16164571-Linear Regression

Youtube Views

I first went about this project by researching linear regression and by gaining an understanding of what kind of data it is usually performed on. The data used must be on a continuous measurement scale.

The goal of linear regression is to help predict values of a variable based upon the values of other independent variables.

I choose a dataset based off youtube which is made up of 3 independent input variables. These variables are likes, dislikes and subscribers. This data set also included a dependant variable , views. Using a linear regression model my goal was to predict the views based off these 3 independent variables.
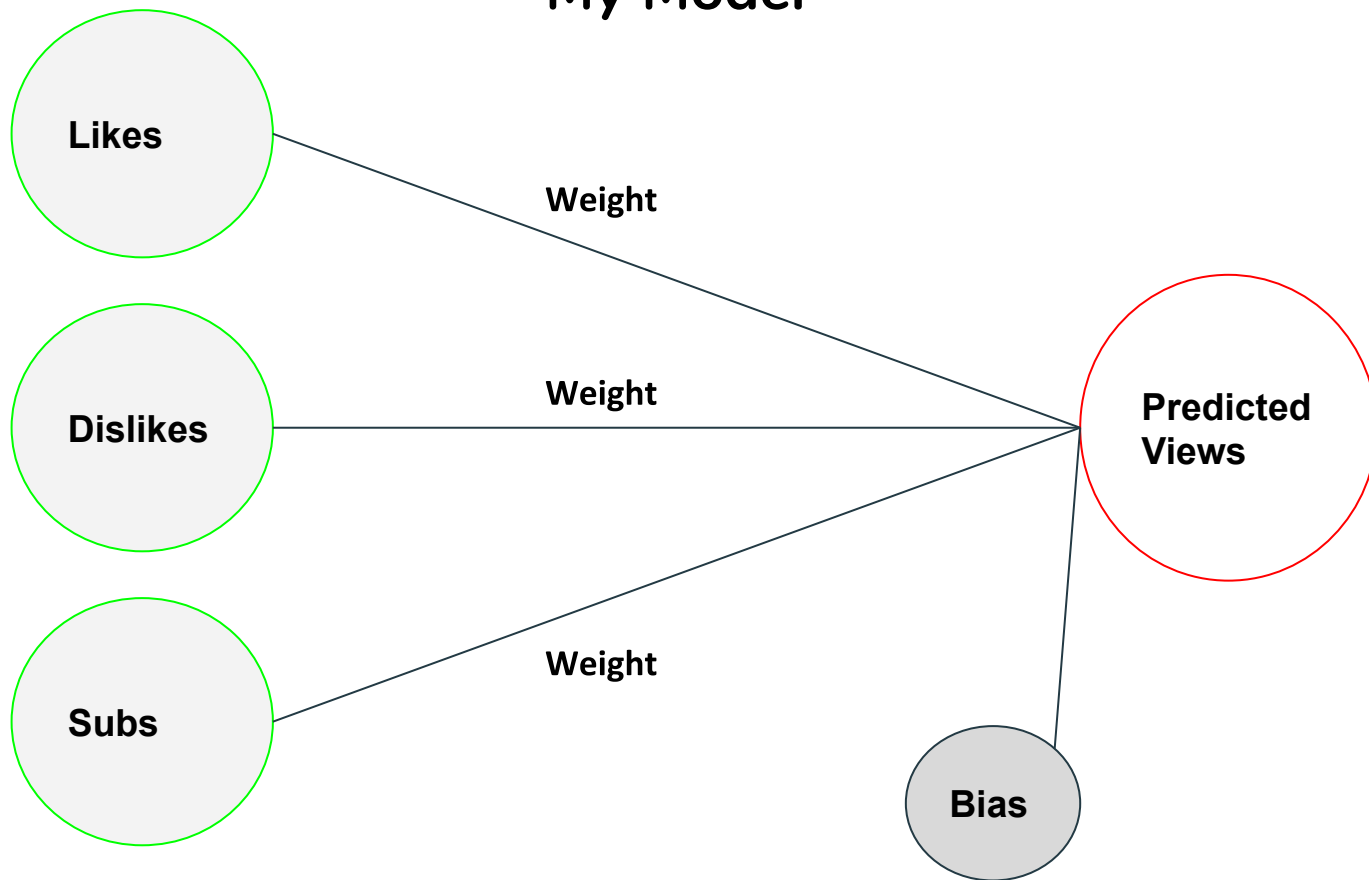
	Likes	Dislikes	Subscribers
0	104460	3365	7300000
1	103203	3570	3300000
2	22485	612	1800000
3	23077	1337	465000
4	12315	522	11000000
...	...	...	...
3359	4833	47	465000
3360	118	16	19000
3361	29	2	6100
3362	321	34	3400
3363	143	7	116000

[3364 rows x 3 columns]

