Lecture 2: Linear and Logistic Regression

Areeb Gani, Michael Ilie, Vijay Shanmugam

Welcome!

ml.mbhs.edu

Outline

Topics

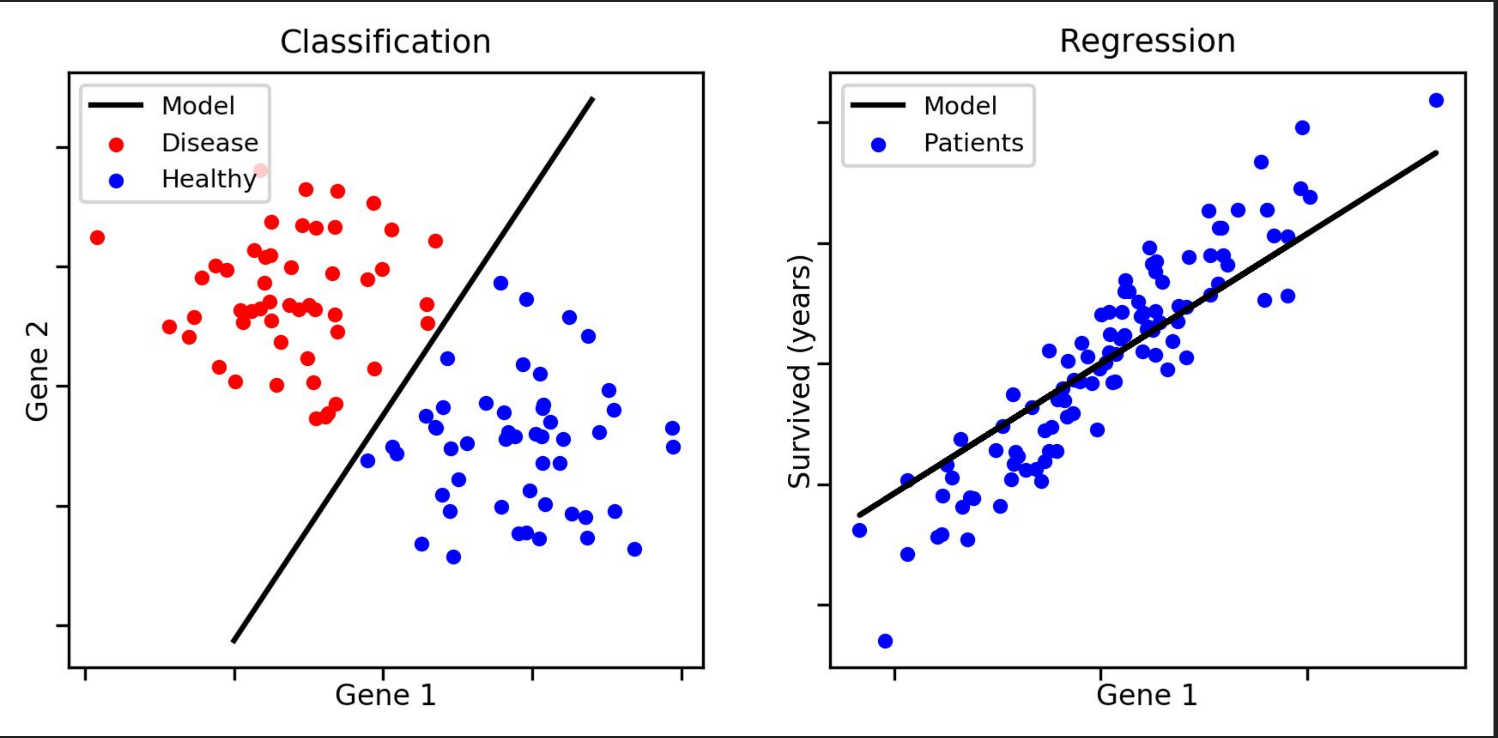
- Regression vs. Classification
- Linear Equation
- Linear Regression
- 2-Dimensional Examples
- Machine Learning Approach
- Hypothesis Function
- Cost Function
- Gradient Descent
- Linear Regression Recap
- Turning Regression into Classification
- Sigmoid Function
- Logistic Regression

Deepnote!

