Notes in Linear Algebra and Neural Networks

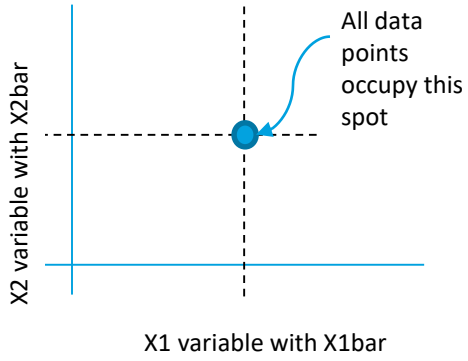
Introduction to machine learning

greatlearning

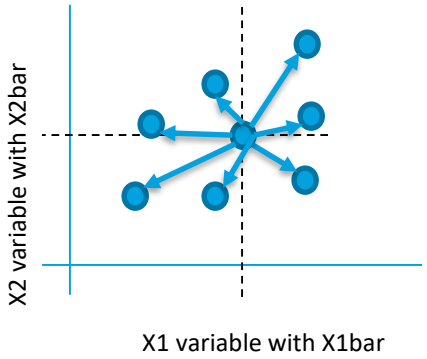
Vectors

1. Vectors are mathematical entities that are associated with magnitude and a direction.
2. Vectors, as mathematical entities, represent real world phenomenon. Usually they represent amount of force acting on an object and the direction of the force.
3. The direction of the force is with respect to a pre-determined frame of reference. For e.g. upward force, downward force etc is with reference to the person observing the phenomenon.
4. How is this related to data science?
5. In data science we talk of models which reflect the relation between independent variables represented by X and the target variable y. For e.g. mpg = -.56 wt – 1.7 cylinder et. Here mpg is the target variable and wt, cylinder are the inputs
6. The independent variables and the target variables have some distribution with central values and spread.
7. Ideally all the data points in all the independent variables should have their respective values as the corresponding central values (which are also known as the expected values). When plotted on the two variables the distribution should ideally look like

Vectors



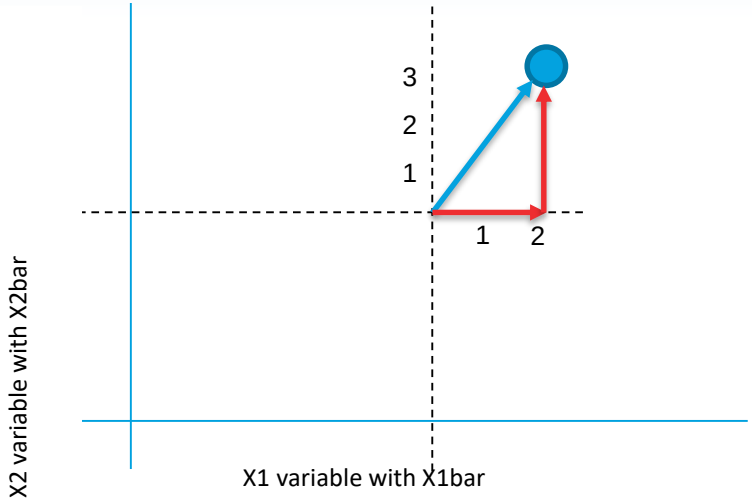
Ideal case where all the data points have the corresponding central values on respective dimensions. Hence in the mathematical space they all stack over one another at the point where the X1bar and X2bar meet!



In real world, the data points get pushed around with a net force of certain magnitude (the length of the arrow) and the direction in space (with respect to the origins).

Hence data points are also known as data vectors as they are associated with a magnitude and a direction in the mathematical space

Vectors



Looking a one single data point or data vector, we say that this vector is the result of two forces / vectors acting on the central point (shown in red)

Thus Blue Vector = Red Horizontal Vector + Red Vertical Vector

Thus vector addition helps us understand the net result of multiple change agents / forces acting on a process (that generates the data points) at any moment.

Note: vector subtraction is vector addition with negative vector i.e. vector pointing in opposite direction.

1. Vectors are represented in square bracket such as [2, 3] for blue, [2, 0] for the red horizontal vector and [0, 3] for the red vertical vector

2. The length of the resulting blue vector is also known as the magnitude of the blue vector which represents the magnitude of the resultant force at the origin. This length is also known as the Norm of the blue vector. Found using Euclidian formula

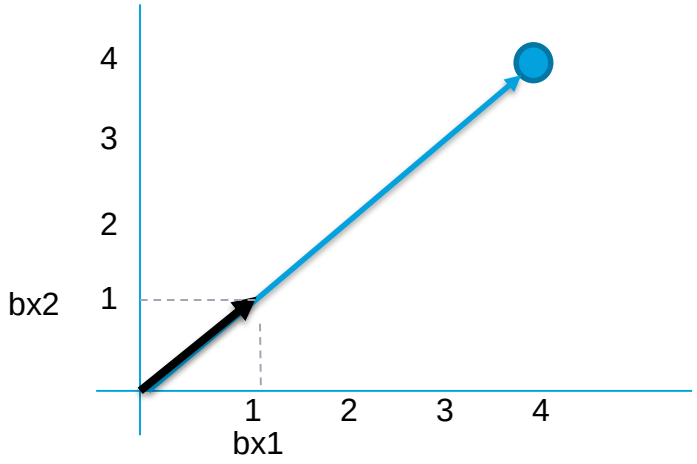
3. Unit vectors are vectors divided by their norm / length. All vectors in the space become unit length. We do this when the angle between vectors is enough for our analysis, not the length.

