

title - Linear regression by using deep
Neural Network.

problem statement - Implement Boston housing price prediction by using linear regression, deep neural network.

objective - 1. pre-process the dataset.
2. identify outliers.

3. check the correlation.

4. Implement linear regression using deep neural network.

5. predict the price of house.

6. Evaluate the models and compare their scores like R^2 , RMSE.

Theory -

Linear Regression :- It is a supervised learning technique that involves learning the relationship between the features and the target.

the target.

\downarrow intercept independent variable
 \downarrow

$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \leftarrow$ random error term.

\uparrow dependent variable slope coefficient

Linear component.

How does linear Regression learn the parameters?

$$\beta_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$$

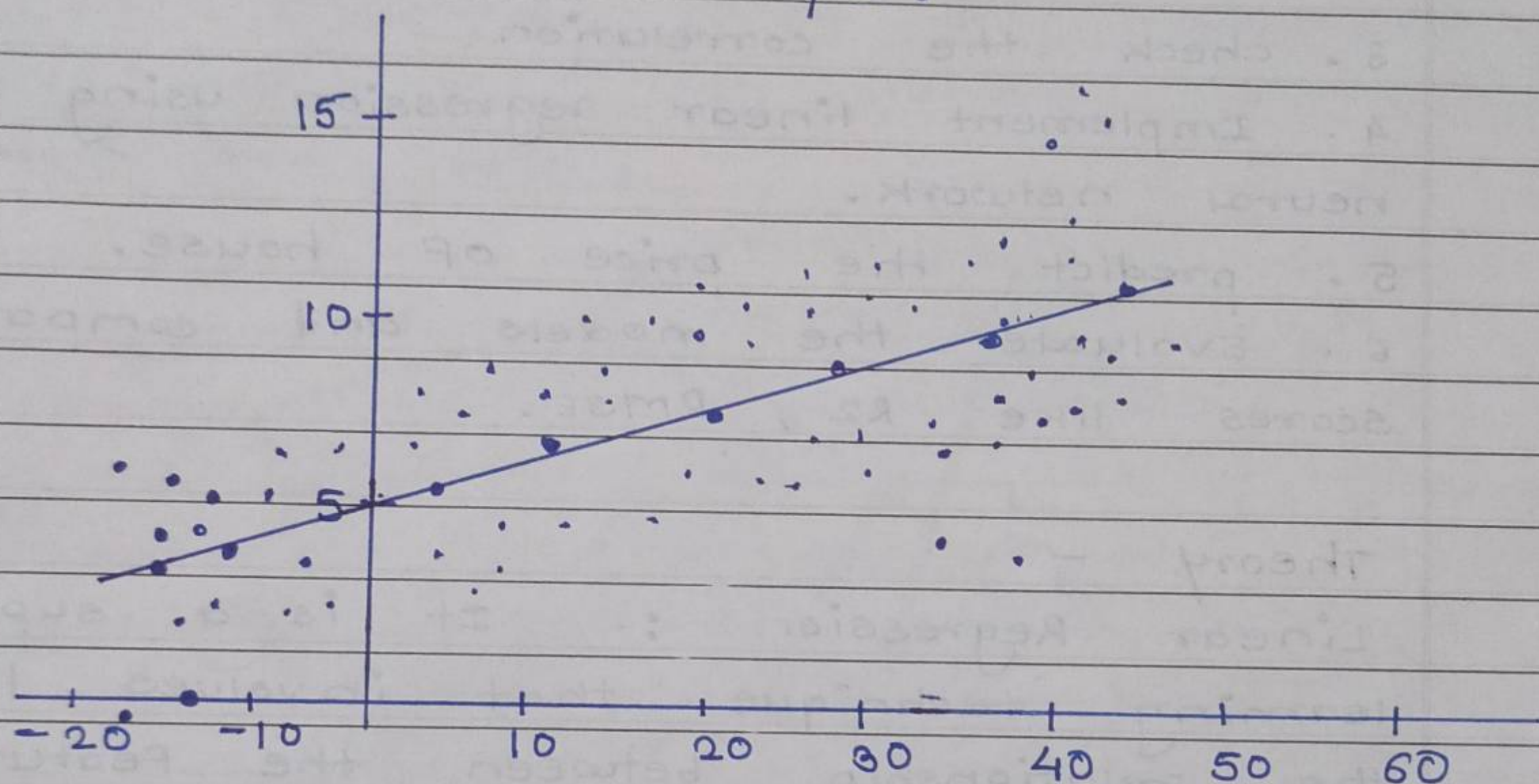
x_i = The Feature x

y_i = The target y

\bar{x} = Mean of x

\bar{y} = mean of y

The numerator denotes the covariance of the data and denominator the variance of the feature x . beta 1 parameter determines the slope.



