

Project 1

For this project I tried to build model with EBImage in R but couldn't produce any significant results and hence used keras in python to proceed with analysis. The image analysis in R was performed by using EBImage package with three tools: Computefeature.shape, Computefeature.moment and Computefearue.basic that on the whole provided us with 19 independent variable which basically provided only the physical aspects of the image rendering insignificant validation to the prediction model. Even after proceeding with the analysis, I was able to have model performance value of only 34%.

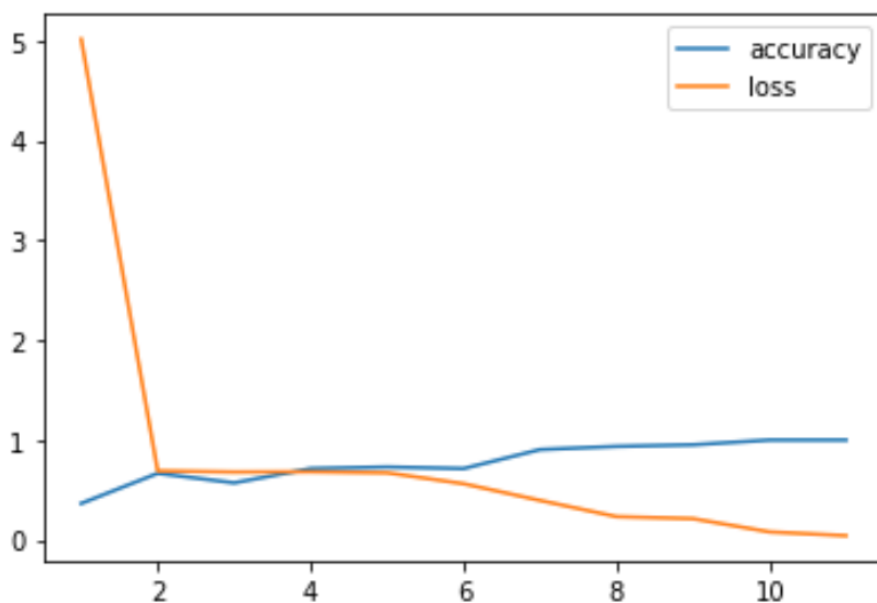
Due to this I performed the analysis using Keras package in python in combination with Neural networks following the steps provided in the supported presentation slides. For better model performance we are required to have at least a dataset of 1000 images.

The predictive model was build using Keras package. The images were resized to 100*100 pixels and depth value of 3(RGB). The epoch value was set to 30 in all cases.

For the **mythological** labels the model performed with an accuracy score for **overall model of 82%**, the additional metrics were as:

index ▼	loss	accuracy	val_loss	val_accuracy	epoch
10	0.0429094135761261	1.0	1.4160023927688599	0.5581395626068115	11

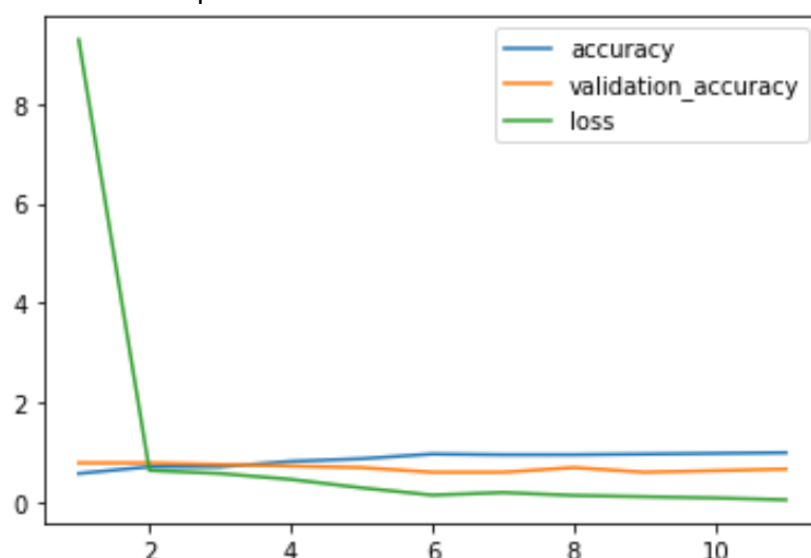
The overall losses dropped significantly for the model. The below plot is a representation of epoch to accuracy and loss of the overall model.



Similar process was performed for **Trees** and the model gave more promising results with

index ▼	loss	accuracy	val_loss	val_accuracy	epoch
10	0.03940586373209953	0.9864864945411682	1.6069831848144531	0.65625	11

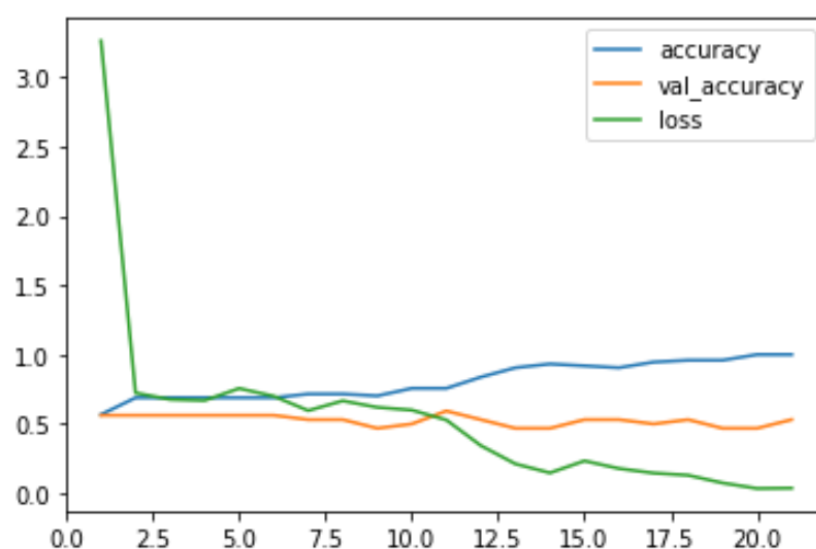
Here we have a representation of losses, accuracy, and validation accuracy (which is for 30% of the dataset). The accuracy score for the **overall model is 89%** and validation accuracy increased to 65% which is impressive.



For **animals** the accuracy score for the **overall model is 85%** and validation accuracy dropped to 53%, the other metrics were as:

index ▼	loss	accuracy	val_loss	val_accuracy	epoch
20	0.037495195865631104	1.0	3.8114848136901855	0.53125	21

The plot representation of various performance measures:



For correlating the text for a given image and the labels, the model provided with a prediction **accuracy score of 62%** and mean squared error of 0.40. I used a linear regression model to build a prediction model. Packages like **tf_idf** to analyze the text content and assigned labels.

The plot represents the predicted value and the test values. As you can see the model still has high degree of randomness.

```
plotGraph(y_test, predictions, "test")
```