4. Unit vectors can also be represented along the axis. They are known as basis vectors. All vectors in the space can be derived from the basis vectors by multiplying them with appropriate magnitude quantities.

5. **Note:** the vertical red vector is shifted to the head of the red horizontal vector to see the net result of the two vectors (red vectors) acting at the origin. The net result is the blue vector

greatlearning

Vectors



Vector dot products -

1. Black vector represents unit vector of blue vector.
2. Black vector is blue vector / length of the blue vector
3. In essence, all the magnitudes are transformed to 1 when we are only interested in the direction

4. The projection of the unit vector on the axis (bx1 and bx2) are known as basis vectors

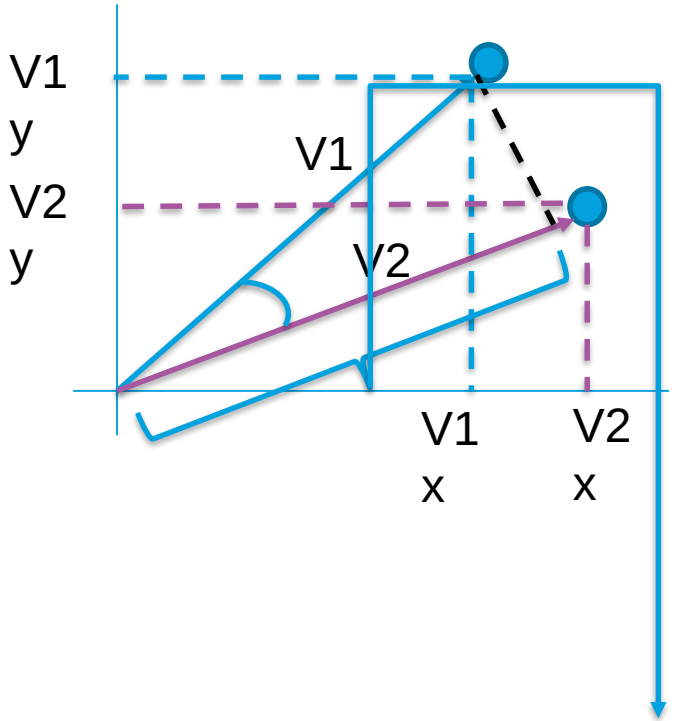
5. All vectors (data points) in the vector space can be represented as the sum of these basis vectors multiplied with appropriate values

6. For e.g. the blue vector = (bx1 * 4 + bx2 * 4)

Introduction to machine learning

greatlearning

Vectors



Vector dot products -

1. Mathematical operation on two vectors to understand how strongly they influence each other
2. The interaction is measured in terms of how much one vector is. **projected over the other** (vertical projection shown with black dashed lines).The more the projection, stronger the interaction
3. For this we multiply their individual components as shown in the grid below.
4. Whenever X component is multiplied with y component, result is zero because the two component vectors are vertical and the projection of one on the other is 0 i.e. they do not interact

$= V1 \cdot V2 \cos(\text{angle between the vectors})$

	V2X	V2y
V1X	V1X * V2X*cos()	V1X * V2y*cos()
V1Y	V1y * V2X*cos()	V1Y * V2y*cos()

$= (V1X * V2X) + 0 + 0 + (V1y*V2Y)$

Note: -
Cos(0) = 1 (between v1x,v2x & v1y, v2y)
Cost(90) = 0 (between v1x,v2y & v2x,v1y)

Vectors

Vector dot products -

1. The geometric interpretation may not always be helpful.
2. Some times the vectors and vector operations are also used to represent operations such as price per unit X quantity
3. Do not try to represent price and quantity as geometrical vectors as we did earlier. It may not make sense

Note:

We would not be using vector operations directly anywhere in our machine learning journey. However, all this is used internally by the algorithms for e.g. in gradient descent process used in linear regression

There are other operations done on vectors such as cross product which is not discussed here. You may refer to any online site if you are keen to know those.