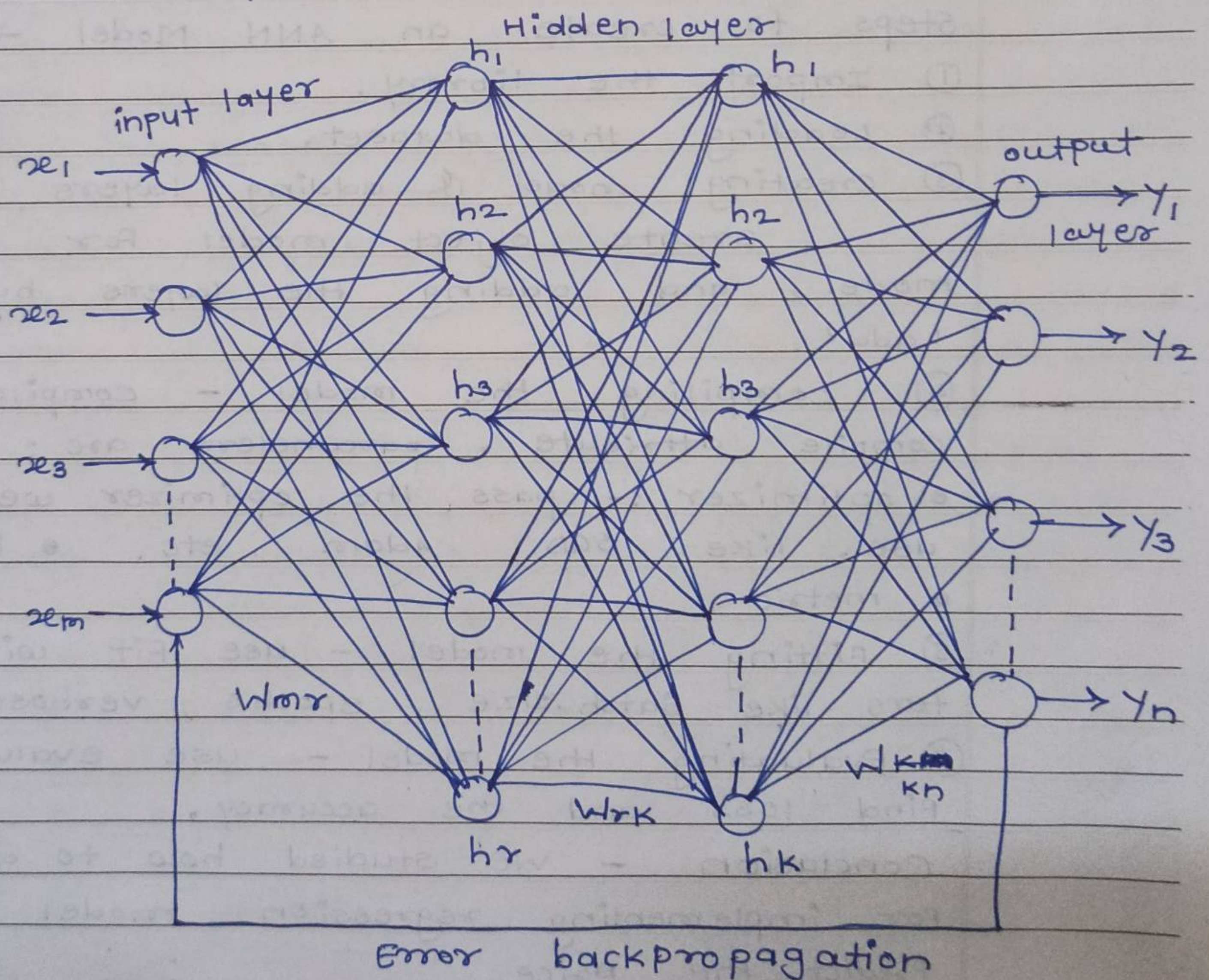
intercept - value is 5.

Regression using Artificial Neural Networks
The purpose of using artificial neural networks for regression over linear regression is that the linear regression can only learn the linear relationship between the features and target and therefore cannot learn the complex non-linear relationship. So, for

this use artificial neural network.

Artificial Neural Networks are one of the deep learning algorithms that simulate the workings of neurons in the human brain. There are many types of ANN, vanilla neural networks, Recurrent neural networks, and convolutional neural network.

The vanilla neural network can handle structured data only. whereas RNN and CNN can handle unstructured data very well.



