Linear Regression

**Objectives**

❖ What is machine learning

❖Types of data and terminology

❖Types of machine learning

❖ Supervised learning

❖ Linear regression

❖ Least square and Gradient Descent

❖ Hands on implementing Linear Regression from sketch.

**Machine Learning**

❖ Machine Learning is the science to make computers learn from data without explicitly program them and improve their learning over time in autonomous fashion.

❖ This learning comes by feeding them **data** in the form of observations and real-world interactions.”

❖ Machine Learning can also be defined as a tool to **predict** future events or values using past **data**.

**Types of Data**

❖***Based on Values***

❖*Continuous data (ex. Age – 0-100)*

❖ *Categorical data (ex. Gender- Male/Female)* ❖ ***Based on pattern***

❖ *Structured data (ex. Databases)*

❖ *Unstructured data (ex. Audio, Video, Text)*

**Types of Data-** continued

❖ ***Labelled data*** *– consists of input output pair. For every set input features the output/response/label is present in dataset. (ex- labelled image as cat’s or dog’s photo)* { �#, �# , �&, �& , �', �' … … … … … �), �) }

❖***Unlabelled data****- There is no output/response/label for the input features in data. (ex. news articles, tweets, audio)* {�#, �&, �' … … … … �)}

**Types of Data-** continued

❖***Training Data*** *– Sample data points which are used to train the machine learning model.*

❖***Test Data****- sample data points that are used to test the performance of machine learning model.*

Note- For modelling, the original dataset is partitioned into the ratio of 70:30 or 75:25 as training data and test data.

**Types of Machine Learning**

Machine 

Learning

Supervised Learning 

Unsupervised Learning 

Reinforcement Learning



Regression Classification

Clustering PCA

**Supervised Learning**

❖ Class of machine learning that work on externally supplied instances in form of predictor attributes and **associated target values**.

❖ The model learns from the training data using these ‘**target variables’** as reference variables.

❖Ex1 : *model to predict the resale value of a car based on its mileage, age, color etc.*

❖ The **target values** are the ‘correct answers’ for the predictor model which can either be a **regression model** or a **classification model.**

**Motivation for learning**

❖ It is being assumed that there exists a relationship/association between **input features** and **target variable.**

❖ Relationship can be observed by plotting a scatter plot between the two variables.

❖ Relationship measure can be quantified by calculating correlation between two the variables.

��� � . ���(�)=∑ �5 − �̅ ∗ �5 − �9

���� �, � =���(�, �)

∑ �5 − �̅& ∑ �5 − �9 &

**Linear Regression**

❖ Linear regression is a way to identify a relationship between two or more variables and use these relationships to predict values of one variable for given value(s) of other variable(s).

❖ Linear regression assume the relationship between variables can be modelled

through linear equation or an equation of line

Slope

Dependent/Regressed variable � = �� + ��� Independent/Regressor variable Intercept

**Multiple Regression**

❖ Last slide showed the linear regression model with one independent and one dependent variable.

❖ In Real world a data point has various important attributes and they need to be catered to while developing a regression model. (Many independent variables and one dependent variable)

� = �A + �#�# + �&�& + �'�'. … … … . wCxC

**Regression –Problem Formulation**

Let you have given with a data:

170

Age in Years **(X)**

Blood Pressure **(Y)**

160

)

56 147 49 145 72 160 38 115 63 130 47 128

Y

(

e

r

u

s

s

e

r

P

do

o

lB

150

140

130

120

110

35 45 55 65 75 Age in Year (X)

**Linear Regression**

❖ For given example the Linear Regression is modeled as:

������������� � = �A + �#���������(�)

*OR*

� = �A + �#� – Equation of line

with �A �� ��������� �� �\_���� ��� �# �� ����� �� ����

**Blood Pressure** - **Dependent Variable**

**Age in Year** - **Independent Variable**

**Linear Regression- Best Fit Line**

❖ Regression uses line to show the trend of distribution. ❖There can be many lines that try to fit the data points in scatter diagram

❖ The aim is to find **Best fit** Line

170

)

Y

(

e

r

u

s

s

e

r

P

do

o

lB

160

150

140

130

120

110

35 45 55 65 75 Age in Year (X)