Vectro Algebra

What a student needs to be aware of

1. What is a vector. Basic vector operations such as adding two vectors, subtracting two vectors and dot product
2. Why are data points also called data vectors and why the feature space is also known as vector space

Matrix Algebra

1. Matrix algebra is a branch of mathematics that evolved over time and proved useful In making many mathematical operations efficient.
2. Matrix is a representation of data in two dimensional form and almost like the tables in and RDBMS where rows represent entities such as people, or object and columns represent properties of the entities such as age and income
3. In data science one of the ways we use matrices is to represent all our data vectors as matrix
4. Doing so helps perform operations on the entire data set in one shot

Matrix Algebra

1. Imagine we collected accuracy scores of multiple algorithm based models on a test data and the information is as follows

ML/Run	Run1	Run2	Run3	Run4
Dtree	67	70	72	68
NaiveBayes	72	68	65	77
SVC	75	72	74	69
LogisticReg	67	76	74	79

2. The same can be represented in a matrix form as shown below as matrix M

$$\begin{bmatrix} 67 & 70 & 72 & 68 \\ 72 & 68 & 65 & 77 \\ 75 & 72 & 74 & 69 \\ 67 & 76 & 74 & 79 \end{bmatrix}$$

3. Representing data in matrix form helps operated on all the data together in one shot. For e.g. to find the correlation between the various algorithms, are their performance correlated?

Matrix Algebra

1. To find the correlation, we would need to find the avg of each column, subtract each value in the column from its average, find the standard deviation, apply the covariance formula.

$$COV(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

2. Instead, if we store the data in a matrix (M), all we need to do is M * MT (MT stands for transpose of matrix M)

3. Transpose of a matrix is flipping the rows and columns. For e.g. MT will look like this (the column headers and row names are shown only for ease of reference).

RUN/ML	Dtree	Navie Bay	SVC	Logistic
Run1	67	72	75	67
Run2	70	68	72	76
Run3	72	65	74	74
Run4	68	77	79	79

Matrix Algebra

1. **Matrix addition / subtraction**

$$\begin{bmatrix} 0 & 1 & 2 \\ 9 & 8 & 7 \end{bmatrix} + \begin{bmatrix} 6 & 5 & 4 \\ 3 & 4 & 5 \end{bmatrix} = \begin{bmatrix} 0+6 & 1+5 & 2+4 \\ 9+3 & 8+4 & 7+5 \end{bmatrix} = \begin{bmatrix} 6 & 6 & 6 \\ 12 & 12 & 12 \end{bmatrix}$$

   a. For adding / subtracting matrices both the matrices should be of similar dimensions i.e. rows and columns.

   b. Adding two matrices is like adding two vector for e.g. vector [0,9] to vector [6,3] yields [6,12] a new vector i.e. result of the first two

2. **Matrix multiplication**

   a. Matrix multiplication is like the dot product between vectors. Each matrix can be considered as a bunch of data vectors. Multiplying the matrices is finding the dot product between the vectors i.e. how they interact / influence and the magnitude of the interaction

Matrix Algebra

2. Matrix multiplication (Contd…)

b) Let two matrices A and B be

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \quad \begin{bmatrix} 7 & 8 \\ 9 & 10 \\ 11 & 12 \end{bmatrix}$$

c) Matrix multiplication is finding the dot product among system of vectors represented by A and B

d) For multiplication the number of columns in matrix A should be number of rows in B

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 7 & 8 \\ 9 & 10 \\ 11 & 12 \end{bmatrix} = \begin{bmatrix} 58 & 64 \\ 139 & 154 \end{bmatrix}$$

A_transpose **B**

3. Matrix division –

a. there is no division operation in matrices, instead we make use of inverse of a matrix

b. Inverse of a matrix M is another matrix M-inv which on multiplication with M gives 1

Matrix Algebra

1. Matrix algebra can also be used for solving equations. For e.g.

$$2x1 + 9x2 = 5$$
$$3x1 - 4x2 = 7$$

$$\begin{bmatrix} 2 & 9 \\ 3 & -4 \end{bmatrix} \times \begin{bmatrix} x1 \\ x2 \end{bmatrix} = \begin{bmatrix} 5 \\ 7 \end{bmatrix}$$

2. Instead of solving by substitution and associated algebraic steps, find X by inversing the matrix

$$\begin{bmatrix} x1 \\ x2 \end{bmatrix} = \begin{bmatrix} 5 \\ 7 \end{bmatrix} \times \begin{bmatrix} 2 & 9 \\ 3 & -4 \end{bmatrix}$$

Inverse matrix

$$= \begin{bmatrix} 2.3714 \\ 0.0286 \end{bmatrix}$$

Matrix Algebra

3. Solving a system of linear equations is finding the set of inputs at which all the concerned equations give the same output. Imagine the two equations represent two different process such as two engines of an aircraft. Though both the engines are manufactured by same manufacturer, they may not be absolutely same giving different thrust for same input amount of fuel. To ensure the thrust generated is balanced, we need to find that value of X which gives same output in both engines
4. Matrix algebra for solving the equations makes it easy and fast to find the solution

