Lecture 3: Logistic Regression and Optimization

Areeb Gani, Michael Ilie, Vijay Shanmugam

Welcome!

ml.mbhs.edu

Outline

Topics

- Linear Regression Recap
- Turning Regression into Classification
- Sigmoid Function
- Logistic Regression
- Gradient Descent Contours

Deepnote!

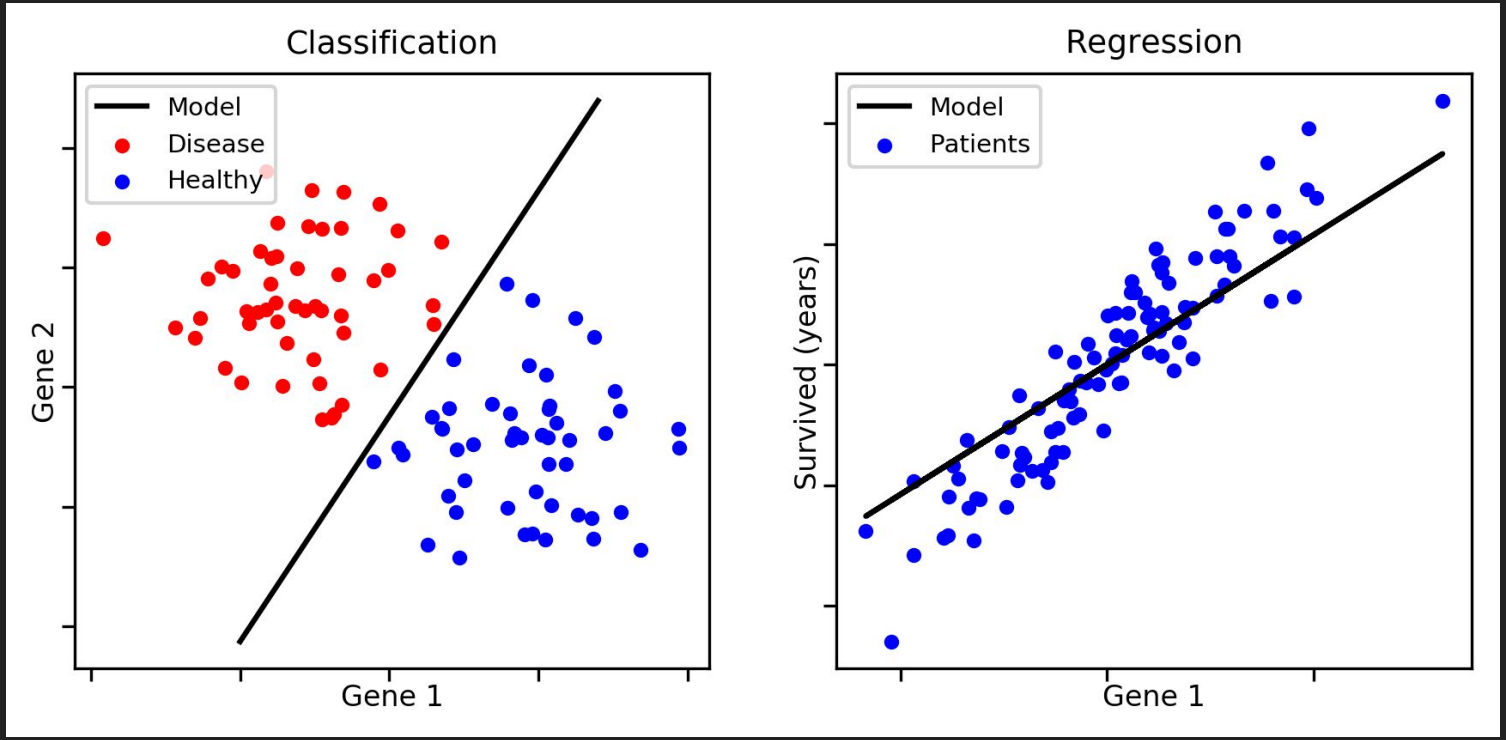
Regression vs. Classification

- Regression
  - Predicting continuous values
  - Example: predicting house prices, body weight, height
  - Types of algorithms: linear regression, polynomial regression, exponential regression
- Classification
  - Predicting discrete label values
  - Example: predicting if tumor is benign or malignant, if car is new or used, if dog is of a certain breed
  - Types of algorithms: logistic regression, k-nearest neighbors, decision trees