**What is Best Fit Line**

❖ Best fit line tries to explain the variance in given data. (minimize the total residual/error)

**What is Best Fit Line**

❖ Best fit line tries to explain the variance in given data. (minimize the total residual/error)

**Linear Regression- Methods to Get Best**

❖**Least Square**

❖ **Gradient Descent**

****

**Linear Regression- Least Square**

**Model:** � = �A + �#�

**Task:** Estimate �ℎ� ����� �� �A ��� �#

According to pr������� �� ����� ������ �ℎ� ������ ��������� �� ����� ��� �A ��� �#

)

c

5d#

)

)

�5 = � �A + �#c 5d#

)

�5 … … … … … (1) )

c 5d#

�5�5 = �Ac 5d#

�5 + �#c5d#

�5& … … … … . (2)

**Linear Regression–Least Square** Let divide the equation (1) by *n* (number of sample points) we get:

1

)

)

�5

OR

�c 5d#

�5 = �A + �#1�c 5d#

�9 = �A + �#�̅………….(3)

So line of regression will always passes through the points (�̅, �9)

**Linear Regression–Least Square**

Now we know :

��� �, � =#) ∑5d#

) �5�5 − �̅�9 ≔#) ∑5d#

) �5�5 = ��� �, � + �̅�9 ………(4)

and

��� � =#) ∑5d#

) �5& − �̅& and ��� � =#) ∑5d#

) �5& − �9&

Dividing equation (2) by n and using equation (4) and (5) we get: ��� �, � + �̅�9 = �A�̅+ �#(��� � + �̅&)…………………….(5)

**Linear Regression–Least Square**

Now by using equation

�9 = �A + �#�̅

and

��� �, � + �̅�9 = �A�̅+ �#(��� � + �̅&)

We will get:

�# =���(�, �)

���(�)

and �A = �9 − �#�̅

**Performance metric for least square regression**

1

) (�5 − �ℎ��5)&

�& = 1 −

~~�~~ ∑5d#

1

) (�5 − �9)&

� ∑5d# 

& = 1 − (1 − �&)(� − 1)

�klm

(� − � − 1)

**Linear Regression- Gradient Descent**

**Model:** � = �A + �#�

**Task:** Estimate �ℎ� ����� �� �A ��� �# Define the cost function,

)

���� �A, �# =1�c

5d#

Objective of gradient Descent

)

(�5 − �ℎ��5)&

��,������ �A, �# =1�c

���

5d#

(�5 − (�A + �#�5))&

**Linear Regression- Gradient Descent**

**Model:** � = �A + �#�

**Task:** Estimate �ℎ� ����� �� �A ��� �# )

the objective,

)

1

w

,

0

w

(t

s

o

��,������ �A, �# =1�c

���

5d#

C

(�5 − (�A + �#�5))&

w0

**Linear Regression- Gradient Descent**

❖ Gradient descent works if following steps:

1. Initialize the parameters to some random variable

2. Calculate the gradient of cost function w. r. t. to parameters

3. Update the parameters using gradient in opposite direction.

4. Repeat step-2 and step-3 for some number of times or till it reaches to minimum cost value.

**Linear Regression- Gradient Descent** )

���� �A, �# =1�c

5d#

Calculating gradients of cost function: �����A =�����(�A, �#)

(�5 − (�A + �#�5))& )

��A=2�c5d#

(�5 − (�A + �#�5))(−1)

)

�����# =�����(�A, �#)

(�5 − (�A + �#�5))(−�)

Parameter update:

��#=2�c5d#

�A = �A − ������������ ∗ �����A �# = �# − ������������ ∗ �����#

**Performance metric for gradient based regression Root Mean Square Error** (**RMSE**) is the standard deviation of prediction errors.

���� =(�5 − �ℎ��5)&

�

**Mean absolute error (MAE)** is a measure of difference between two variables.

��� =�5 − �ℎ��5

�

Thank you !