Regression vs. Classification

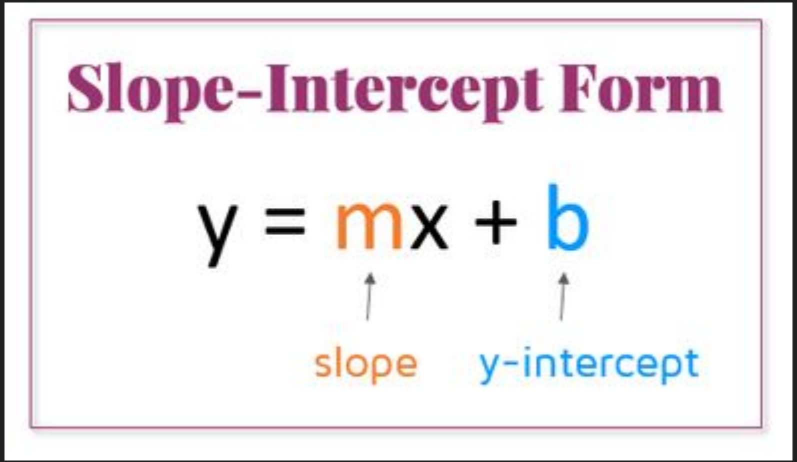
- Regression
  - Predicting continuous values
  - Example: predicting house prices, body weight, height
  - Types of algorithms: linear regression, polynomial regression, exponential regression
- Classification
  - Predicting discrete label values
  - Example: predicting if tumor is benign or malignant,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

Regression vs. Classification



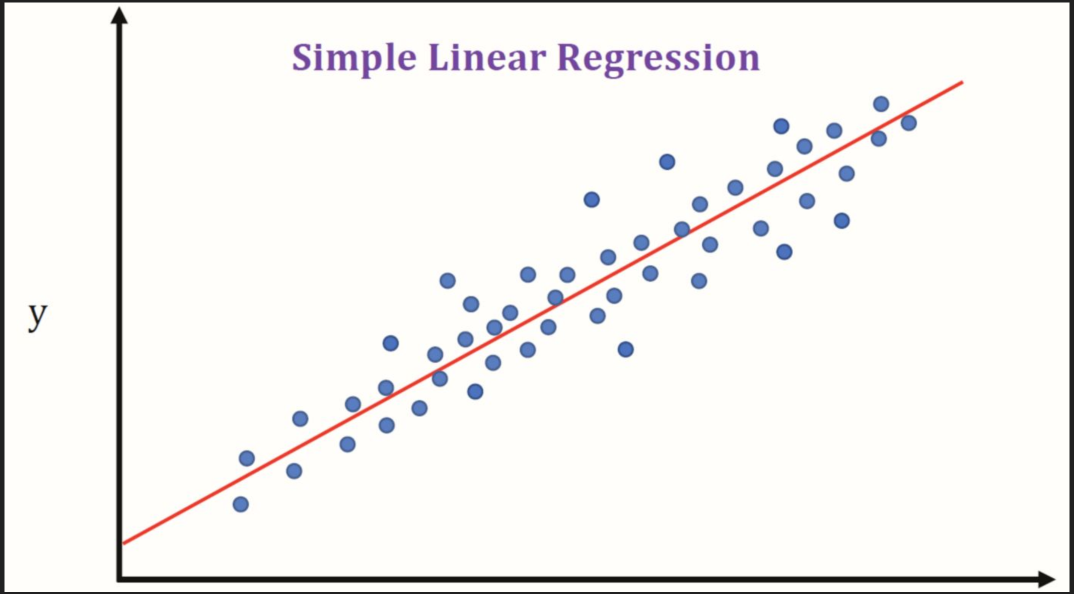
Linear Equation

- Linear equations of the form y = mx + b



Linear Regression

- Linear Regression is trying to predict a linear relationship between data points
    - Image below is 2D, but we can generalize to n-dimensions
- We need to predict the *m* and *b* variables to get an accurate prediction



Example

```
x = [1,  2,  3,  4,  5,  6,  7]
y = [1,  4,  7, 10, 13, 16, 19]
```

Solution for y = mx + b?

(m, b) = (3, -2)

Real-Life Example

- Iris Dataset
  - Predicting sepal length from other flower measurements
- Penguin Dataset
  - Predicting flipper length

	sepal_length float64	sepal_width float64	petal_length float64	petal_width float64	species object
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5	3.6	1.4	0.2	setosa

5 rows × 5 columns

Machine Learning Approach

- Trial and error - but how does the algorithm learn from its mistakes?

