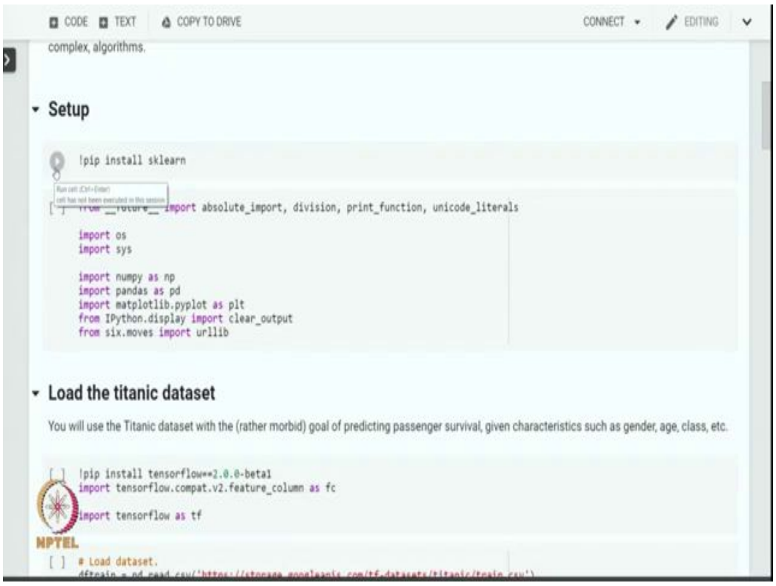
Practical Machine Learning

Dr. Ashish Tendulkar
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture – 28
Logistic Regression

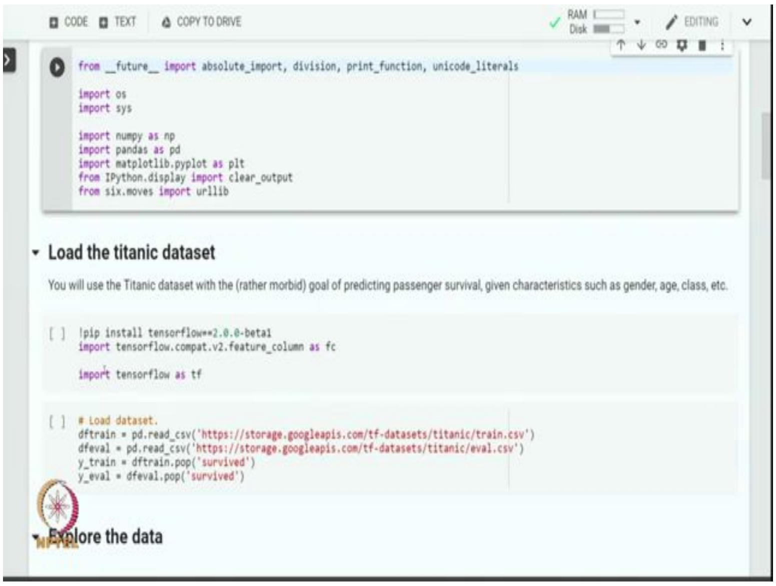
In this session, we will build a Logistic Regression model with tf.estimator API. Logistic regression is often used as a baseline for classification tasks.

(Refer Slide Time: 00:32)