Linear Regression Recap

- Predict values (m, b) in linear equation
  - We call this (W, b)
- Create cost function to tell us difference between our prediction and the real value
- Minimize the cost function using gradient descent
- Now we have the optimal (W, b) values and have fit our line to the data

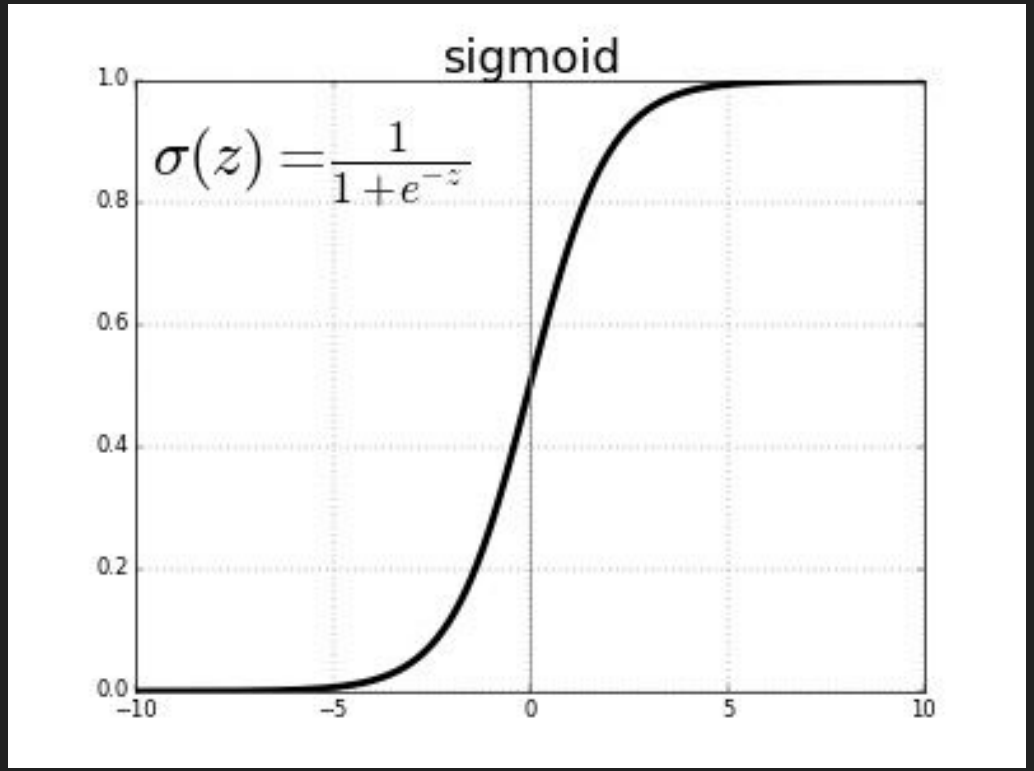
Regression vs. Classification



Turning Regression into Classification

- To perform classification, we need probability values
- In our example, we need the probability that the tumor is malignant
- A probability is a value from [0, 1]
- How do we get such a value?

