5CS037-Concepts and Technologies of AI

Lecture-09

Supervised Machine Learning.

Linear Methods for Classification.

Siman Giri

Story So Far….

Remember!!!: Components of Machine Learning.

- **Dataset:**
  - Labelled vs. Unlabeled Dataset.
- **A Decision Process (Representation/Model):**
  - Machine learning algorithms(Models) are used to make inference or estimate of an output based on input data – labeled or unlabeled.
- **An Error Function (Evaluation):**
  - A performance metric used to evaluate the estimate of a model.
  - Metrics depends on types of learning (supervised or unsupervised) and types of task (Classification or Regression)
- **An model Optimization Process:**
  - An automated algorithm or process used to update parameters of machine learning models until threshold or accepted evaluation metric has been achieved

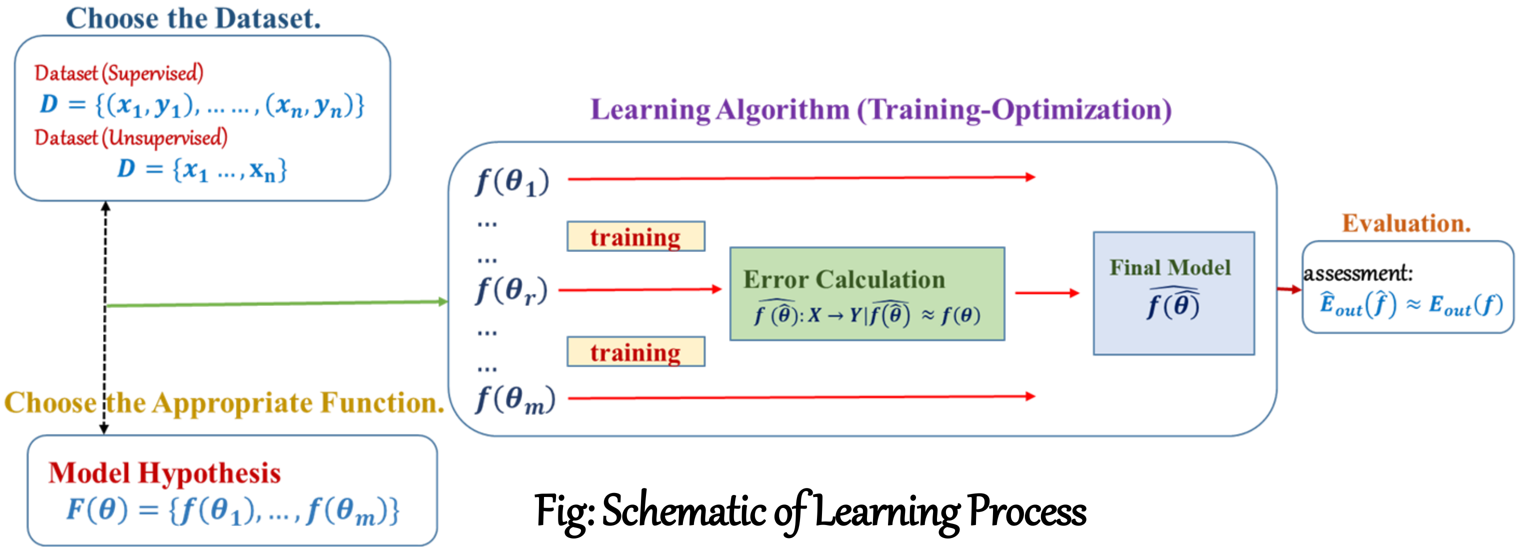
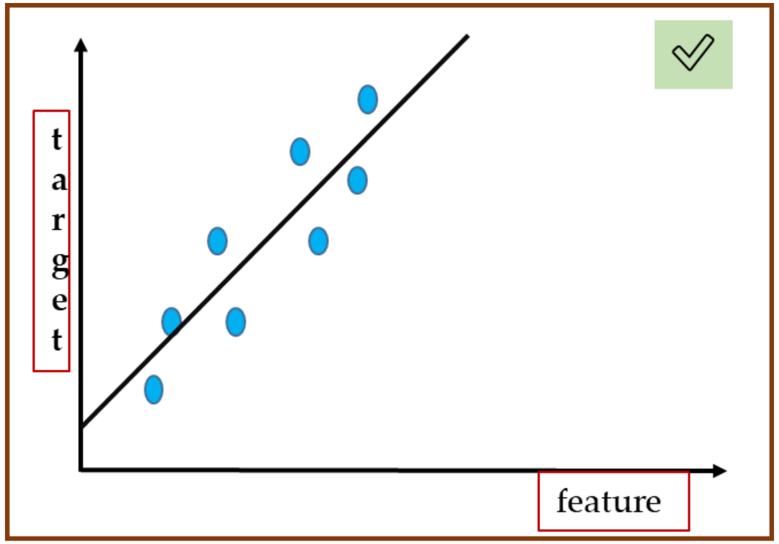