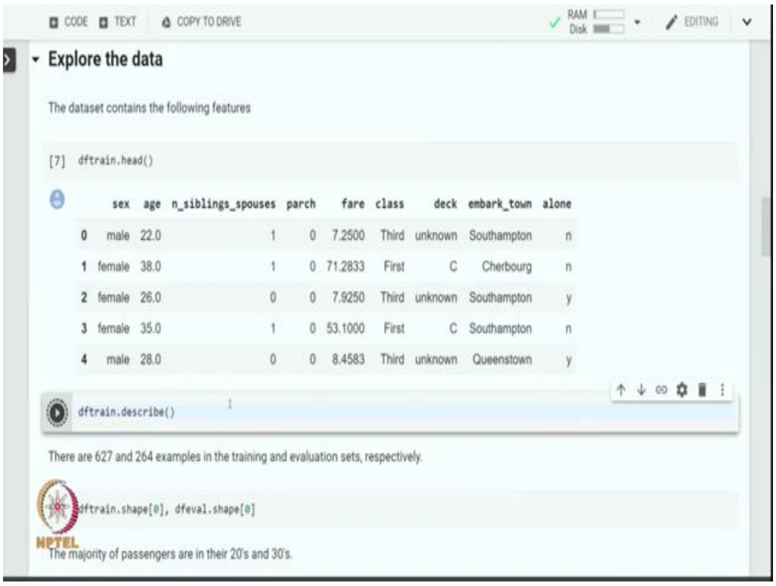
In this exercise, we will use a titanic dataset with a goal of predicting passenger survival given characteristics such as gender, age and class. Let us first install the required libraries like sklearn and tensorflow. We use matplotlib for plotting and numpy and pandas for data manipulation.

(Refer Slide Time: 01:03)



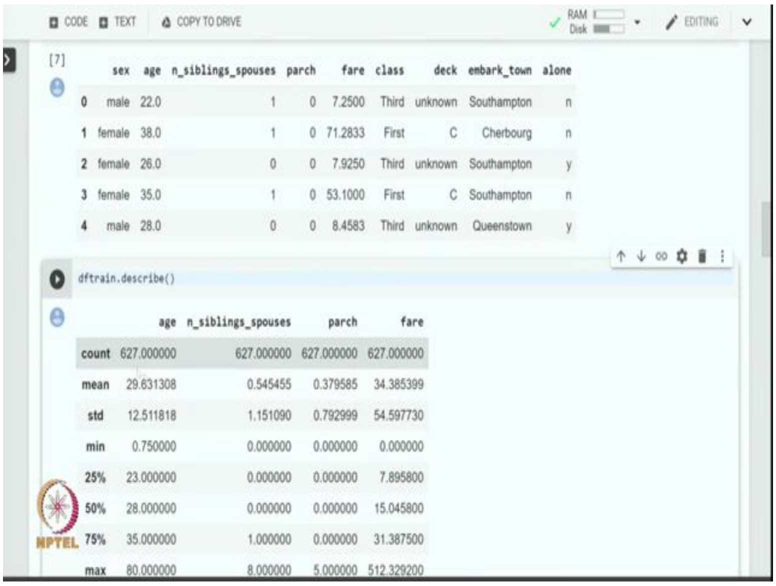
Let us load the titanic dataset, we will install tensorflow 2.0.

(Refer Slide Time: 01:10)



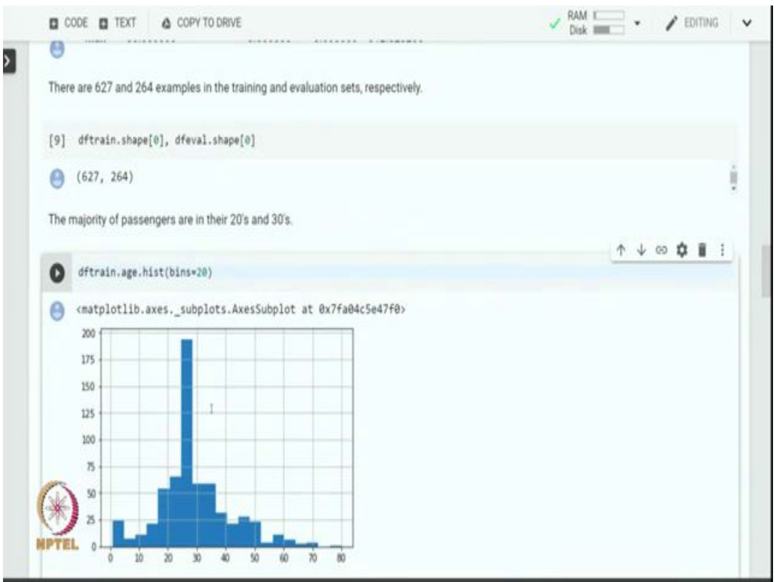
Let us load the dataset. So, we have training data in dftrain dataframe, the evaluation data in dfeval, and the respective labels are in y_train and y_eval. Let us first explore the data, we examine a few rows in the dataset. So, you can see that there are there are features like the sex of the passenger, the age, the number of siblings and spouses, the fare, the class etcetera.