Let see the hands on…

**Regression**

(Parametric Data Reduction: Regression and Log-Linear Models)

**Linear regression**

• Data modeled to fit a straight line

• Often uses the least-square method to fit the line **Multiple regression**

• Allows a response variable Y to be modeled as a linear function of multidimensional feature vector

**Log-linear model**

• Approximates discrete multidimensional probability distributions

Regression Analysis



Regress Analysis and Log-Linear Models

• Linear regression: *Y = w X + b*

• Two regression coefficients, *w* and *b,* specify the line and are to be estimated by using the data at hand

• Using the least squares criterion to the known values of *Y1, Y2, …, X1, X2, ….* • Multiple regression: *Y = b0 + b1 X1 + b2 X2*

• Many nonlinear functions can be transformed into the above

• Log-linear models:

• The technique is used for both hypothesis testing and model building. • Log-linear analysis is a technique used in statistics to examine the relationship between more than two categorical variables.

• Log-linear analysis is a multidimensional extension of the classical cross-tabulation chi square test .

• Approximate discrete multidimensional probability distributions

• Estimate the probability of each point (tuple) in a multi-dimensional space for a set of discretized attributes, based on a smaller subset of dimensional combinations

Linear Algebra

• Linear algebra is the branch of mathematics concerning vector spaces and linear mappings between such spaces.

⮚ Why we do study linear algebra?

• Provides a way to compactly represent and operate on sets of linear equations.

• In machine learning, we represents data as a matrices.

⮚ Consider the following system of the equations:

4X1-5X2= -13 

-2X1+3X2=9

In Matrix Notation ,

Applications of Linear Algebra

• Matrices in Engineering, such as a line of springs.

• Graphs and Networks, such as analyzing networks.

• **Markov Matrices**, Population, and Economics, such as population growth. • Linear Programming, the simplex **optimization** method. • **Fourier Series:** Linear Algebra for functions, used widely in signal processing.

• Linear Algebra for **statistics and probability**, such as least squares for regression.

• **Computer Graphics**, such as the various translation, rescaling and rotation of images.

• Application of linear algebra is that it is the type of mathematics used by **Albert Einstein** in parts of his theory of relativity.

Examples of Linear algebra in Lachine Learning • Dataset and Data Files

• Images and Photographs

• One Hot Encoding

• Linear Regression

• Regularization

• Principal Component Analysis (PCA)

• Singular-Value Decomposition (SVD)

• Latent Semantic Analysis

• Recommender Systems

**Mathematical Optimization**

• Mathematical optimization is the selection of a best element from set of available alternatives.

Represented as: min f0(x) , x is optimization variable, and f0 is objective function. f0(x) is the minimum possible values in the feasible region. ⮚Example:

• **Data Fitting:**

**Variable**: Parameter of the model.

**Constraint:** Parameter Limits, Prior information

**Objective**: Measures of fit(E.g. Minimizing of error)

**Solving Optimization Problem**

• Optimization is the very tough problems to solve.

• Optimization problems are classified in to various classes based on the properties of objectives and constraints.

• Some of these classes can be solved efficiently :

1)Linear program

II) Least Square problem

III) Convex Optimization Problem

Convex Optimization Problem



**Data Visualization**

• **“Seeing” the Data**

****

Visual Presentation

• For any kind of high dimensional data set, displaying predictive relationships is a challenge.

• The picture on the previous slide uses 3-D graphics to portray the weather balloon data numbers in text Table 11-4. We learn very little from just examining the numbers .

• Shading is used to represent relative degrees of thunderstorm activity, with the darkest regions the heaviest activity.

Market Basket Analysis

• This is the most widely used machine learning techniques. • It essentially determines what products people purchase together. • Stores can use this information to place these products in the same area. • Direct marketers can use this information to determine which new products to offer to their current customers.

• Inventory policies can be improved if reorder points reflect the demand for the complementary products.

⮚We first need a list of transactions and what was purchased. This is pretty easily obtained these days from scanning cash registers.

⮚Next, we choose a list of products to analyze, and tabulate how many times each was purchased with the others.

⮚The diagonals of the table shows how often a product is purchased in any combination, and the off-diagonals show which combinations were bought.

