Model used - resnet34 - CNN
Basic structures

a. Convolution- It is a filtering technique whose filter parameters are trained for a given output sample.
    - Depending on the stage of training, different convolutions identifies different properties from the sample input.
b. Batch normalization - normalize the hidden layers so that training is not specific and activations are not generated for a specific features like one color or one shape. Since, there are no off-the-chart activations, we can use higher learning rate to train our model. This makes training faster. It also reduces overfitting.
    - It operates over batch by subtracting from the batch mean and dividing by batch standard deviation. https://towardsdatascience.com/batch-normalization-in-neural-networks-1ac91516821c
c. Relu - Rectified linear unit
    - This is the non-linear part of deep learning model which lets model to train like a complex set of linear equations interacting in a non-linear fashion.
d. Max pool- It downsamples the previous layer by taking the most activated neural out of a pool of nearby pixels.
    - It helps in identifying features such as edges, corners etc.

Dataset:

a. downloaded images from google using the steps given in fastai lesson2_download.
b. Categories - Fat, Slim
c. Binary classification problem, i.e. logistic regression instead of linear regression.
d. Cost - $-\sum Yi\, log(Pi) \ + \ (1-Yi)log(1-Pi)$
e. Adam optimizer with momentum and all.
f. Training data 80% mix of fat and slim.
g. Validation data 20%
h. Normalize data with imagenet_stats (mean and std dev. To be operated over each pixel of an image.)

Training steps-

a. First trained the model with freezing the starting layers.
b. Did 4 epochs with default learning rate of hopefully 1e-3
c. Unfroze the initial layers.
d. Did lr_find which tries to find out the learning rate when loss drops the most.
e. Use this learning rate for outer layer and split learning rate for lower layers with lr/10.
    i. Range (lr/10, lr)
f. See the inference results.
g. Look at worst predictions and try to sense what might be wrong with data first.
h. Fix the data.
i. Try to play around with learning rate and epochs to see which gives better accuracy and how the error improves.
j. Keep a close eye on training and validation losses with each epoch.

Inference results - 87 to 91 % accuracy

```
                 Confusion matrix
         ////////////////////////////////////////
         /               /                 /
Fat      /      29       /        4        /
         /               /                 /
 Actual////////////////////////////////////////
         /               /                 /
 Slim    /        3      /        40       /
         /               /                 /
         ////////////////////////////////////////
                  Fat              Slim
/////////predicted ///////////////////////////
```

Accuracy - true positives (fat + slim) / total

Observation

a. Most difficult task is to clean the data.
b. Training accuracy depends on learning rate.
    i. Higher learning rate helps to achieve the global optimum faster.
    ii. Lower learning rate can get stuck at local minima.
    iii. Do not overfit before unfreezing (starting layers), your model may never come out of local minima
    iv. With increase in epochs i.e. more and more training which tries to fit to the data, the training loss should always reduce.
    v. Validation loss also reduces with training loss with more epochs.
        1. The usual case optimum case is reached when validation loss starts to increase.
        2. If validation loss is not decreasing and stuck at a point but training loss is reducing, it means you need more training data to represent other general cases.

Concepts explored -

a. fit_one_cycle - in this case learning rate is first increased and then decreased slowly to avoid local minimums and reach to optimal point quicker. This whole cycle is one cycle.
b. Lr_find to find the best learning rate.
c. Validation loss drops to a certain point untill model reaches minima with more epochs or higher degree of polynomial.

Library used - fastai.vision

Conclusion:

a. Explored a binary classification problem using deep learning technique. Used state of the art image model resnet34 with fastai library to clean the data and train the model.

Future work:

a. Try more training set
b. Use resnet50.

c. Try to reach accuracy of more than 95%.
d. Take state of the art classification problem from kaggle competition.