(Refer Slide Time: 01:59)



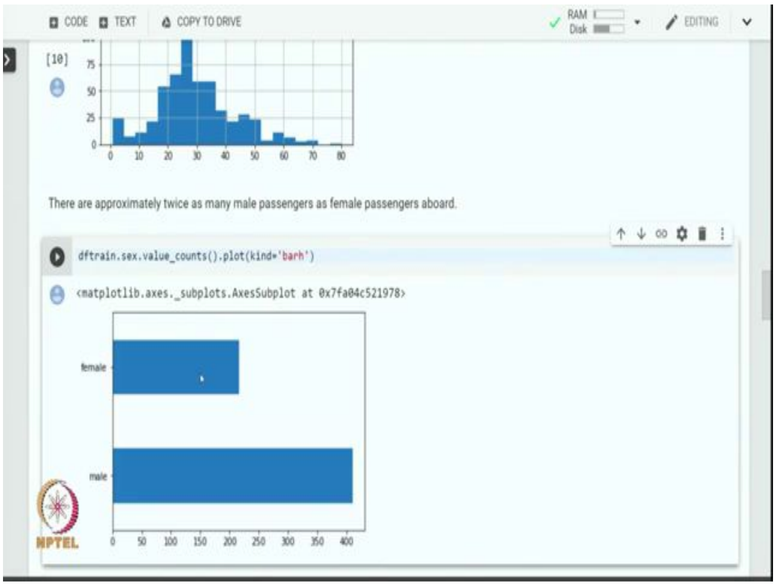
Let us also is a describe command on the data frame to get statistics about each of the numeric columns. So, we will get statistics like count, mean, standard deviation and various quartiles. You can see that there are 627 examples in training and 264 examples in evaluation set.

(Refer Slide Time: 02:22)



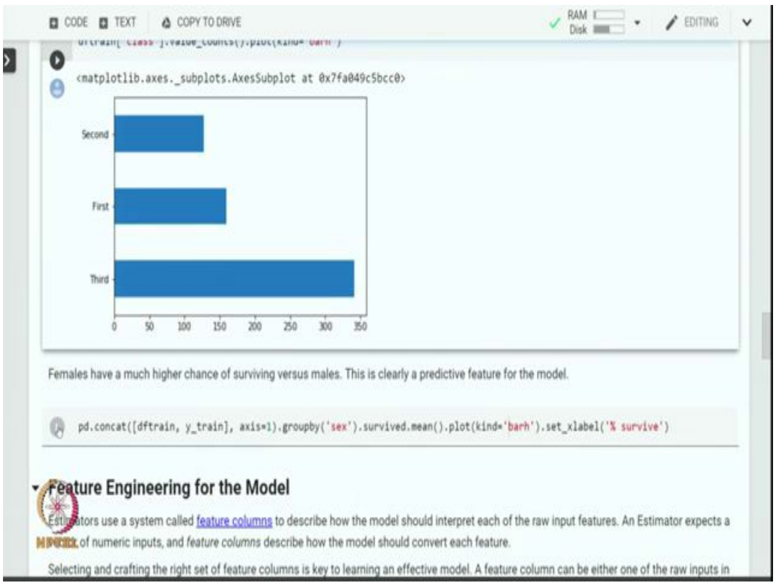
Let us plot histogram of the age and we can see that the majority of the passengers are in their 20s and 30s.