A Convenience Store Example

(5 transactions)

Consider the following simple example about five transactions at a convenience store:

**Transaction 1: Frozen pizza, cola, milk**

**Transaction 2: Milk, potato chips**

**Transaction 3: Cola, frozen pizza**

**Transaction 4: Milk, pretzels**

**Transaction 5: Cola, pretzels**

These need to be cross tabulated and displayed in a table.

A Convenience Store Example (5 transactions)

**Using the Results**

• The tabulations can immediately be translated into association rules and the numerical measures computed.

• Comparing this week’s table to last week’s table can immediately show the effect of this week’s promotional activities.

• Some rules are going to be *trivial* (hot dogs and buns sell together) or *inexplicable (*toilet rings sell only when a new hardware store is opened).

• **Limitations to Market Basket Analysis:**

• A large number of real transactions are needed to do an effective basket analysis, but the data’s accuracy is compromised if all the products do not occur with similar frequency.

• The analysis can sometimes capture results that were due to the success of previous marketing campaigns (and not natural tendencies of customers).

Computing Measures of Association



Data Preprocessing

• **Data quality**: accuracy, completeness, consistency, timeliness, believability, interpretability

• **Data cleaning**: e.g. missing/noisy values, outliers

• **Data integration** from multiple sources:

• Entity identification problem

• Remove redundancies

• Detect inconsistencies

• **Data reduction**

• Dimensionality reduction

• Numerosity reduction

• Data compression

• **Data transformation and data discretization**

• Normalization

• Concept hierarchy

Why Preprocess the Data?

• Measures for data quality: A multidimensional view • Accuracy: correct or wrong, accurate or not

• Completeness: not recorded, unavailable, …

• Consistency: some modified but some not, dangling, … • Timeliness: timely update?

• Believability: how trustable the data are correct? • Interpretability: how easily the data can be understood?

Data Transformation

• A function that maps the entire set of values of a given attribute to a new set of replacement values s.t. each old value can be identified with one of the new values

• **Methods**

• Smoothing: Remove noise from data

• Attribute/feature construction

• New attributes constructed from the given ones

• **Aggregation:** Summarization, data cube construction

• **Normalization:** Scaled to fall within a smaller, specified range

• min-max normalization

• z-score normalization

• normalization by decimal scaling

• **Discretization**: Concept hierarchy climbing

Normalizing data



Discretization

• Three types of attributes

• Nominal—values from an unordered set, e.g., color, profession • Ordinal—values from an ordered set, e.g., military or academic rank • Numeric—real numbers, e.g., integer or real numbers

• Discretization: Divide the range of a continuous attribute into intervals • Interval labels can then be used to replace actual data values • Reduce data size by discretization

• Supervised vs. unsupervised

• Split (top-down) vs. merge (bottom-up)

• Discretization can be performed recursively on an attribute • Prepare for further analysis, e.g., classification

Data Augmentation

• **Data augmentation** is a **data**-depended process.

• In general, you need it when your **training data** is complex and you have a few samples.

• It can help, the **features** the network will learn to extract are easy and highly different from each other.

• It means increasing the number of data points.

• In terms of **images,** it may mean that increasing the number of images in the dataset.

• It adds value to base **data** by adding information derived from internal and external sources.

• **Crops random** regions of the source image according to a given range and speed, to generate a smooth animation.

Linearity vs non Linearity

**Linearity:**

Linearity and non linearity defined on the basis of activation function.

• In linear regression, data are modeled using linear predictor functions.

• linear methods involve only linear combinations of data, leading to easier implementations, etc.

• it is a statistical analytic technique used for separating sets of observed values and allocating new values.

• It is used PCA and creating an Index for missing data.

• **Non Linearity:** ( why do we need Non-Linearities)

• Non-linear functions are those which have degree more than one and they have a curvature.

• when we plot a Non-Linear function. Now we need a Neural Network Model to learn and represent almost anything and any arbitrary complex function which maps inputs to outputs.

• Non linear techniques are ANN and SVM.

• They use, in general, non linear functions of the data (the activation function in ANN or the kernel in SVM .