Matrix Algebra

What a student needs to be aware of -

1. A general awareness of matrices, matrix multiplication, addition and subtraction is just enough to understand the neural network, forward propagation and backward propagation

2. A deeper knowledge of the vector and matrix algebra will be useful when exploring the nuances of the various algorithms and implementing tailor made functions

Functions and function optimization

The concept of function plays a vital role in machine learning

The model built itself is a function representing the unknown real function in the population

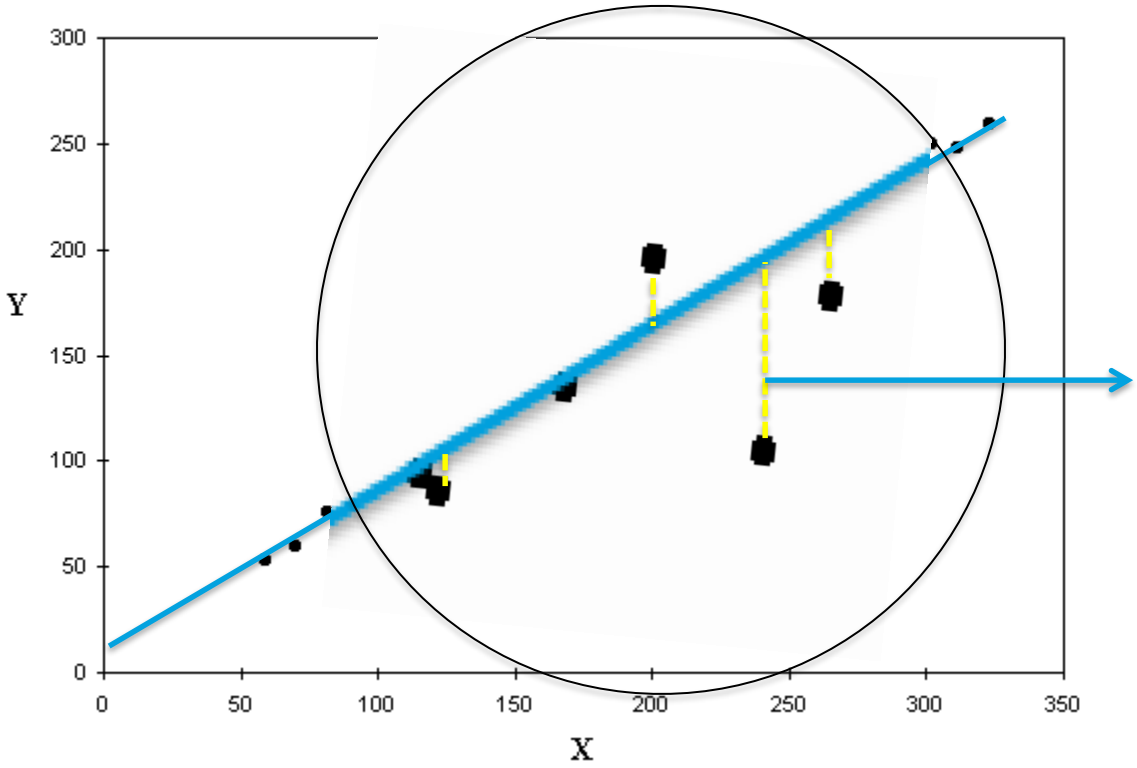
What we call model is mathematically a function representing the relation between the independent and dependent variables i.e. how the input variables impact the output / target

The difference between the real universal function and the model function is error. This error is often expressed in squared terms to eliminate the cancelling effect of the positive and negative errors.

greatlearning

Error in linear model -

a. The distance between a point and the line (drop a line vertically (shown in yellow)) is the error in prediction
b. That line which gives least sum of squared errors is considered as the best line



Error = (T – (mx + C)

Sum of all errors can cancel out and give 0

We square all the errors and sum it up. That line which gives us least sum of squared errors is the best fit