(Refer Slide Time: 02:35)



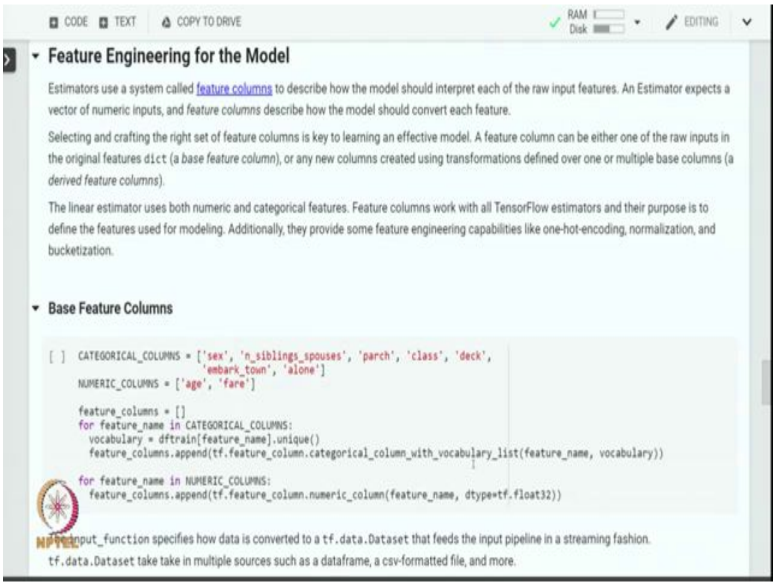
There are approximately twice as many male passengers as female passengers on board.

(Refer Slide Time: 02:41)



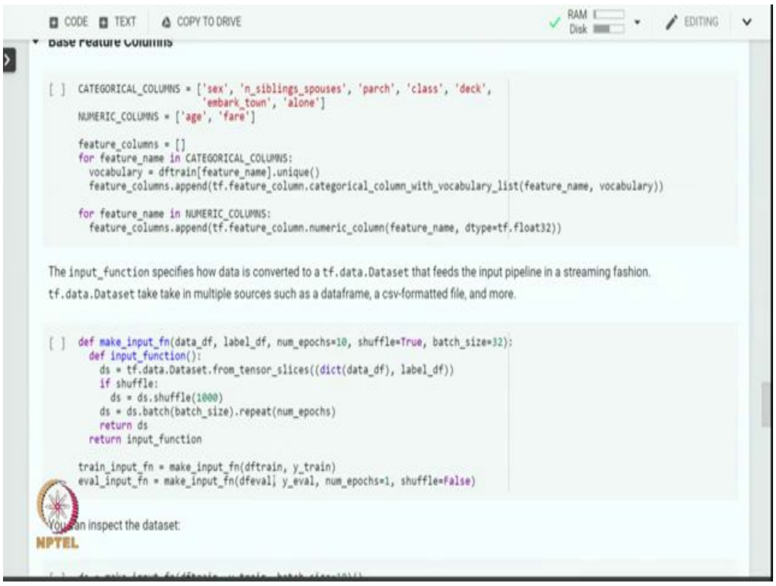
You can also see that a majority of the passengers were in the third class. Let us look at the survival by sex of the passenger. And you can see that females have higher chances of survival than their male counterparts. So, you can see that sex can be a very good predictive feature for the model. After exploring data a bit, let us get into engineering features for the model. Feature engineering is extremely important to get good results.

(Refer Slide Time: 03:22)



So, we will be using feature columns for converting the raw features into a form that can be consumed by estimator API as well as for constructing cross features from the original features. So, we have several categorical features and a few numeric features. We use numeric feature columns for numeric features. For categorical features, we first find out unique values from each of the categorical features and use categorical column with vocabulary list for converting the categorical features into one hot encoding.