Fig: Schematic of Learning Process

Regression So far…..



- **Assumptions:**
  - Exist a liner relationship → $\{Y = w_0 + W^T X.\}$
- **Simple linear regression:**
  - Relationship between {**one**} **numerical response/independent/feature**{**X**} and {**a**} **numerical predictor/dependent/target**{**Y**}.
- **Multiple linear regression:**
  - Relationship between {**multiple**} **numerical response/independent/feature**{**X**} and {**a**} **numerical predictor/dependent/target**{**Y**}.

Classification.

1. Introduction.

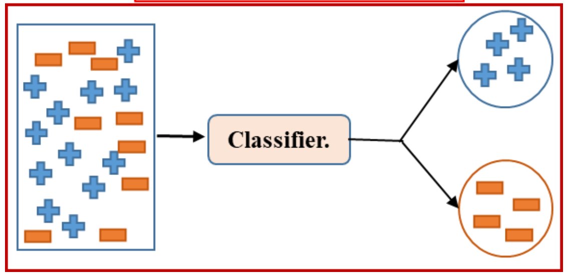
1.1 Classification: Motivation.

- **Classification problems** occur often, perhaps **even more** so than **regression problems**. Some examples include:
  - A person arrives at the emergency room **with a set of symptoms** that could possibly be attributed to **one of three medical conditions**.
    - Which of **the three conditions** does the **individual have**?
  - An **online banking service** must be able to determine **whether or not** a **transaction** being performed on the site is **fraudulent**, on the **basis of the user's IP address**, past **transaction history**, and **so forth**.
- Up to this point, **the methods** we have seen have **centered around modeling** and the **inference** of a **quantitative response variable** (ex: House prices).
  - **Linear regression** perform well under these situations
- When the **response variable is categorical**, then the problem is no longer called a regression problem but is instead labeled as a **classification problem**.
- The goal is to attempt to classify each observation into a category (aka, class or cluster) defined by Y, based on a set of predictor variables X.

1.2 Classification: Definition.

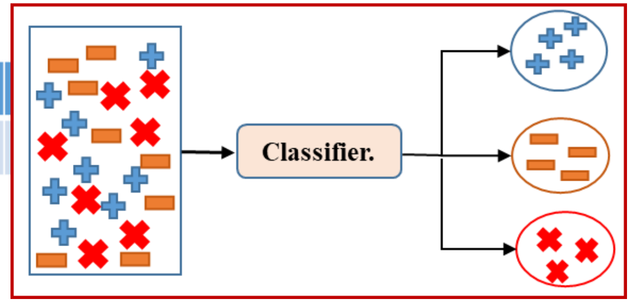
- In classification:
  - we take an input vector $\{x^1, \dots, x^d\} | X \in \mathbb{R}$ and **assign** it to one of K **discrete classes or groups** C_k where $k = 1, \dots, k$.
- In most common cases, **the classes are taken to be disjoint**, so that each **input is assigned to one and only one class**.
- There can be multiple scenario for the label space C.

Binary Classification.



Binary Classification	$c = \{0\ or\ 1\}$
Multi Class Classification	$c = \{1,2,..k\}(k \geq 2)$

Multi-class Classification.