	Views
0	1988888
1	2617005
2	932230
3	1165800
4	551165
...	...
3359	193093
3360	21784
3361	5427
3362	150226
3363	29077

[3364 rows x 1 columns]

In the previous slide you can see my model. It is a pretty basic model but is designed in mind with this dataset. My network has no hidden layers and contains just input and output layers.

```
#Defines linear regression model and its structure
model = Sequential()
model.add(Dense(1, input_shape=(3,)))
```
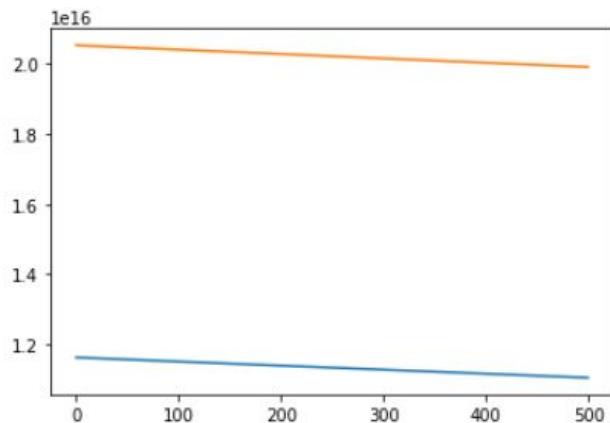
I also added a second model to the project which contained a ReLu activation to see if this would improve the accuracy of my model but it didn't change much

```
model = Sequential()
model.add(Dense(1, input_shape=(3,), activation = 'relu'))
```
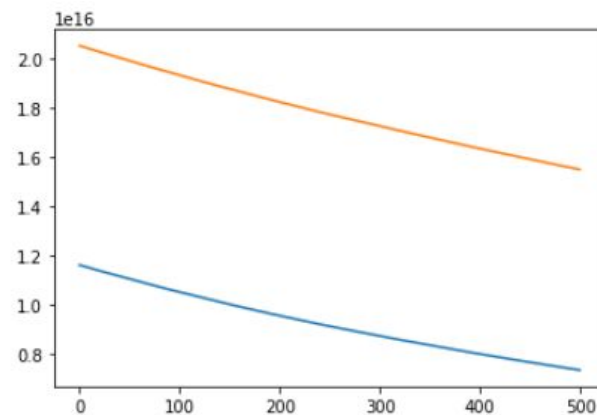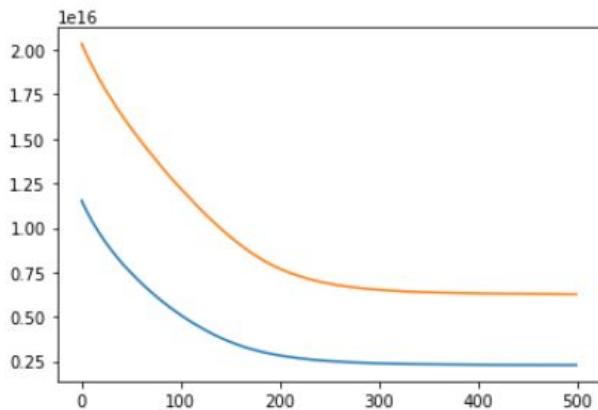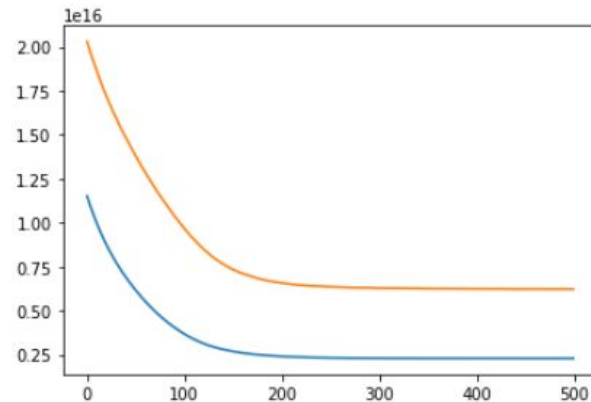
Lr = learning rate

Results

Lr = 100

Lr = 1000

Lr = 10000

Lr = 15000

From the previous slide we can see how the modules accuracy increased as the loss value decreased when the learning rate of the model was increased over 4 iterations of the model.

There is also a major increase in the R2 score through each iteration. With an increase originally from 0.01 to a score of 0.794 this is showing our model trained to be pretty accurate, just how we wanted it

```
1: The R2 score on the Train set is:    0.011
1: The R2 score on the Test set is:     0.023

2: The R2 score on the Train set is:    0.323
2: The R2 score on the Test set is:     0.339

3: The R2 score on the Train set is:    0.777
3: The R2 score on the Test set is:     0.795

4: The R2 score on the Train set is:    0.805
4: The R2 score on the Test set is:     0.794
```

In our last image we can see the scatter plot of the data with the linear regression model formula showing how it represents the data in an accurate manner