Input variable - x

Target variable - y

Predicted variable - $\hat{y}$

(Subscripts represent i-th item)

Weight - w , Bias - b, Length - m

Hypothesis Function

- Hypothesis function represents our prediction
- In univariate (one variable/feature) linear regression, our hypothesis function is:

$$h(x) = wx + b$$

$$\hat{y} = wx + b$$

Cost Function

- The cost function represents the difference between our target and prediction

$$\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i) \qquad \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$$

- Our goal is to maximize our accuracy → minimize our cost function

Cost Function

```
y     = [1, 2, 3, 4]
y_hat = [2, 3, 3, 4]
```

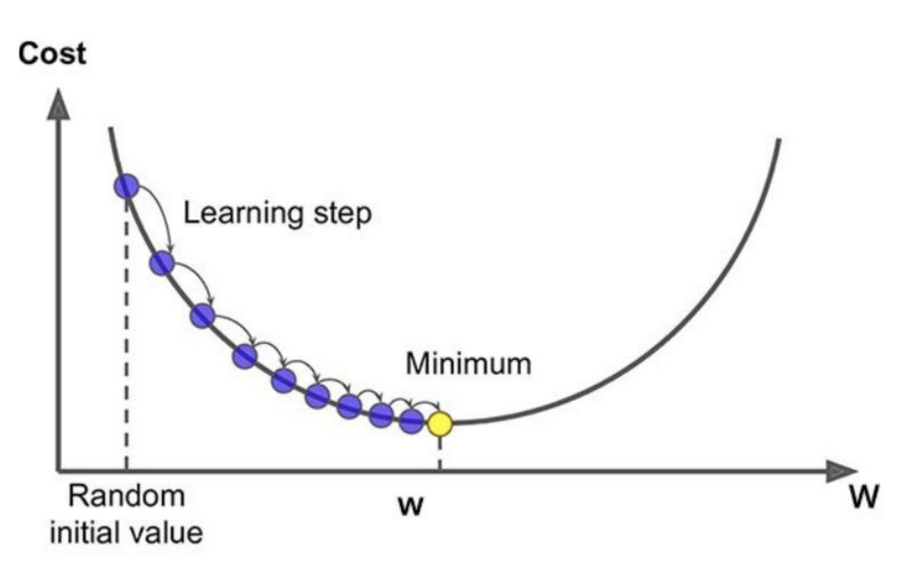
Cost value in this scenario?

$$\frac{(2-1)^2 + (3-2)^2 + (3-3)^2 + (4-4)^2}{4} = \frac{1}{2}$$

Gradient Descent

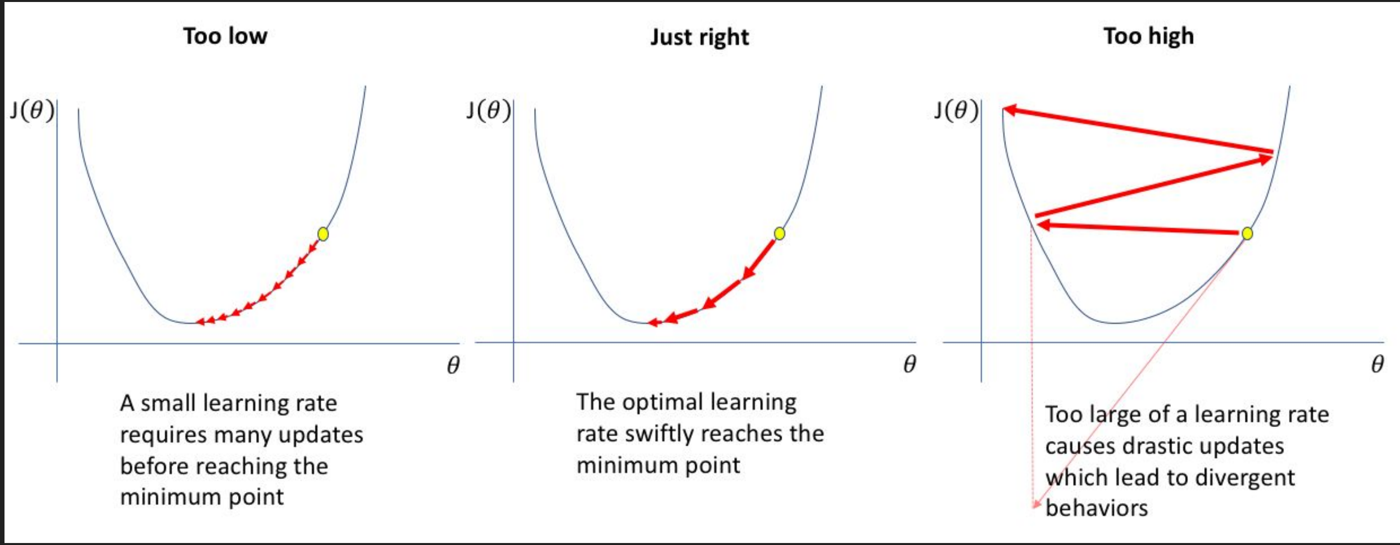
$$\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$$