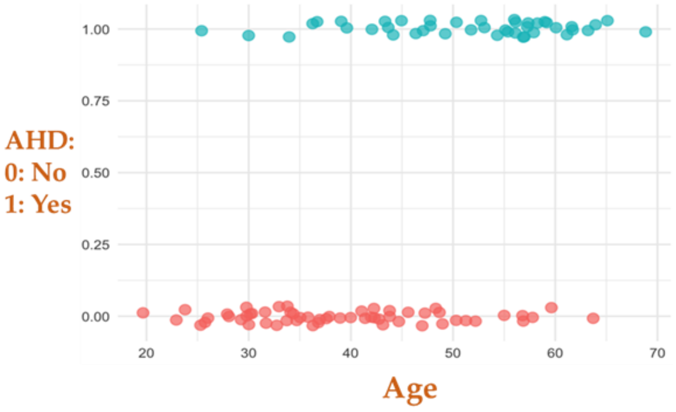


1.3 Classification: Example – Binary Classification.

- Given a pair of dataset: (X, Y)
  - $X \coloneqq x_i^1 \coloneqq$ is a feature vector $\coloneqq$ Age.
  - $Y \coloneqq y_i \coloneqq$ is a target scalar $\coloneqq$ Acute Heart Disease
- A categorical variable y could be encoded to be quantitative: For example:
  - $y = \begin{cases} 0 \text{ } if \text{ } y = No \\ 1 \text{ } if \text{ } y = Yes. \end{cases}$

Age	Heart Disease
63	No
67	Yes
67	Yes
…….	……..
37	No
41	No

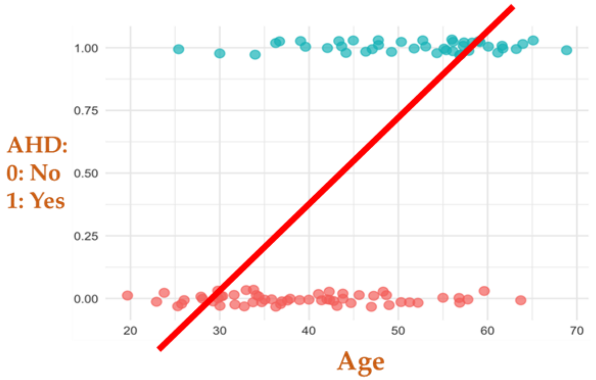
- Our Objective is to assign a class Yes or No to our Feature Space Age.
- Let's Explore the data:



- Can we fit a line?
  - What happens if we use linear regression to predict this?

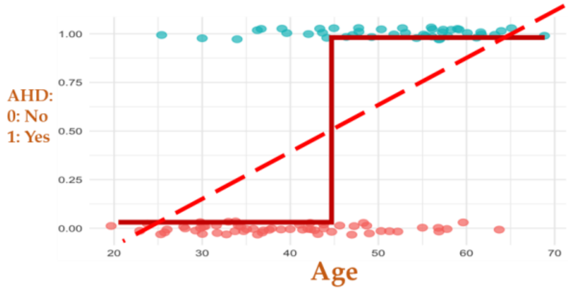
1.4 Classification: Example – Why not Linear Regression?

- If we use linear regression:



- A linear regression could be used to predict ***y* from *x***. What would be wrong with such a model?
  - The model would imply a specific ordering of the outcome, and would treat **a one-unit change in *y* equivalent.**
  - The jump from ***y* = 1 to y = 2** should not be interpreted as the same as a jump from ***y* = 2 to *y* = 3.**
  - Similarly, the response variable could be reordered such that ***y* = 0 represents Yes** and ***y* = 1 represents No**, and then the model estimates and predictions would be **fundamentally different**.

- One idea to try to solve the issues from the regression line would be to set some threshold "T" such as:
  - $\hat{y}_i = \begin{cases} 1 \; if \; w_0 + W^T X \; \geq T \\ 0 \; if \; w_0 + W^T X \; < T \end{cases}$



- If we apply a non linear transformation to aforementioned equation:
  - $\hat{y}_i = \begin{cases} 1 \; if \; sign(w_0 + W^T X) \geq 0 \\ 0 \; if \; sign\,(w_0 + W^T X) < 0 \end{cases}$
    - **Can we find such function?**

