Progress report: Winter ProjectDoubts regarding winter project (Classification of infected cells)

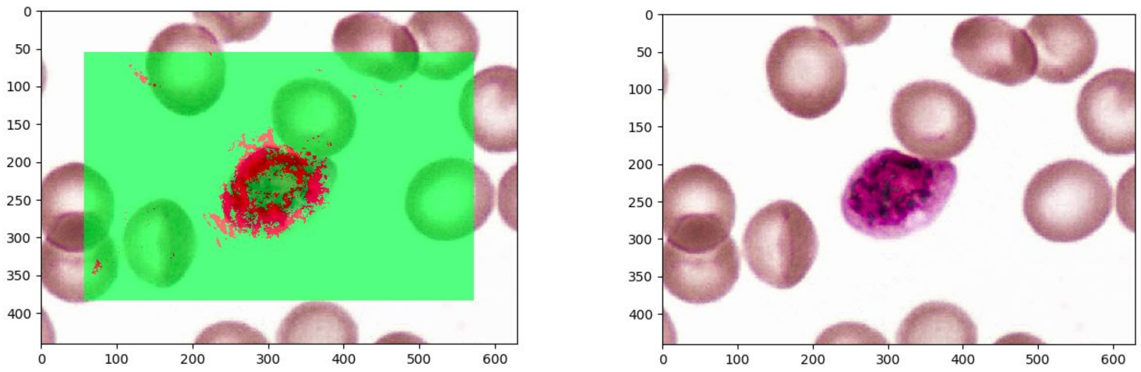


Figure demonstrating pixel-wise detection of malarial plasmodium

(Green - “Normal”, Red - “Infected”)

Introduction:

The dataset seems to be of be of "thin blood films" from malaria tests. Visually there are a lot of variations in the images as if the tests were recorded from different instruments. Another reason for such variation in the dataset is the dyes used to create the stain. As discussed with you, I had to adjust the images in the dataset such that the classification algorithm works properly. I have tried many things and I am still working on the various parameters that would improve classification accuracy.

Observations:

The parameters that can be tweaked to improve the accuracy of our model are:

1. Patch size
2. Image size (feature vector)
3. Pixel count thresholding for demarcating regions
4. No. of infected, normal and background images used to train the classifier
5. No. of estimators - Trees in the forest

Choosing patch size as mean size of the bounding box.

Average height for patch = 110.74407373750486

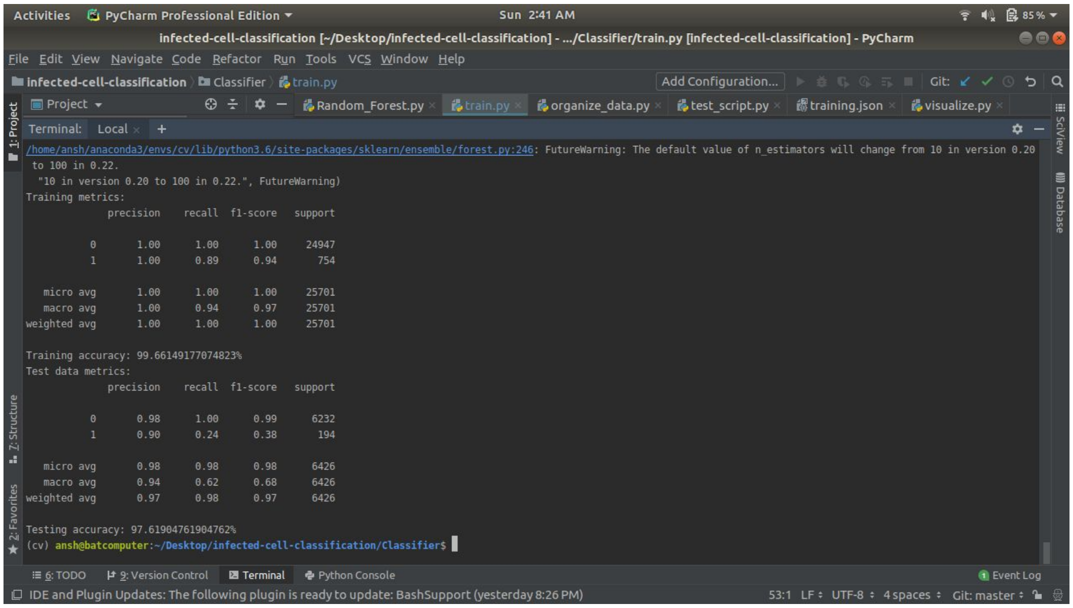
Average width for patch = 111.27817733350642

So we use a 110*110 moving patch.