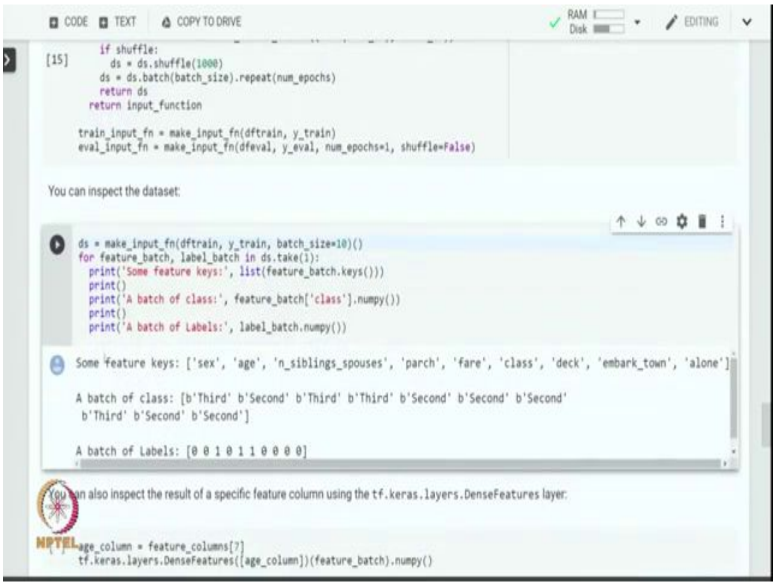
(Refer Slide Time: 04:03)



Then we define input function to specify how data is converted into tf.data.Dataset format that feeds the input pipeline in a streaming fashion. The dataset takes multiple sources such as dataframe, csv, formatted files and many more. Let us define a dataset from tensor slices and in if required shuffle the dataset and return the dataset in the batches.

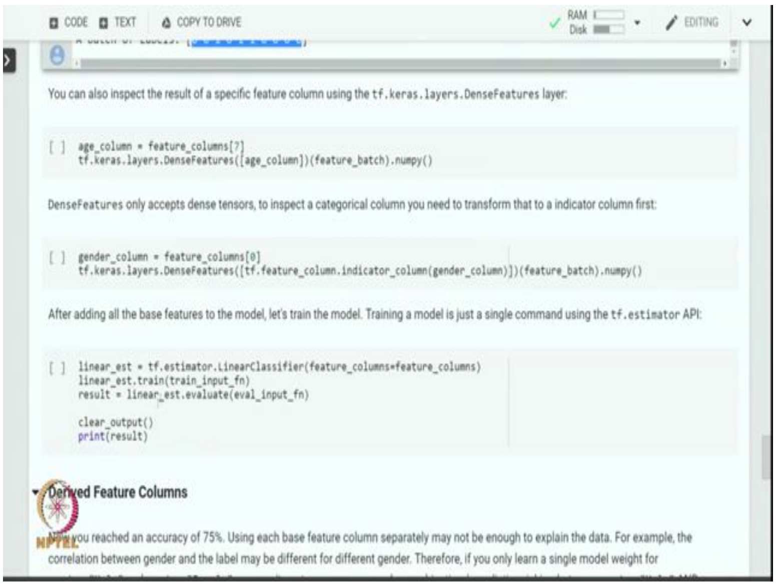
We repeat the batching operation for the number of specified epochs. This is how we define our input function and, we make the input function for training as well as for evaluation. In training we set shuffle to be true, and in evaluation we set shuffle to be false and we specified number of epochs to be one in case of evaluation. Let us inspect the dataset.

(Refer Slide Time: 05:09)



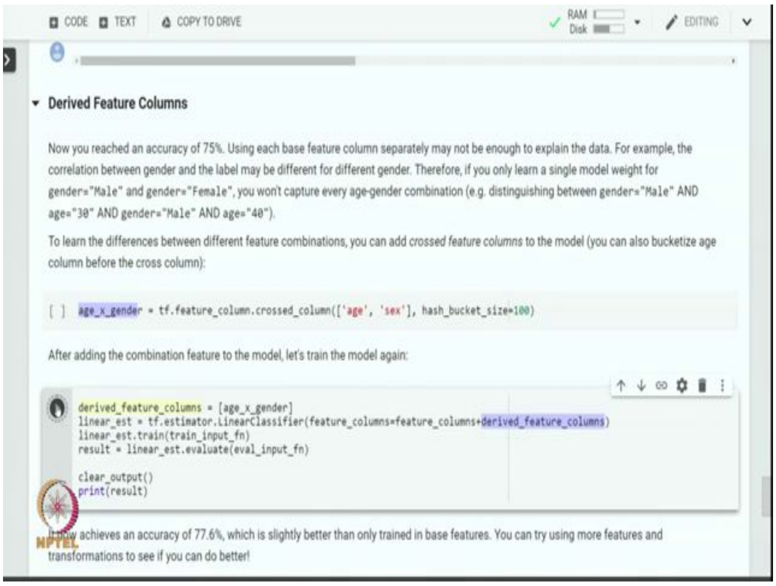
You can see that the dataset has feature keys which are the same as the original dataset that we explored and we can also see the first batch for the attribute class and you can see values like third, second, third and so on. There are 10 values in a batch because we specified a batch size of 10. And in the same manner, we can see that for a label batch also there are 10 values that are present.