Logistic Regression for Binary Classification

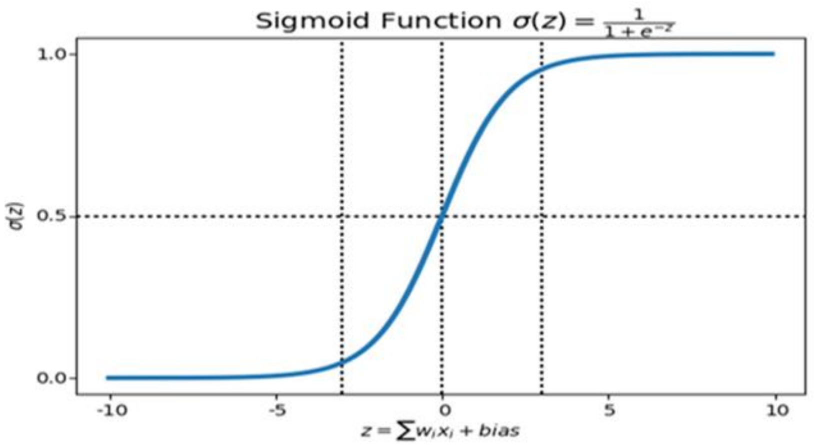
2. Component 1: A Decision Process.

2.1 Logistic Regression: The Logistic Model.

- {Disclaimer!!
  - Throughout this section we will assume that the outcome has two classes, for simplicity.
  - We return to the general K class setup at the end.}
- Setting Logistic Regression for binary classification:
  - **Here:** $D = \{x_i, y_i\}_{i=1}^{m}$. **Where** $x_i \in \mathbb{R}^d, y_i \in \{0, 1\}$.
- Logistic regression starts with different model setup than linear regression:
  - instead of modeling **Y** as a function of **X** directly,
    - we model the probability that **Y is equal to class 1**, **given X** i.e.
      - $P(Y = 1|X) = \frac{\exp^{W^T X}}{1+\exp^{W^T X}}$
- The function on the right-hand side above is called the **sigmoid of** $W^T X$.(a.k.a **logistic function**)

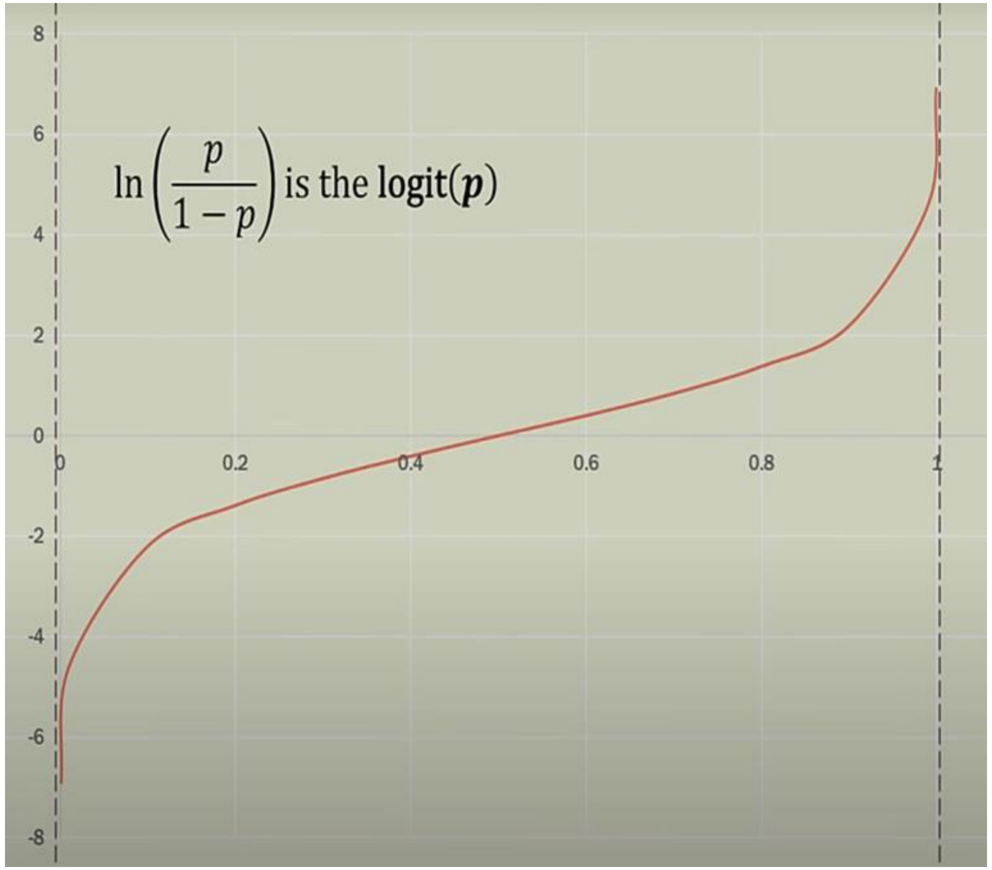
2.2 Logistic Regression: The Logistic Function.