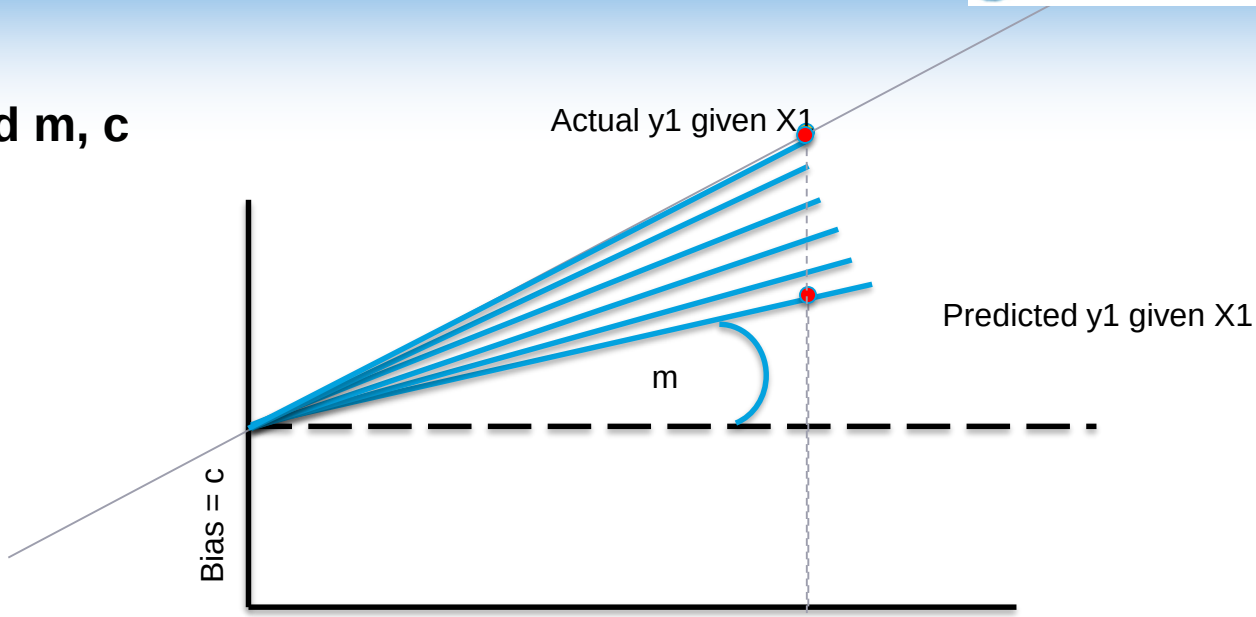
Functions and function optimization

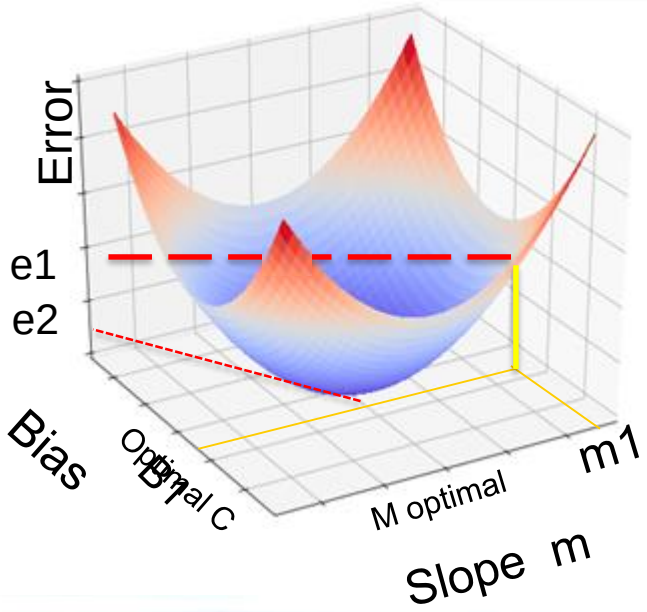
1. If we tilt the line a bit, we get a different model with different sum of squared errors.
2. Since every line is a function of two variables m and c (from the equation y = mx + c)
3. We can say that the SSE is a function of m and c. Different combination of m and c give different errors and hence different sum of squared error

4. Plotting the SSE for each combination of m, c results in a convex error surface as shown below

greatlearning

Error function and m, c



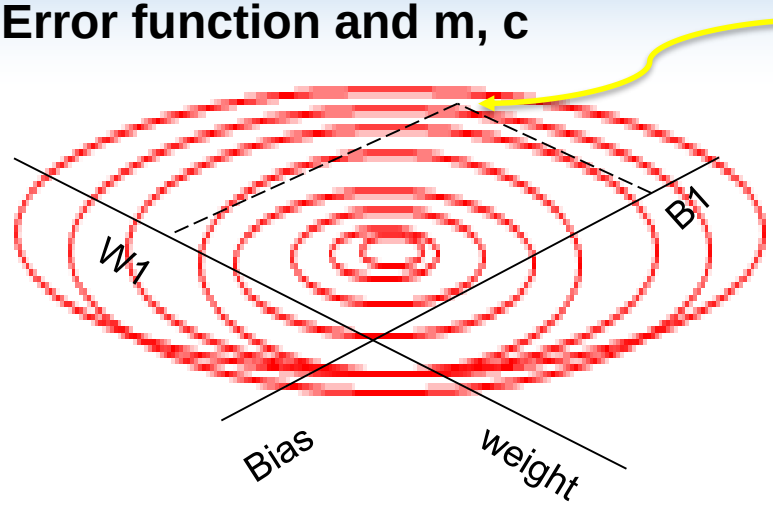


Least error E2 is at the global minima of the convex function which only one unique combination of weight (woptimal) and bias (boptimal) will fetch us.