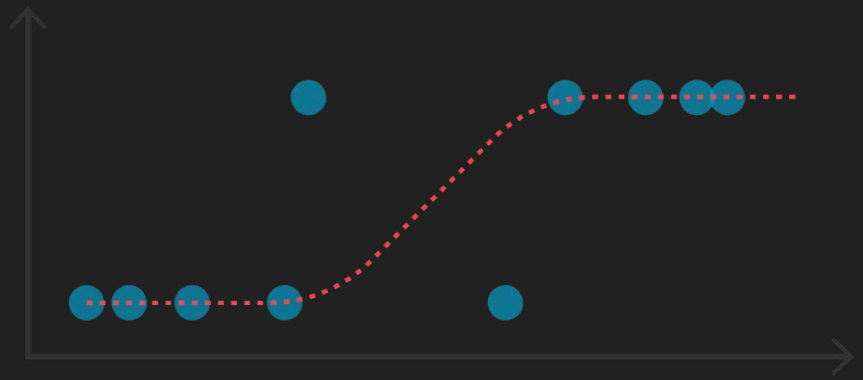
Sigmoid Function



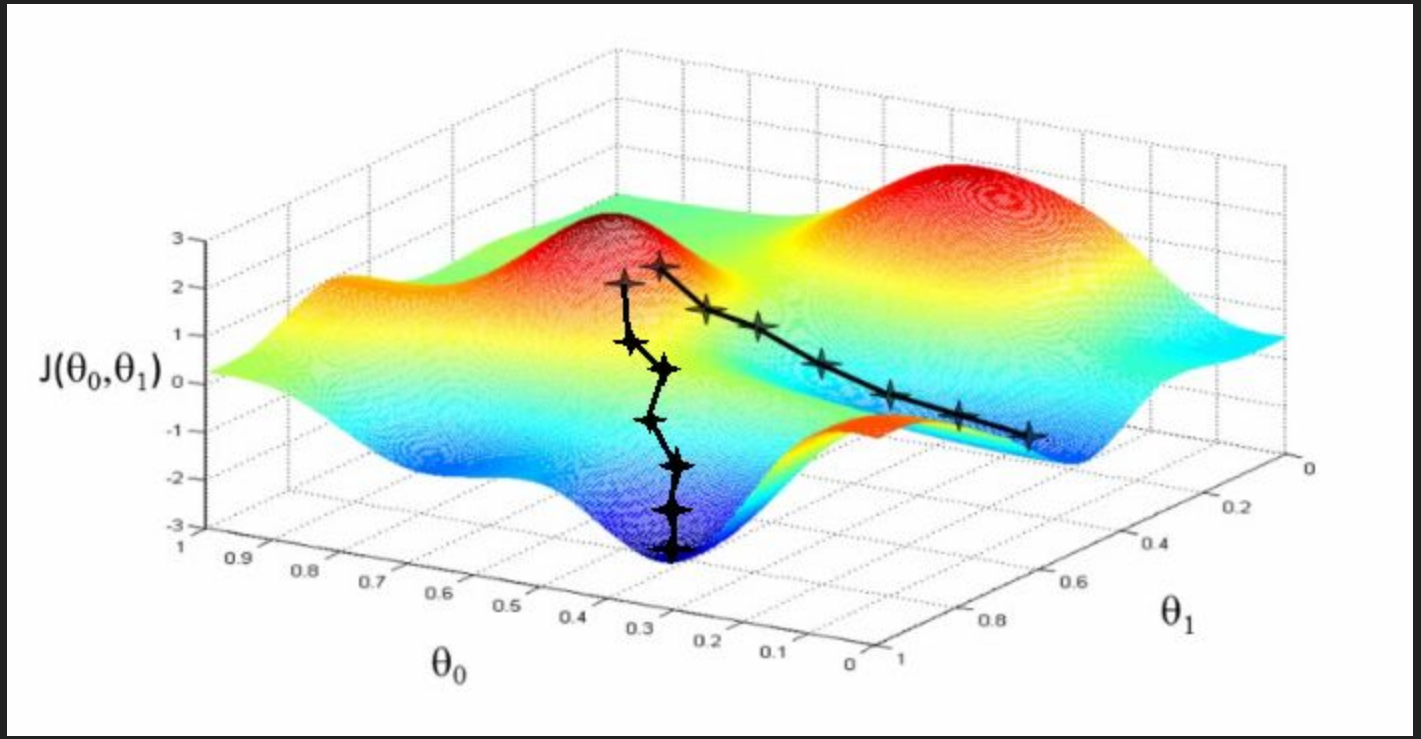
- Range (input) → (-inf, inf), Domain (output) → [0, 1]