**General discussion**

• **Pattern recognition**

• Patterns: images, personal records, driving habits, etc. • Represented as a vector of features (encoded as integers or real numbers in NN)

• Pattern classification:

• Classify a pattern to one of the given classes

• Form pattern classes

• Pattern associative recall

• Using a pattern to recall a related pattern

• **Pattern completion**: using a partial pattern to recall the whole pattern

• **Pattern recovery:** deals with noise, distortion, missing information





• If n = 3 (three input units), then the decision boundary is a two dimensional plane in a three dimensional space

• In general, a decision boundary is a n-1 +∑ =

***n***

***b xiwi***

0

***i***

=

1

• dimensional hyper-plane in an n dimensional space, which partition the space into two decision regions

• This simple network thus can classify a given pattern into one of the two classes, provided one of these two classes is entirely in one decision region (one side of the decision boundary) and the other class is in another region.

• The decision boundary is determined completely by the weights ***W*** and the bias ***b*** (or threshold ***q***).

**Linear Separability Problem**

• If two classes of patterns can be separated by a decision boundary,

represented by the linear equation

+∑ =

***n***

***b xiwi***

0

***i***

=

1

then they are said to be linearly separable. The simple network can correctly classify any patterns.

• Decision boundary (i.e., ***W, b*** or θ) of linearly separable classes can be determined either by some learning procedures or by solving linear equation systems based on representative patterns of each classes.

• If such a decision boundary does not exist, then the two classes are said to be linearly inseparable.

• Linearly inseparable problems cannot be solved by the simple network , more sophisticated architecture is needed.







Activation Function

• **What** are **Activation functions** and what are it uses in a Neural Network Model?

• Activation functions are really important for a Artificial Neural Network to learn and make sense of something really complicated and Non-linear complex functional mappings between the inputs and response variable.

• It is also known as Transfer Function.

• Activation functions are important for a Artificial Neural Network to learn and understand the complex patterns.

• The main function of it is to introduce non-linear properties into the network. • **What it does?**

• Their main purpose is to **convert** a input signal of a node in a A-NN to an output signal. That output signal now is used as a input in the next layer in the stack.

• The non linear activation function will help the model to understand the complexity and give accurate results.

• **why can’t we do it without activating the input signal?**

• If we do not apply a Activation function then the output signal would simply be a simple linear function.

• A linear function is just a polynomial of one degree. Now, a linear equation is easy to solve but they are limited in their complexity and have less power to learn complex functional mappings from data.

• A Neural Network without Activation function would simply be a Linear regression Model, which has limited power and does not performs good most of the times.

• Also without activation function our Neural network would not be able to learn and model other complicated kinds of data such as images, videos , audio , speech etc.

3-Types of Activation Functions

**Binary Step Function**

• A **binary step function** is a threshold-based activation function. If the input value is above or below a certain threshold, the neuron is activated and sends exactly the same signal to the next layer.

• **Linear Activation Function**

A linear activation function takes the form: **A = cx**

****

**Non-Linear Activation Functions**

• Non-linear functions are those which have degree more than one and they have a curvature. • when we plot a Non-Linear function. Now we need a Neural Network Model to learn and represent almost anything and any arbitrary complex function which maps inputs to outputs. • Non linear techniques are ANN and SVM.

• Modern neural network models use non-linear activation functions.

• They allow the model to create complex mappings between the network’s inputs and outputs, which are essential for learning and modeling complex data, such as images, video, audio, and data sets which are non-linear or have high dimensionality.

• Almost any process imaginable can be represented as a functional computation in a neural network, provided that the activation function is non-linear.

• Non-linear functions address the problems of a linear activation function: They allow backpropagation because they have a derivative function which is related to the inputs. • They allow “stacking” of multiple layers of neurons to create a deep neural network. Multiple hidden layers of neurons are needed to learn complex data sets with high levels of accuracy. • **Most popular types of Activation functions -**

• Sigmoid or Logistic

• Hyperbolic tangent

• ReLu -Rectified linear units

**Sigmoid Activation function**

• It is a activation function of form **f(x) = 1 / 1 + exp(-x) .**

• Its Range is between 0 and 1. It is a S — shaped curve.

• It is easy to understand and apply but it has major reasons which have made it fall out of popularity -

• Vanishing gradient problem

• Secondly , its output isn’t zero centered. It makes the gradient updates go too far in different directions. **0 < output < 1, and it makes optimization harder.**

• Sigmoids saturate and kill gradients. 