- Logistic/Sigmoid function:
  - The logistic function $\boldsymbol{\sigma}$ is a function from the real line to the **unit interval (0,1)**
    - $\boldsymbol{\sigma(t) = \frac{1}{1+e^{-t}} = \frac{e^t}{1+e^t} \quad -\infty < t < \infty}$
  - The function maps any real value into another value between 0 and 1.
  - In machine learning, we use sigmoid to map predictions to probabilities.
  - Properties:
    - Range: $\boldsymbol{0 < \sigma(t) < 1}$.
    - **Inverse:** $\boldsymbol{t = \sigma^{-1}(p) = \ln\left(\frac{p}{1-p}\right)}$ **: logit function.**
    - Derivative: $\boldsymbol{\frac{d}{dt}\sigma(t) = \sigma(t)\left(1 - \sigma(t)\right) = \sigma(t)\sigma(-t)}$



2.3 Logistic Regression: The logit function.

- Rewriting our logistic equation→ {A Decision Process}:
  - $P(Y=1|X) = p(x) = \frac{exp^{W^TX}}{1+exp^{W^TX}}$
- In terms of logit function:
  - $\ln\left(\frac{p(x)}{1-p(x)}\right) = W^TX.$ → {A Decision Process}
- If we Visualize the *logit* Function: Interesting Observation.
  - When $p = 0;$
    - $ln\left(\frac{0}{1}\right) = ln(0) = (-\infty)$
  - When $p = 1;$
    - $ln\left(\frac{1}{0}\right) = (\infty)$
  - When $p = 0.5;$
    - $ln\left(\frac{0.5}{0.5}\right) = ln(1) = 0$



2.4 Decision Boundary: Logistic Regression.

- Suppose that we have formed the estimate $\boldsymbol{W}$ of the logistic coefficients, as discussed in the last section.
- To predict the outcome of a new input $\mathbf{x} \in \mathbb{R}^{\boldsymbol{d}}$, we form:
  - $\widehat{\boldsymbol{p}(\boldsymbol{x})} = \frac{\exp^{W^T X}}{1+\exp^{W^T X}}$;
- and infer the associated class according:
  - $\widehat{\boldsymbol{f}(\boldsymbol{x})} = \begin{cases} \mathbf{0}, & \widehat{\boldsymbol{p}(\boldsymbol{x})} \leq \mathbf{0.5} \\ \mathbf{1}, & \widehat{\boldsymbol{p}(\boldsymbol{x})} > \mathbf{0.5} \end{cases}$
- Equivalently for logits :
  - $\mathbf{logit}\, \widehat{\boldsymbol{p}(\boldsymbol{x})} = \boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{x}$
- and infer the associated class according:
  - $\widehat{\boldsymbol{f}(\boldsymbol{x})} = \begin{cases} \mathbf{0} & \widehat{\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{x}} \leq \mathbf{0} \\ \mathbf{1} & \boldsymbol{w}^{\boldsymbol{T}} \widehat{\boldsymbol{x}} > \mathbf{0} \end{cases}$

Logistic Regression for Multiclass Classification.

3. Component 1: A Decision Process.

3.1 Logistic Regression: Multiclass Classification.

- For example: predicting 3+ classes.
- There are several extensions to standard logistic regression when the response variable ***Y*** has more than **2 categories**. The two most common are:
  - ordinal logistic regression
  - multinomial logistic regression

3.2 Multinomial Logistic Regression: Approach-1.

- The first approach sets one of the class in the response variable as the *reference* group, and then fits separate logistic regression models to predict the other cases based off of the reference group.
- For example:

$$y = \begin{cases} \mathbf{1}\ \boldsymbol{class\ A} \\ \mathbf{2}\ \boldsymbol{class\ B} \\ \mathbf{3}\ \boldsymbol{class\ C} \end{cases}$$

- We could select the $y = 3$ (class C) case as the reference group, and then fit two separate models:
  - Model 1: predicts $\boldsymbol{Y = 1}$ from $\boldsymbol{Y = 3}$
  - Model 2: predicts $\boldsymbol{Y = 2}$ from $\boldsymbol{Y = 3}$