Logistic Regression

- Perform linear regression, then apply sigmoid function
- We call sigmoid function the “activation function”
- This gives us a probability → class label
- Generally, threshold is 0.5, but this level can be adjusted



Cost Function Graphs

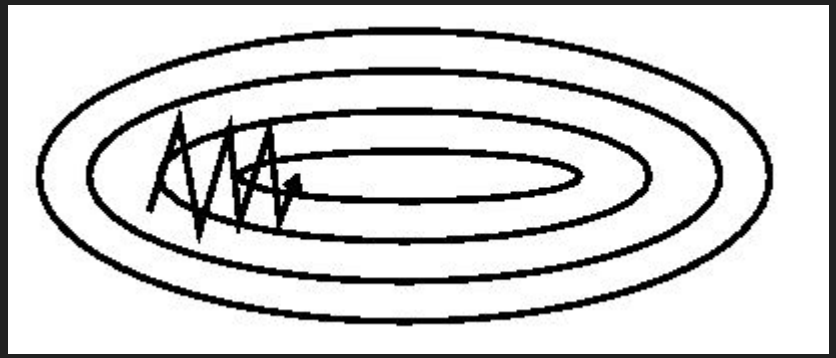


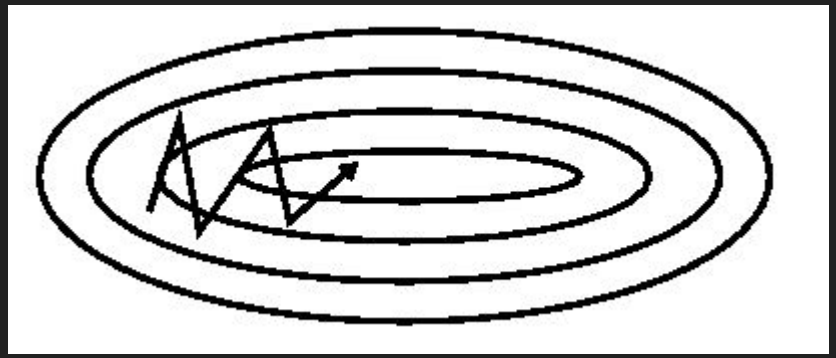
Cost Function Graphs

Types of Gradient Descent

- **Batch gradient descent** (training on whole dataset)
- **Stochastic gradient descent** (training on individual entries)
- **Mini-batch gradient descent** (training on subsets of dataset)
    - Must define *batch size parameter* (size of each batch), e.g. in a dataset of 10000 rows we may have a batch size of 512

Gradient Descent with Momentum





- Minimize movement vertically, maximize movement horizontally
- Take exponential average of previous changes (derivatives), smoothing out our movement (concept of “momentum”)
- Since we are moving up and down vertically, this smoothes out to minimal vertical movement, whereas horizontal movement remains high