• Sigmoids have slow convergence.

**Sigmoid / Logistic**

**Advantages**

• **Smooth gradient**, preventing “jumps” in output values.

• **Output values bound** between 0 and 1, normalizing the output of each neuron.

• **Clear predictions**—For X above 2 or below -2, tends to bring the Y value (the prediction) to the edge of the curve, very close to 1 or 0. This enables clear predictions.

**Disadvantages**

• **Vanishing gradient**—for very high or very low values of X, there is almost no change to the prediction, causing a vanishing gradient problem. This can result in the network refusing to learn further, or being too slow to reach an accurate prediction.

• **Outputs not zero centered**.

• **Computationally expensive**

**Hyperbolic Tangent function- Tanh**

It’s mathamatical formula is **f(x) = 1 — exp(-2x) / 1 + exp(-2x).**

• Now it’s output is zero centered because its range in between -1 to 1 • ie -1 < output < 1 .

• Hence optimization is *easier* in this method hence in practice it is always preferred over Sigmoid function . But still it suffers from Vanishing gradient problem.

**ReLu- Rectified Linear units**

• It has become very popular due to 6 times improvement in convergence from Tanh function.

• It’s just R(x) = max(0,x) i.e if x < 0 , R(x) = 0 and if x >= 0 , R(x) = x. • Hence as seeing the mathematical form of this function we can see that it is very simple and efficient .

• A lot of times in Machine learning and computer science we notice that most simple and consistent techniques and methods are only preferred and are best.

• Hence it **avoids and rectifies vanishing** gradient problem . Almost all deep learning Models use ReLu nowadays.

• But its **limitation** is that it should only be used **within Hidden layers** of a Neural Network Model.

• For output layers we should use a **Softmax** function for a Classification problem to compute the probabilities for the classes.

• For a regression problem it should simply use a **linear** function.

• Another problem with ReLu is that some gradients can be breakable during training and can die.

• It can cause a weight update which will makes it never activate on any data point again. Simply saying that ReLu could result in Dead Neurons.

• To fix this problem another modification was introduced called ***Leaky ReLu*** to fix the problem of dying neurons. It introduces a small slope to keep the updates alive.

• We then have another variant made form both ReLu and Leaky ReLu called **Maxout** function **.**

****

weights and bias

Neural networks

1

Neural networks

• Neural networks are made up of many artificial neurons. • Each input into the neuron has its own weight associated with it illustrated by the red circle.

• A weight is simply a floating point number and it's these we adjust when we eventually come to train the network.

2

Neural networks

• A neuron can have any number of inputs from one to n, where n is the total number of inputs.

• The inputs may be represented therefore as *x1, x2, x3… xn.*

• And the corresponding weights for the inputs as *w1, w2, w3… wn*.

• Output *a = x1w1+x2w2+x3w3... +xnwn*

**3

How do we actually *use* an artificial neuron?

• feedforward network: The neurons in each layer feed their output forward to the next layer until we get the final output from the neural network.

• There can be any number of hidden layers within a feedforward network.

• The number of neurons can be completely arbitrary. 4

Multi-Layer Perceptron (MLP) 5

x1

xn

We will introduce the MLP and the backpropagation algorithm which is used to train it

MLP used to describe any general feedforward (no recurrent connections) network

However, we will concentrate on nets with units arranged in layers

6

x1

xn

NB different books refer to the above as either 4 layer (no. of layers of neurons) or 3 layer (no. of layers of adaptive weights). We will follow the latter convention

1st question:

what do the extra layers gain you? Start with looking at what a single layer can’t do

7

Perceptron Learning Theorem

• A perceptron (threshold unit) can *learn* anything that it can *represent* (i.e. anything separable with a hyperplane)

8

The Exclusive OR problem

A Perceptron cannot represent Exclusive OR since it is not linearly separable.



9

Minsky & Papert (1969) offered solution to XOR problem by combining perceptron unit responses using a second layer of Units. Piecewise linear classification using an MLP with threshold (perceptron) units

+1

1

3

2

+1

10