Cats vs. Dogs Classification Report

1. Definition

Project Overview

The project is to write an algorithm to classify whether images contains either a dog or a cat. This is easy for humans, dogs, and cats. However, the computer will find it a bit more difficult. Such a challenge is often called a CAPTCHA[1] (Completely Automated Public Turing test to tell Computers and Humans Apart) or HIP (Human Interactive Proof). HIPs are used for many purposes, such as to reduce email and blog spam and prevent brute-force attacks on web site passwords.

This project will focus on algorithm approaches, from classic Convolutional Neural Network (CNN) to state-of-the-art models (e.g. VGGNet, ResNet, Inception, Xception). I will compare the accuracy and performance of all of these algorithms.

There are many public image datasets online for this challenge. Here, I will use the dataset from Kaggle’s Dogs vs. Cats Redux: Kernels Edition competition[2]. It has two folders: train and test. The train folder contains 25,000 images of dogs and cats. Each image in this folder has the label as part of the filename. The test folder contains 12,500 images, named according to a numeric id. For each image in the test set, I should predict a probability that the image is a dog (1 = dog, 0 = cat).

Project Statement

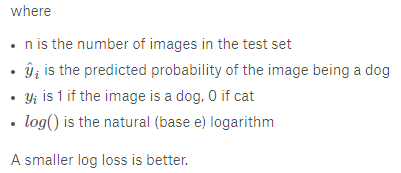
The “Cats vs. Dogs” competition is a supervised binary classification problem.There are two types of images: cats and dogs. The goal is to extract proper features and build an effective model to classify each image contains either cat or dog.

By exploring the dataset, basic information about the dataset can be obtained. The size of each image could be different and have to be resized as each model has a standard for input image size (e.g. 224\*224 for VGGNet and 299\*299 for Inception). After that, the full dataset (under train folder) should be split into training set and validation set. When the models are being trained, the performance is also evaluated on the testing set. By checking the loss and accuracy on training and validation sets, I can know if the model built is correct and if the model is under- or over-fitting. Finally, I will run prediction on test set and upload the result to Kaggle to see what position I can occupy.

Mertics

Usually we use ROC curve (receiver operating characteristic curve) and PR curve (Precision-Recall curve) to evaluate models for binary classification problem. However, in order to comply with Kaggle’s rule, I will use log loss instead.



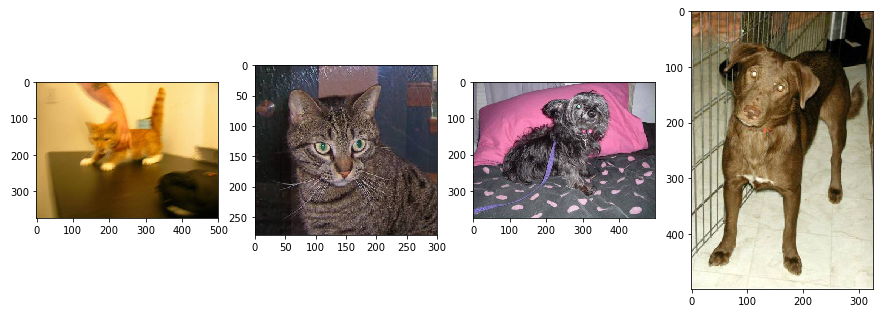


1. Analysis

Data Exploration

The dataset is downloaded from <https://www.kaggle.com/c/dogs-vs-cats-redux-kernels-edition/data>. There are two folders: train and test. The train folder has 25,000 images with names like dog.1.jpg, dog.2.jpg, cat.1.jpg, etc. The test folder has 12,500 images with names like 1.jpg, 2.jpg, 3.jpg, etc.

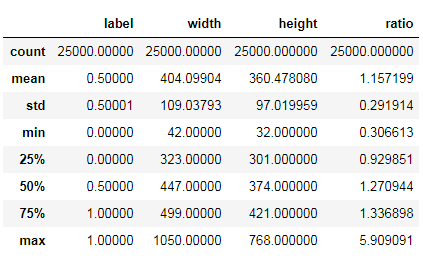
Here are some sample images with associated width and height.



Here are a few things we can notice directly by looking at above images:

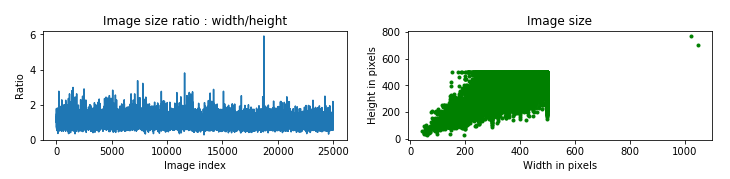
1. Images are generally well centered on the animal.
2. Images have different sizes.
3. The quality of some images are poor (e.g. 1st image).
4. Background images seem to vary a lot and be independent from the categories (the human hand in 1st image and pink pillow in 3rd image).

The following table shows the summary of image info: width, height and ratio (width / height).You can neglect “label” as it is a categorical value. The table clearly shows the sizes of images are varied.



Exploratory Visualization

We have noticed the dataset contains different sizes of images. The following plots show the distribution of each image’s width, height and ratio (width / height).



The average ratio is 1.157.

We can see two outliers in the “Image size” graph which have far too high height and width compared to the rest of the data points. On the “ratio” graph, we can also notice an outlier, which has a ratio of 6 whereas most other images have a ratio below 2.

Since most of models (e.g. VGGNet, Inception) require fixed image size, these graphs can help us determine if we need to resize or crop our images before using them as input of the models.

References

[1] https://www.kaggle.com/c/dogs-vs-cats-redux-kernels-edition