The ANN consists of input layer, hidden layer & output layer. Each layer will be having activation function associated with each neurons. it is responsible for introducing non-linearity in the relationship. Regularizers are responsible for preventing overfitting.

Two phases -

- Forward propagation - process of multiplying weights with each feature and adding them.
- Backward propagation - updating weights.

Steps to create an ANN Model →

① Import the library.

② Loading the dataset.

③ creating model & adding layers. -

create object model for sequential model, and adding the layers by using 'add'.

④ compiling the model - compile using compile attribute. parameters are :

- optimizer :- pass the optimizer we want to use. like SGD, Adam, etc.
- loss function
- metric.

⑤ fitting the model - use fit with parameters like batch-size, epochs, verbose.

⑥ Evaluating the model - use evaluate & find loss and the accuracy.

Conclusion - We studied how to use ANN for implementing regression model and predict the price.

Title - convolutional Neural Network (CNN)

problem statement - Use MNIST Fashion Dataset and create a classifier to classify fashion clothing into categories.

Objective - To understand and implement :-
1. CNN 2. classify create for Fashion clothing.

Theory -

CNN :

A convolutional neural network is used for image and video recognition tasks. It is made up of multiple layers of filters that are applied to an input image to extract features and patterns. These filters, called convolutional layers, help to reduce the dimensions of the images.

The output of the CNN layer is then fed into fully connected layers, which perform the classification task.

CNN Architecture :

It has three layers :-

1. convolutional layer
2. pooling layer
3. fully connected layer,

- convolutional layer -



Date:

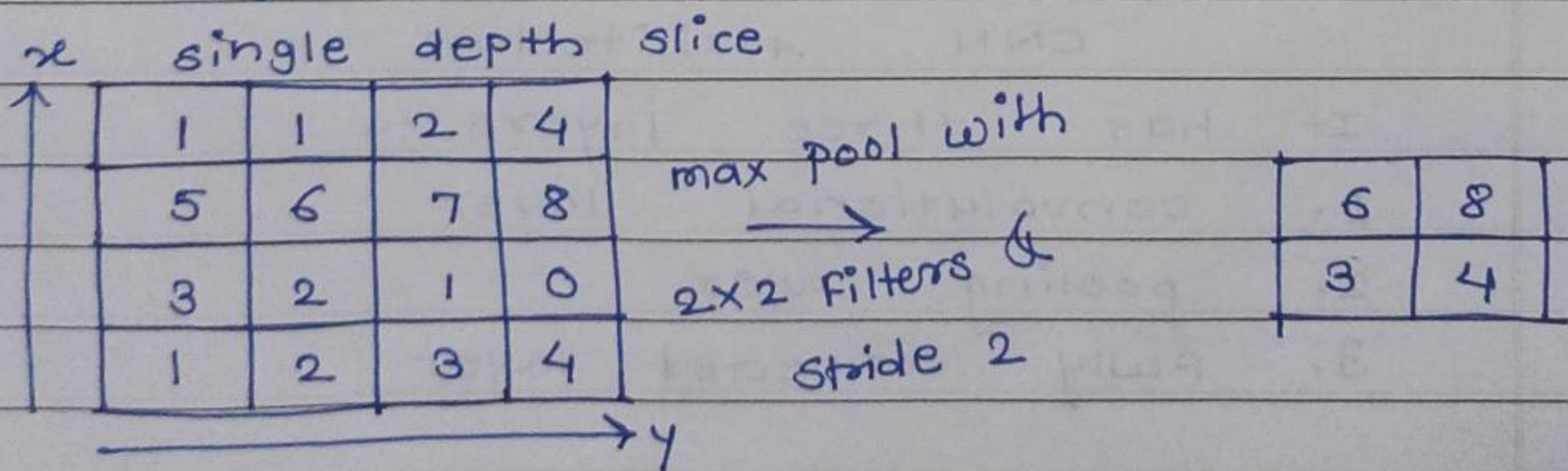
Page No.:

It is the most important part of CNN and carries most of the computational load. It uses a kernel matrix to perform a dot product with a restricted portion of the input image called the receptive field.