Results:

1. Training the RandomForestClassifier

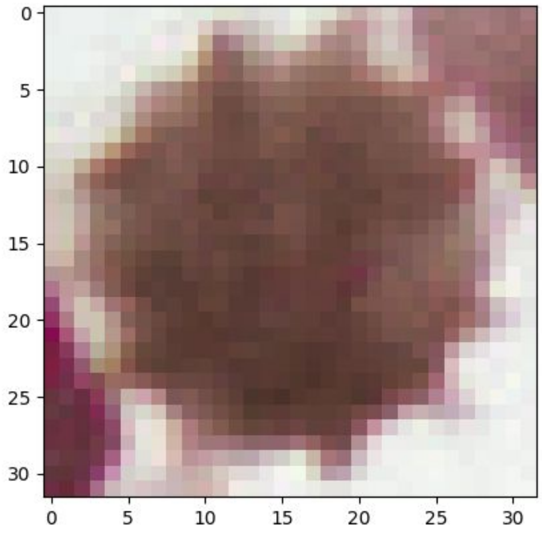
    a. No adjustments made to the image, 32*32*3 features:



```
/home/ansh/anaconda3/envs/cv/lib/python3.6/site-packages/sklearn/ensemble/forest.py:246: FutureWarning: The default value of n_estimators will change from 10 in version 0.20
 to 100 in 0.22.
  "10 in version 0.20 to 100 in 0.22.", FutureWarning)
Training metrics:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     24947
           1       1.00      0.89      0.94       754

   micro avg       1.00      1.00      1.00     25701
   macro avg       1.00      0.94      0.97     25701
weighted avg       1.00      1.00      1.00     25701

Training accuracy: 99.66149177074823%
Test data metrics:
              precision    recall  f1-score   support

           0       0.98      1.00      0.99      6232
           1       0.90      0.24      0.38       194

   micro avg       0.98      0.98      0.98      6426
   macro avg       0.94      0.62      0.68      6426
weighted avg       0.97      0.98      0.97      6426

Testing accuracy: 97.61904761904762%
(cv) ansh@batcomputer:~/Desktop/infected-cell-classification/Classifier$
```

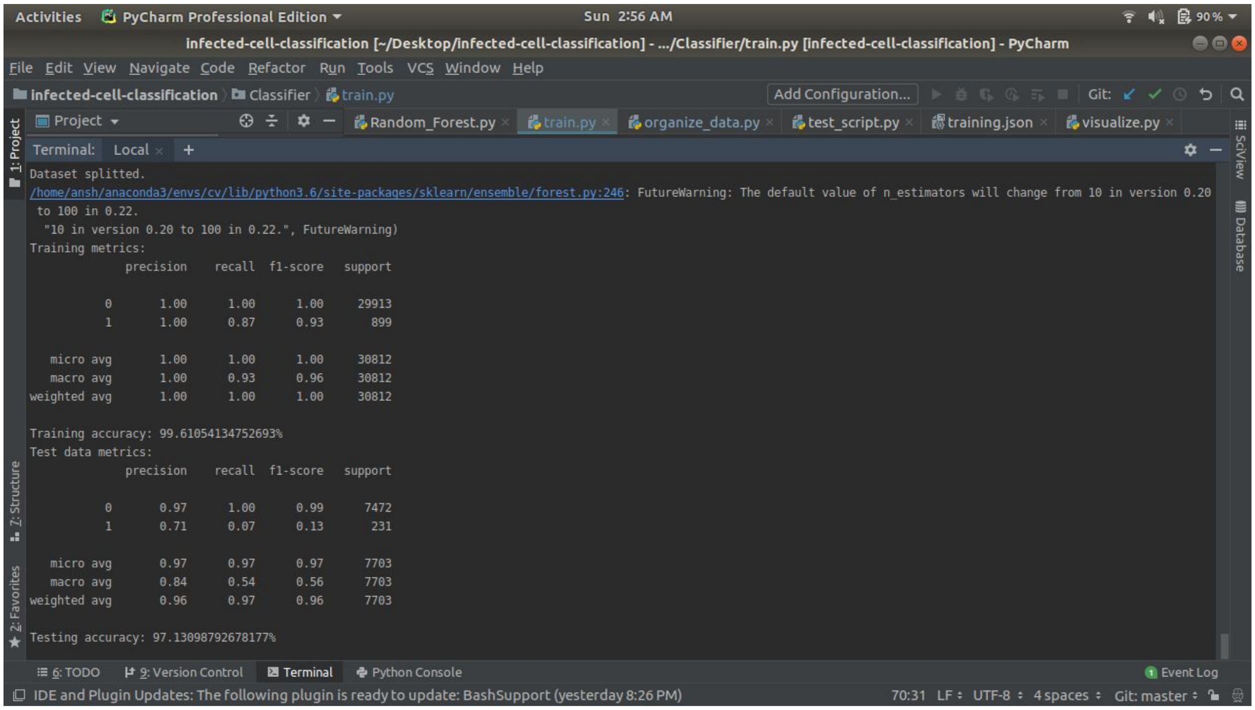
IDE and Plugin Updates: The following plugin is ready to update: BashSupport (yesterday 8:26 PM)