Note: in this animation b1 = boptimal. Not necessary

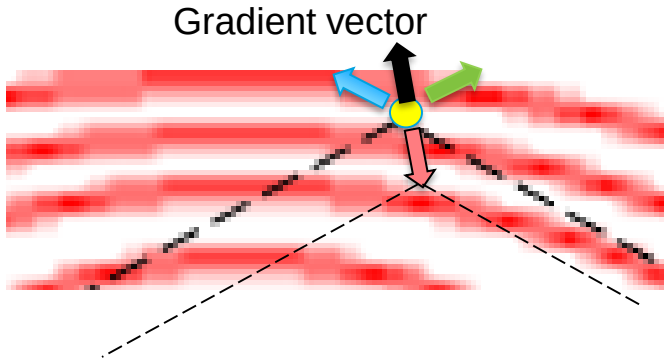
greatlearning

Error function and m, c



About the contour graph –

1. Outermost circle is highest error while innermost is the least error circle
2. A circle represents combination of paramters which result in same error. Moving on a circle will not reduce error.
3. Objective is to start from anywhere but reach the innermost circle



Gradient Descent Steps –

1. First evaluate dy(error)/d(weight) to find the direction of highest increase in error given a unit change in weight (Blue arrow)
2. Next find dy(error) /d(bias) to find the direction of highest increase in error given a unit change in bias (green arrow)
3. Add the two vectors to get the gradient (black arrow) i.e. direction of max increase in error
4. We want to decrease error, so find negative of the gradient i.e. opposite to black arrow (Orange arrow)
5. The arrow tip give new value of bias and weight.
6. Recalculate the error at this combination an iterate to step 1 till movement in any direction only increases the error