- To predict k classes $(\boldsymbol{k > 2})$ from a fixed set of predictors$\boldsymbol{X}$;
  - **How does this approach fits**?
- We can generalize as:

$$\boldsymbol{ln}\left(\frac{\boldsymbol{P(Y = K-1)}}{\boldsymbol{P(Y = K)}}\right) = \boldsymbol{w_{0,K-1} + w_{1,K-1}X_1 + w_{2,K-1}X_2 + \cdots + w_{p,K-1}X_p}$$

- Each separate model can be fit as independent standard logistic regression models!
- **Challenges with this approach:**
  - How many parameters would need to be estimated?
  - How could these models be used to estimate the probability of an individual falling in each concentration?

3.3 Multinomial Logistic Regression: Approach-II.

- **One vs. Rest Logistic Regression (OvR):**
  - If there are 3 classes, then 3 separate logistic regressions are fit, where the probability of each category is predicted over the rest of the categories combined. So for our example, 3 models would be fit:
    - a **first model** would be fit to predict **A from (B and C)** combined.
    - a **second model** would be fit to predict **B from (C and A)** combined
    - a **third model** would be fit to predict **C from (A and B)** combined

3.4 Multinomial Logistic Regression: One Vs. Rest.

- To predict k classes $(\boldsymbol{k > 2})$ from a fixed set of predictors$\boldsymbol{X}$;
  - **How does this approach fits**?

$$\boldsymbol{ln\left(\frac{P(Y=1)}{P(Y\neq 1)}\right) = w_{0,1} + w_{1,1}X_1 + w_{2,1}X_2 + \cdots + w_{p,1}X_p}$$

$$\boldsymbol{ln\left(\frac{P(Y=2)}{P(Y\neq 2)}\right) = w_{0,2} + w_{1,2}X_1 + w_{2,2}X_2 + \cdots + w_{p,2}X_p}$$

- We can generalize as:

$$\boldsymbol{ln\left(\frac{P(Y=K)}{P(Y\neq K)}\right) = w_{0,K} + w_{1,K}X_1 + w_{2,K}X_2 + \cdots + w_{p,K}X_p}$$

- Each separate model can be fit as independent standard logistic regression models!

- **Challenges with this approach:**
  - How do we convert a set of probability estimates from separate models to one set of probability estimates?
    - **In our example; we created three different model and calculated the probability for each class such as:**
      - $\boldsymbol{P(Y \in A) = 0.55; P(Y \in B) = 0.66; P(Y \in C) = 0.44.}$
    - **In above B has the highest probability:**
      - **Does that mean "B" must be assigned?**
  - When there are more than 2 categories in the response variable then there is no guarantee that $P(Y = k) \geq 0.5$ f0r any one category. So any classifier (logistic or other) will instead have to select the group with the largest estimated probability.
  - In such cases classification boundaries are much more difficult to determine mathematically.
- **Solutions:**
  - **Is there a way to calculate the probability of all that sums up to 1**.

3.5 Multinomial Logistic Regression: Softmax

- Softmax is a mathematical function that is often used in machine learning and deep learning for various purposes, but most commonly for multiclass classification problems.
  - It is used to **transform a vector of raw scores or logits** (real numbers) into a **probability distribution** over **multiple classes**.
- The **softmax function** takes an **input vector (commonly denoted as "z")** of length "**N**" and **computes a new vector of the same length**, where **each element in the new vector represents the probability** of the **corresponding class**.
- Represented by:
  - $softmax(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}}$
  - Here:
    - **e: Euler's number.**
    - **Zi : is the raw score or "logit" for class "i".**
    - **Denominator: sum of the exponentials of all the raw scores, ensuring output probabilities to sum 1.**
      - "chatgpt"

3.6 Multinomial Logistic Regression: Softmax ~ illustrations.

- Example: Softmax illustrations:
  - Raw scores for three classes:
    - $\boldsymbol{P(Y \in A) = 0.55; P(Y \in B) = 0.66; P(Y \in C) = 0.44}$

$$\boldsymbol{softmax(z)_A = \frac{e^{0.55}}{e^{0.55} + e^{0.66} + e^{0.44}} \approx \frac{1.739}{5.230} \approx 0.332}$$

$$softmax(z)_B = \frac{e^{0.66}}{e^{0.55} + e^{0.66} + e^{0.44}} \approx \frac{1.937}{5.230} \approx 0.370$$

$$\boldsymbol{softmax(z)_C = \frac{e^{0.44}}{e^{0.55} + e^{0.66} + e^{0.44}} \approx \frac{1.554}{5.230} \approx 0.298}$$