(Refer Slide Time: 05:50)



Let us define our logistic regression classifier using tf.estimator.LinearClassifier command. It takes feature columns as its argument, then we train the model by giving train_input_fn as an argument and we evaluate the model using eval_input_function as an argument. We get we store the result in the result variable and we will print it at the end of the execution.

(Refer Slide Time: 06:28)



You can see that we got accuracy of 76 percent on the evaluation data. The baseline accuracy was 62 percent and there are a bunch of other metrics that are also printed. Now, we have reached accuracy of 75 percent. Now, what we will do is we will try to include even more features and see whether we can get a better accuracy. So, what we will do is we will combine different features.

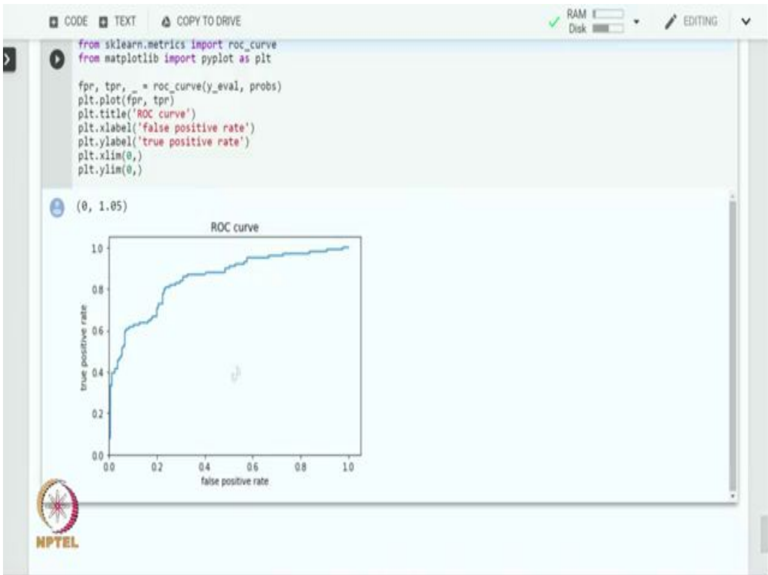
One of the feature combination that we will try is between age and gender. So, we construct a crossed column between age and gender and we set the hash bucket size to be 100. I would like to remind you about crossed column. So, whenever we cross whenever we construct a crossed column, it uses hashing for storing the values in order to avoid the problem of sparsity. Here we specify the hash bucket size to be 100.

Let us add the combined feature in the model along with the earlier feature columns. We can construct a derived feature column with all the crossed features, but for now, there is only a single crossed feature. And, we specify feature columns to be original feature columns and derived feature columns and that is how we instantiate a linear classifier and then we train it and evaluate it as earlier.

(Refer Slide Time: 08:24)

We have achieved an accuracy of 76 percent which was which is 1 percentage point higher than the earlier model.

(Refer Slide Time: 08:38)



Finally let us look at the ROC curve to get an idea about the tradeoff between true positive rate and false positive rate. So, this is the ROC curve that we got. It is a reasonable ROC curve for a classification task. So, in this session we studied how to build logistic regression classifier with tf.estimator API. We use titanic dataset as an example dataset to predict the probability of survival of a passenger using a logistic regression model.