Error function and m, c

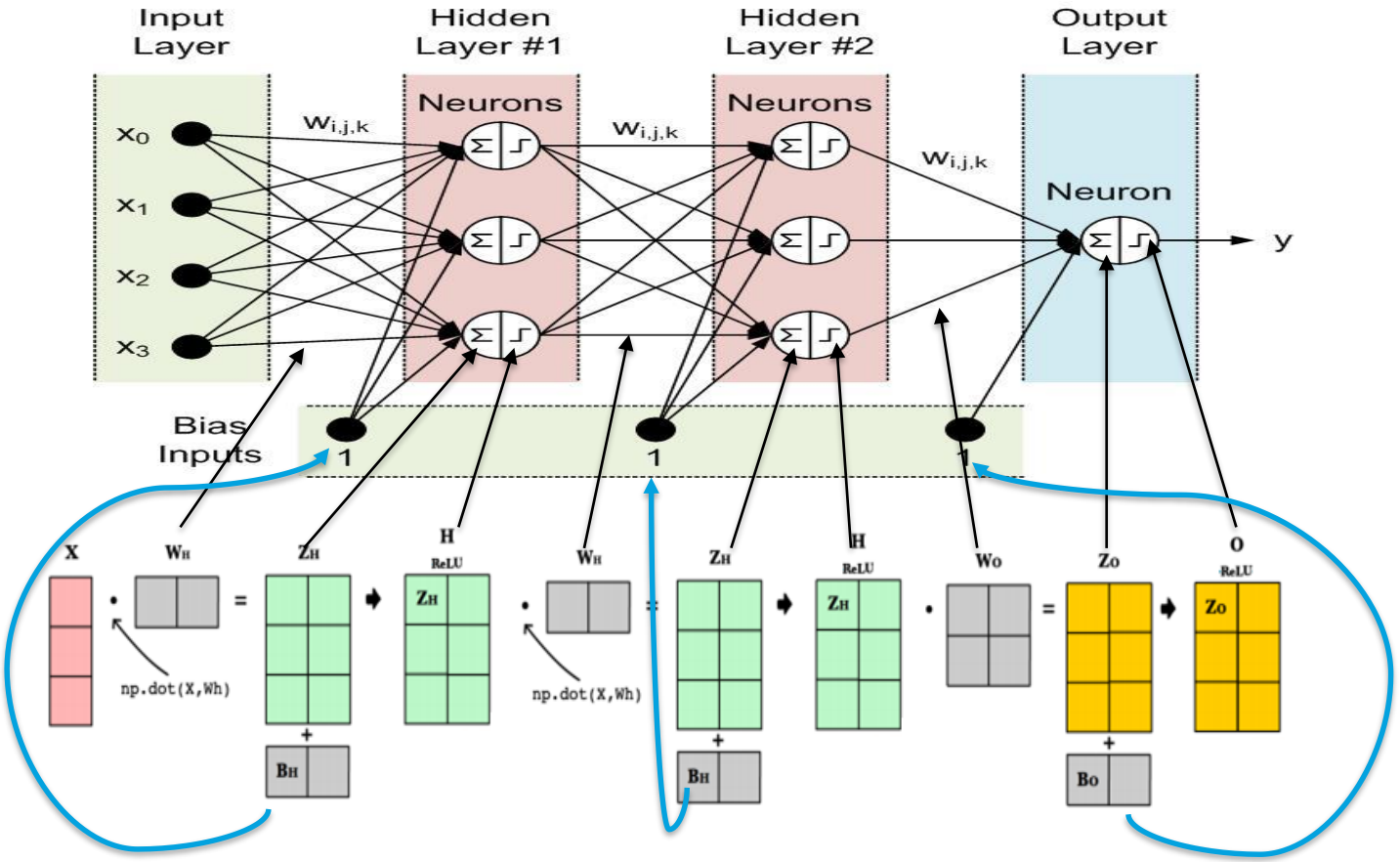
Error function and m, c

What a student needs to be aware of -

1. A conceptual understanding of the gradient descent algorithm is enough
2. Knowledge of the partial derivatives is not necessary to be able to use any algorithm including dep learning

greatlearning

Forward Propagation and Matrix operations



Note: The diagram shows step function instead of ReLU in each neuron

The bias is all set to 1. The bias supplied to a neuron depends on the weight assigned to the connector connecting bias to the neuron

Back Propagation

1. Back propagation is the process of learning that the neural network employs to re-calibrate the weights and bias at every layer and every node to minimize the error in the output layer

2. During the first pass of forward propagation, the weights and bias are random number. The random numbers are generated within a small range say 0 – 1

3. Needless to say, the output of the first iteration is almost always incorrect. The difference between actual value / class and predicted value / class is the error

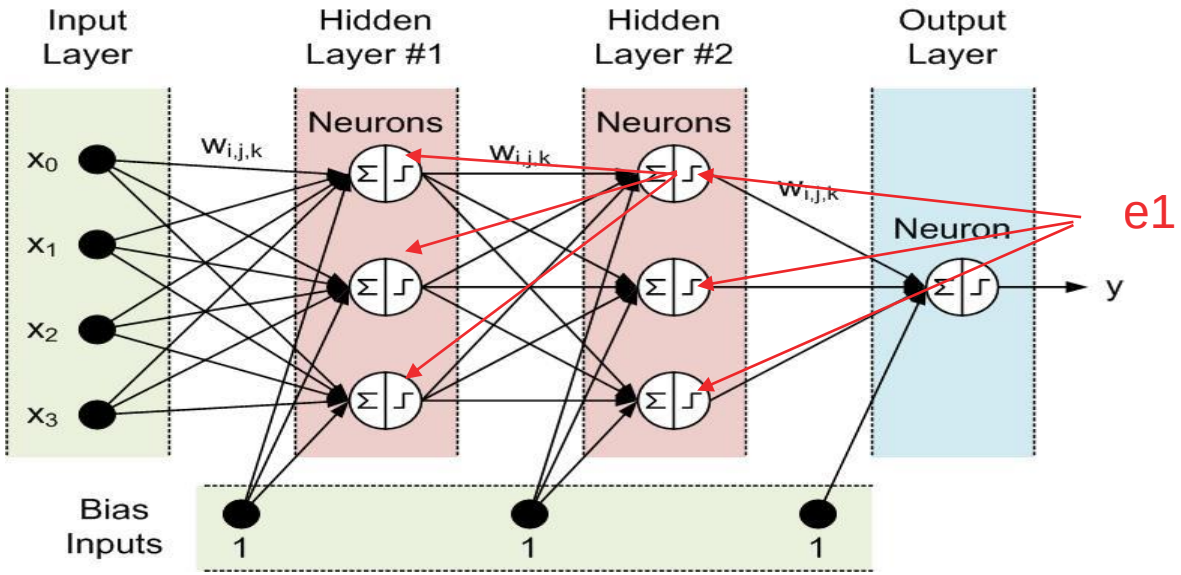
4. All the nodes in all the preceding layers have contributed to the error and hence need get their share of the error and correct their weights

5. This process of allocating proportion of the error to all the nodes in the previous layer is back propagation

6. The goal of back propagation is to adjust weights and bias in proportion to the error contribution and in iterative process identify the optimal combination of weights