The kernel is smaller than image but has same depth as image's channels. During forward pass, the kernel slides over the image & produces an activation map that shows the kernel response. The stride is size of kernel's sliding.

$$W_{out} = \frac{W - F + 2P}{S} + 1$$

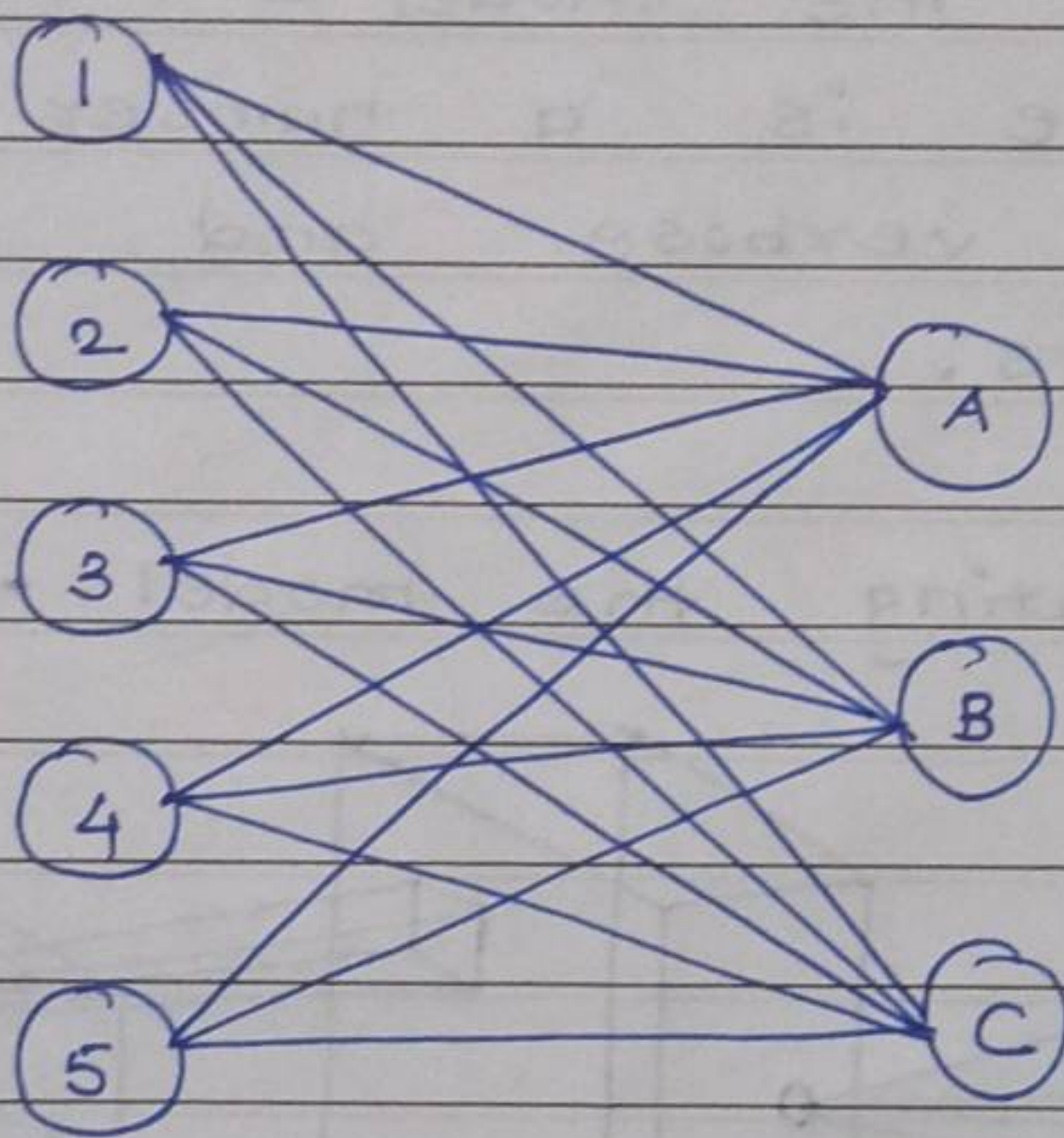
- pooling layer - It is used to reduce the spatial size of the input representation by summarising nearby outputs with a pooling function, such as max pooling. This decreases the computation and weights required in the network.



Formula →

$$W_{out} = \frac{W - F}{S} + 1$$

- **fully connected layer** - This layer performs the classification task. It takes the output of the preceding convolutional and pooling layers and flattens it into a one-dimensional vector. This vector is then fed into a traditional neural network with fully connected layers, which learns to classify the input into one or more classes.



steps to create CNN Model -

- ① import the library.
- ② Loading the dataset - keras. datasets module provides a collection of benchmark datasets for machine learning and import fashion-mnist.



Date:	
Page No.:	

③ creating model and adding layers -
fashion_model = sequential()
and compile it using compile() method.

④ compiling the model - loss : categorical
cross-entropy loss is used for multi-
class classification problems.
optimize : Adam optimizer due to its
adaptive learning rate and momentum.
metrics : evaluate performance of model.

⑤ fitting the model - model.fit()
batch-size is a number of samples ,
epochs , verbose and validation-data
parameters.

⑥ Evaluating the model -