- Gradient descent is the most fundamental optimization function
    - Method to minimize the cost function



Gradient Descent (value updates)

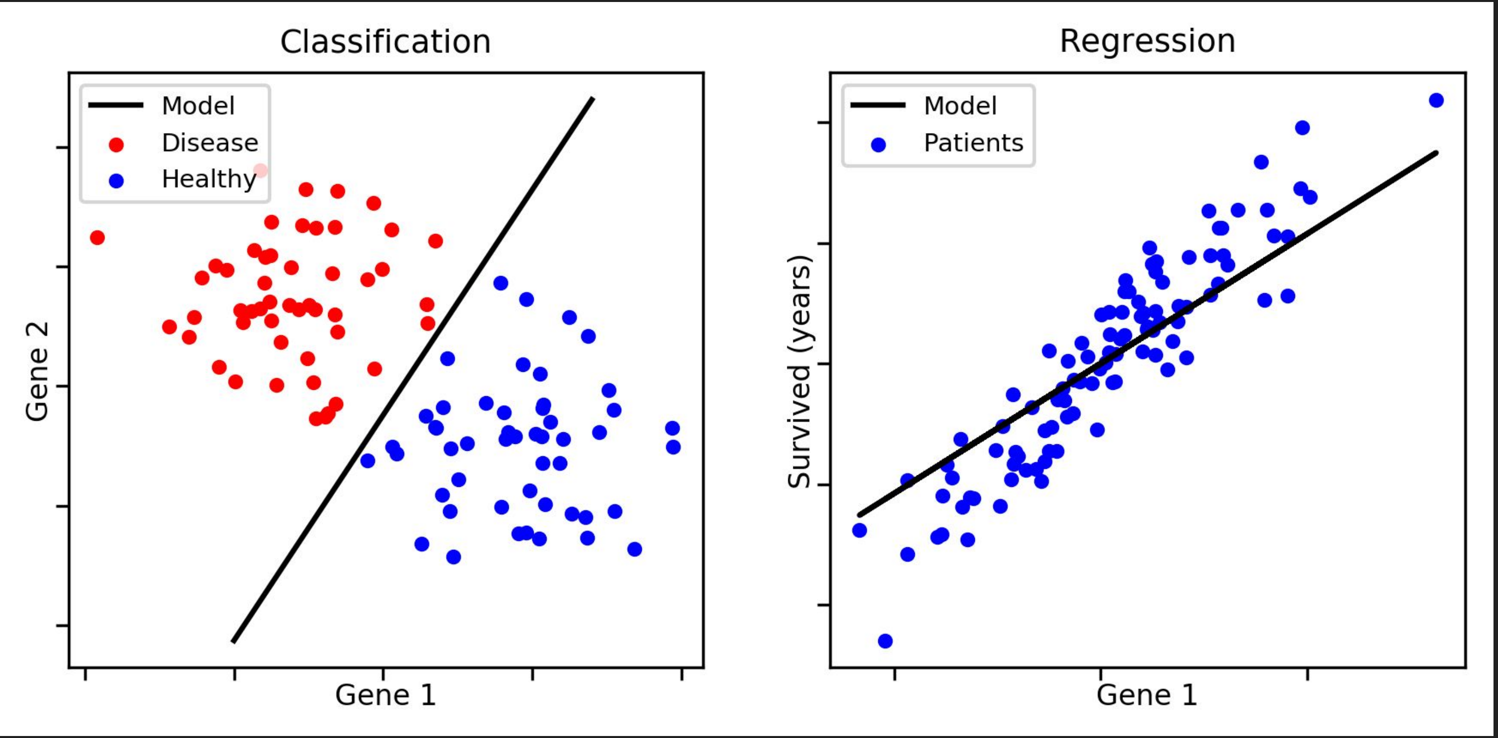
- Learning rate **too low**: takes too long
- Learning rate **too high**: will diverge
- Learning rate **perfect**: reaches minimum