RMSProp and Adam Optimization

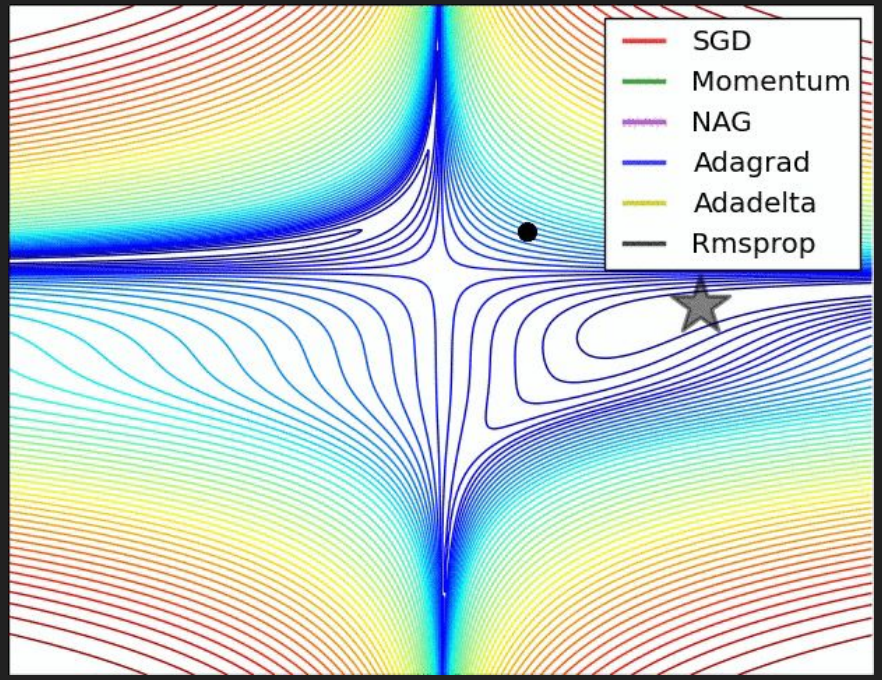
RMSProp

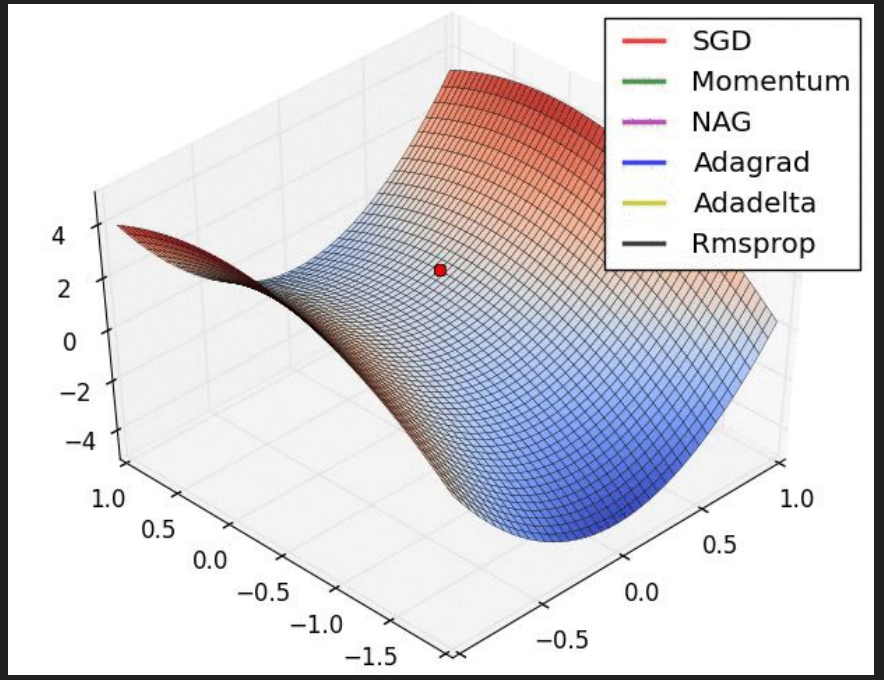
- Adaptive learning rate, instead of having constant
- Ensures that we are not too slow (undershoot), and not too fast (overshoot)

Adam Optimization

- Combination of Momentum and RMSProp (moves efficiently, adaptive learning rate)

Visual Comparisons





Visual Comparisons

Join Our Groups

- Sign up for Discord (https://discord.gg/3Z5YuPqt)
- Join Deepnote (https://deepnote.com/join-team?token=af3af0284bc8497)
- Fill out our form (https://forms.gle/Fr31aFLWx8cHdtTY8)
    - Join mailing list + Github organization