3.6 Multinomial Logistic Regression: Softmax Regression.

- **Softmax Regression (Multiclass Classification):**
  - Given a test input $\boldsymbol{x}$, we want out hypothesis function to estimate the probability that $\boldsymbol{P(y = k|x)}$ for each value of $\boldsymbol{k}$.
  - We want to estimate the probability of the class label taking on each of the different possible values.
  - Our hypothesis will output a $\boldsymbol{k}$ dimensional vector giving us our $\boldsymbol{k}$ estimated probabilities, represented as:

$$\boxed{\widehat{f_w(x)} = \begin{bmatrix} P(y = 1|x, w) \\ P(y = 2|x, w) \\ \dots \\ P(y = k|x, w) \end{bmatrix} = \frac{\begin{bmatrix} e^{w_1^T x} \\ e^{w_2^T x} \\ \dots \\ e^{w_k^T x} \end{bmatrix}}{\sum_{j=1}^{k} e^{w^T x}}}$$

Estimating parameters of Linear Regression.

4. An Error Functions

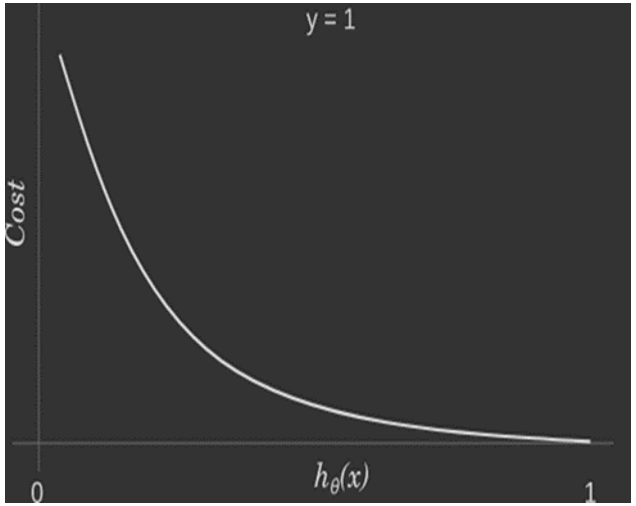
4.1 Logistic Regression: Error Function.

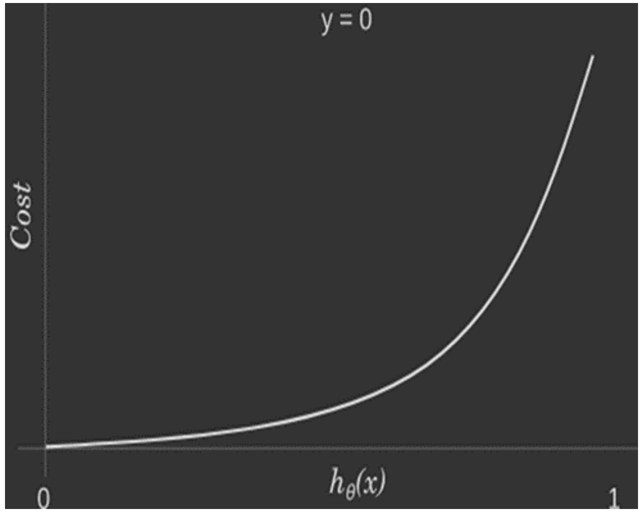
- Cost function for linear regression:
  - $MSE = \frac{1}{N} \sum (y - \hat{y})^2$
- What happens if we use the same cost function?
  - Remember we have ***sigmoid function*** in the logistic regression.
- It introduces the ***non linearity*** and we would end up with weirdly shaped non convex graph.
- We need Better cost function for LR.