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

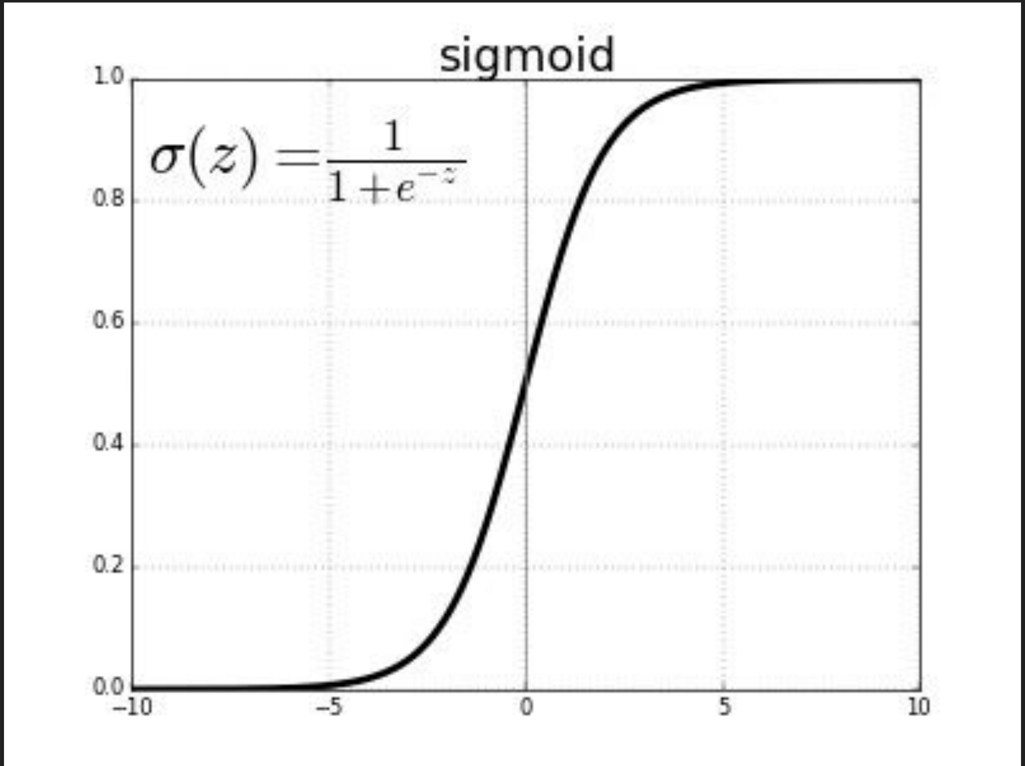
Regression vs. Classification



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?

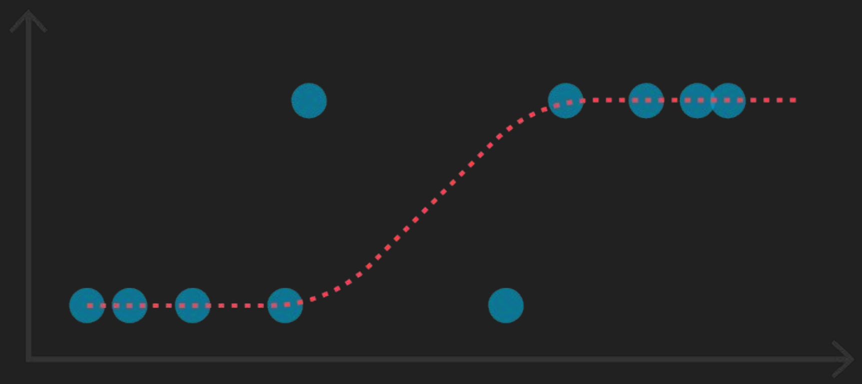
Sigmoid Function



- 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



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