greatlearning

Back Propagation



1. Error in output node shown as e1, is contributed by node 1 ,2 and 3 of layer 2 through weights w(3,1), w(3,2), w(3,3)

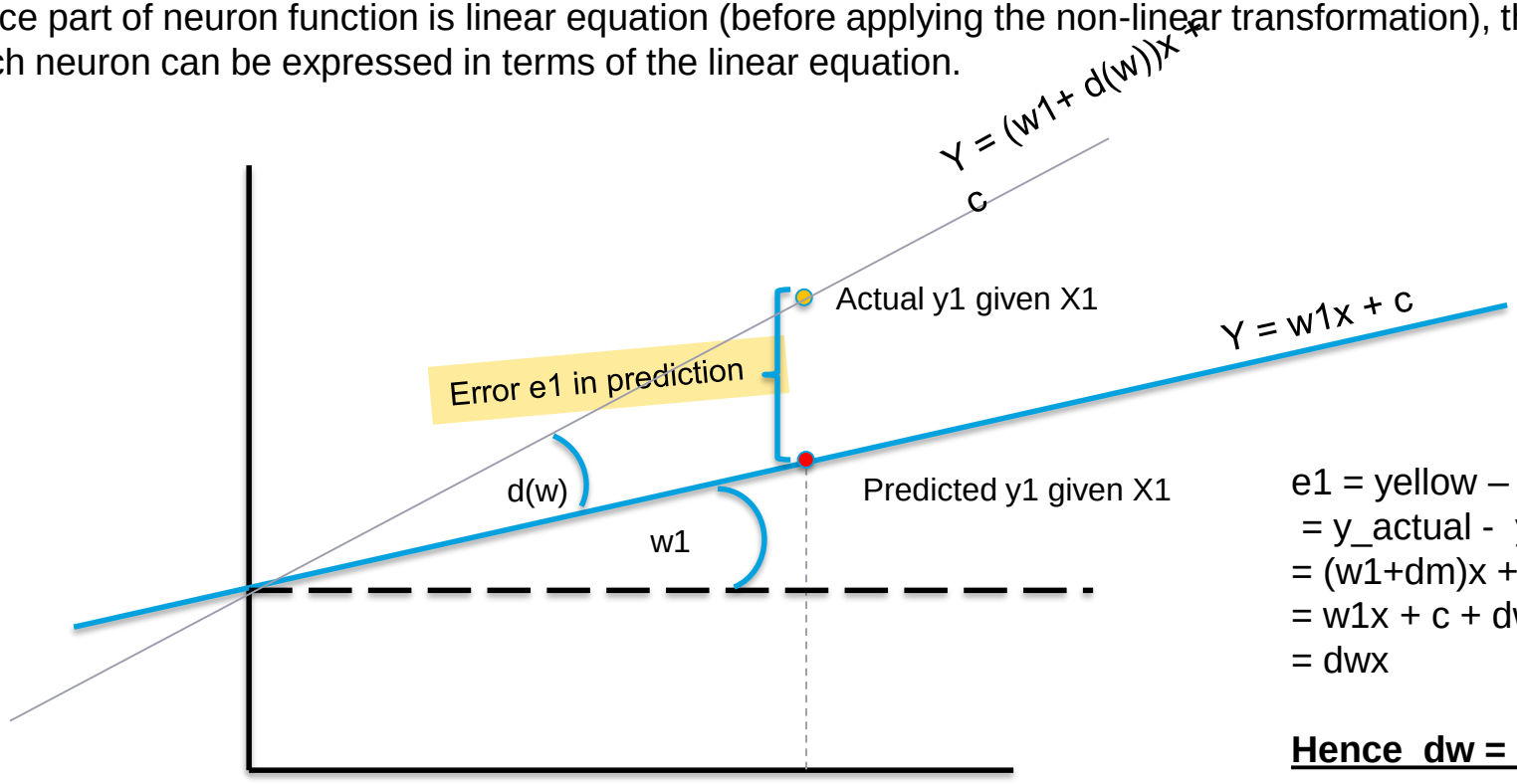
2. Proportionate error is assigned back to node 1 of hidden layer 2 is (w(3,1) / w(3,1) + w(3,2) + w(3,3)) * e1

3. The error assigned to node 1 of hidden layer 2 is proportionately sent back to hidden layer 1 neurons

4. All the nodes in all the layers re-adjust the input weights and bias to address the assigned error (for this they use gradient descent)
5. The input layer is not neurons, they are like input parameters to a function and hence have no errors to adjust

Relation between error and weights

Since part of neuron function is linear equation (before applying the non-linear transformation), the error at each neuron can be expressed in terms of the linear equation.



e1 = yellow – red
= y_actual - y_pred
= (w1+dm)x + c - (w1x + c)
= w1x + c + dwx – w1x -c
= dwx

Hence dw = e1 / x

The change required in m (dw) is e1/x.

Summary

1. Vectors and vector algebra (Addition, Subtraction, Multiplication,). We did not cover cross products
2. Matrix Algebra (Addition, Subtraction, Multiplication, finding inverse). We did not cover deteminants of matrices, inverse of matrices etc
3. Error function and it's relation with the model parameters of slop and bias
4. Gradient descent based on partial derivatives. Did not cover the types of gradient descent algorithms
5. Neural network forward and backward propagation
6. Relation between error and amount of change in m

Note: Conceptual understanding is more important than mugging up the formulas.

End