4.2 Logistic Regression: Error Function.

- For logistic regression the cost function is defined as:

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





4.2 Logistic Regression: Error Function.

- It is also known as the log loss or cross-entropy loss, is a measure of the error between the predicted probabilities and true class labels.
- The logistic regression cost function is also known as the cross-entropy loss function or the log loss function.
- If we further optimize our cost functions we get;
  - $Cost(f_w(x, y) = -y\,log\big(f_w(x)\big) - (1 - y)\,log\big(1 - f_w(x)\big)$
  - Proof: try to replace y with 0 and 1 we will end up with two pieces of the original function.
- The final cost function will be:
  - $J(w) = -\frac{1}{N}[\sum_{i=1}^{n} y^i\,log\left(f_w(x^i)\right) + (1 - y^i)log(1 - f_w(x^i)))]$
  - $Here: f_w(x) = \frac{1}{1+e^{W^T X}}$

Learning a Parameters of Logistic Regression.

5. An Optimization Process.

2.6 Logistic Regression: Optimization – Gradient Descent.

- **Algorithm**:
  - Have cost function $\boldsymbol{J(\theta)}$, where $\boldsymbol{\theta}$ = $[\boldsymbol{\theta_0}, \dots, \boldsymbol{\theta_m}]$
  - Start off with some guesses for $\theta_0, \dots, \theta_m$
    - It does not really matter what values you start off with, but a common choice is to set them all initially to zero
- Repeat until convergence{

$$\theta_j = \theta_j - \alpha \sum_{i=1}^{n} \left(\frac{1}{1 + e^{-\theta^T x^i}} - \mathrm{y}^{\mathrm{i}}\right) x_j{}^{(i)}$$

Learing rate*, which controls how big a step we take when we update θ_j*

6. Classification - Evaluation Metrics.

6.1 Evaluation Metrics-Error in Classification.

- There are 2 major types of error in classification problems based on a binary outcome. They are:
  - False positives: incorrectly predicting $\boldsymbol{\hat{Y} = 1}$ when it truly is in $\boldsymbol{Y = 0}$.
  - False negative: incorrectly predicting $\boldsymbol{\hat{Y} = 0}$ when it truly is in $\boldsymbol{Y = 1}$.

6.1 Evaluation Metrics-Confusion Matrix.

- A confusion matrix, is a technique for summarizing the performance of classification algorithm.
  - Example: we have a machine learning model classifying passengers as COVID positive and negative. When performing classification predictions, there are four types of outcomes that could occur:

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

Confusion Matrix.

6.2 Confusion Matrix-Example.

- True Positives:
  - When you predict an observation belongs to a class and it actually does belong to that class.
  - In this case, a passenger who is classified as COVID positive and is actually positive.

- False Positives:
  - When you predict an observation belongs to a class and it actually does not belong to that class. In this case, a passenger who is classified as COVID positive and is actually not COVID positive (negative).

6.2 Confusion Matrix-Example.

- False Negative
  - When you predict an observation does not belong to a class and it actually does belong to that class.
  - In this case, a passenger who is classified as not COVID positive (negative) and is actually COVID positive.

- True Negatives:
  - When you predict an observation does not belong to a class and it actually does not belong to that class.
  - In this case, a passenger who is classified as not COVID positive (negative) and is actually not COVID positive (negative).

6.2 Confusion Matrix-Example.

- *Accuracy*
- *simply a ratio of correctly predicted observation to the total observations.*

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TP}$$

- *Precision:*
- *Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.*

$$\text{Precision} = \frac{\overbrace{TP}^{\text{Correctly Predicted as COVID +ve}}}{\underbrace{TP + FP}_{\text{Total Predicted as COVID +ve}}}$$

6.2 Confusion Matrix-Example.

- **Recall(Sensitivity):**
- *Aka True Positive Rate.*
- Recall is the ratio of correctly predicted positive observations to the all observations in actual class – yes
- Out of all the positive classes, how many instances were identified correctly.
- i.e. Sensitivity describes how good a model at predicting positive classes.
- higher the sensitivity value means your model is good in predicting positive classes

$$\text{Recall} = \frac{\overbrace{TP}^{\text{Correctly predicted as COVID +ve}}}{\underbrace{TP + FN}_{\text{Total COVID +ve Passengers}}}$$

- **F1 Score:**
- F1 Score is the weighted average of Precision and Recall.

$$\text{F1 Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

6.2 Confusion Matrix-Example.

- *how good a model can predict each of the classes.*
- *ROC (Receiver Operating Characteristic) curve is a visualization of false positive rate (x-axis) and the true positive rate (y-axis).*

Thank You any Question!!!