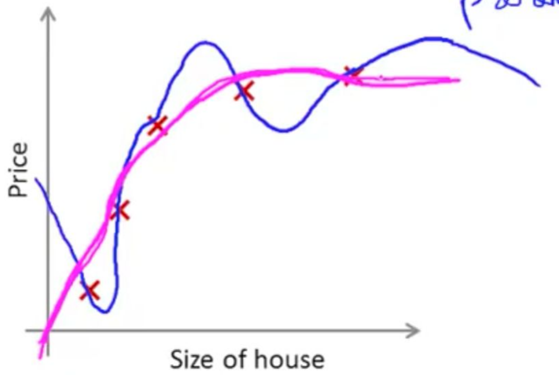
Adaptation of the Cost Function

- Select small theta values
- Add regularization

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

regularization parameter

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



Regularized Linear Regression

- Rewrite the Gradient Descent Equation to:
  - $\theta_j = \theta_j * (1 - \alpha * (\lambda/m)) - (\alpha/m) * (\Sigma((h_\theta(x^i) - y^i)) * h_\theta(x_j^i)$ for $i = 1$ to $m)$
- Normal Equation to min($J(\theta)$):
  - $\underline{\theta} = (\underline{X}^T \underline{X} + \lambda * ((\partial/\partial\theta_j)J(\theta)))^{-1} * X^T y$

Regularized Logistic Regression

- Add to the cost function $(\lambda/2m) * (\Sigma\theta_j^2$ for j=1 to n) as shown below
- $J(\theta)=(-1) * (1/m) * (\Sigma y^i * \log(h_\theta(x^i)) + (1-y^i) * \log(1-h_\theta(x^i))$ for i=1 to m)

  $+ (\lambda/2m) * (\Sigma\theta_j^2$ for j=1 to n)
- Rewrite the Gradient Descent Equation to:
  - $\theta_j=\theta_j - (\alpha/m) * (\Sigma((h_\theta(x^i)-y^i)) * h_\theta(x_j^i) + (\lambda/m) * \theta_j$ for i=1 to m)

Sources

- https://stats.stackexchange.com/questions/212965/how-to-achieve-a-nonlinear-decision-boundary
- https://hackernoon.com/introduction-to-machine-learning-algorithms-logistic-regression-cbdd82d81a36
- https://www.ritchieng.com/logistic-regression/#4c-regularized-logistic-regression
- https://machinelearningmastery.com/logistic-regression-for-machine-learning/
- http://courses.washington.edu/css490/2012.Winter/lecture_slides/05b_logistic_regression.pdf
- http://www.cs.cmu.edu/~pradeepr/convexopt/Lecture_Slides/conjugate_direction_methods.pdf
- http://www.seas.ucla.edu/~vandenbe/236C/lectures/qnewton.pdf
- https://cs.nyu.edu/overton/mstheses/skajaa/msthesis.pdf