53:1 LF UTF-8 4 spaces Git: master



Sample image used to train the classifier.

b. Grayscale images, 32*32 features:

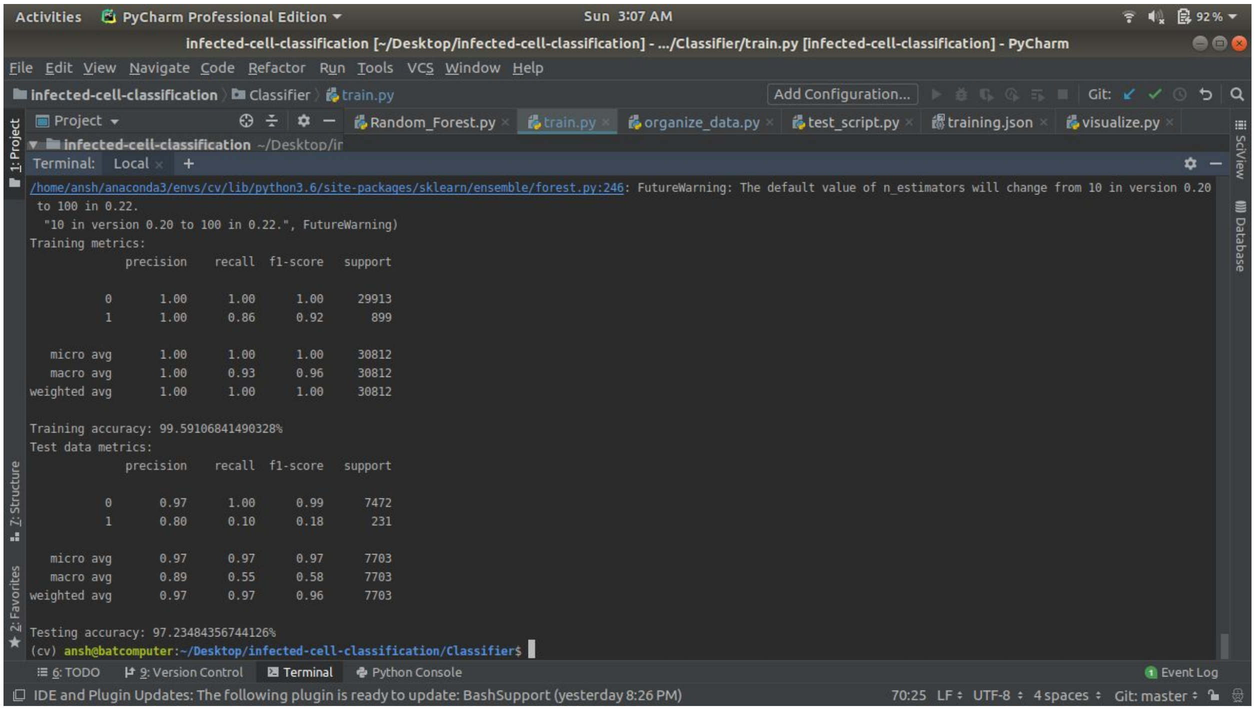

Activities
PyCharm Professional Edition
Sun 2:56 AM
90 %
infected-cell-classification [~/Desktop/infected-cell-classification] - .../Classifier/train.py [infected-cell-classification] - PyCharm
File Edit View Navigate Code Refactor Run Tools VCS Window Help
infected-cell-classification
Classifier
train.py
Add Configuration...
Git:
Project
Random_Forest.py
train.py
organize_data.py
test_script.py
training.json
visualize.py
Terminal:
Local
1: Project
7: Structure
2: Favorites
SciView
Database
6: TODO
9: Version Control
Terminal
Python Console
Event Log
IDE and Plugin Updates: The following plugin is ready to update: BashSupport (yesterday 8:26 PM)
70:31
LF
UTF-8
4 spaces
Git: master

```
Dataset splitted.
/home/ansh/anaconda3/envs/cv/lib/python3.6/site-packages/sklearn/ensemble/forest.py:246: FutureWarning: The default value of n_estimators will change from 10 in version 0.20
 to 100 in 0.22.
  "10 in version 0.20 to 100 in 0.22.", FutureWarning)
Training metrics:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     29913
           1       1.00      0.87      0.93       899

   micro avg       1.00      1.00      1.00     30812
   macro avg       1.00      0.93      0.96     30812
weighted avg       1.00      1.00      1.00     30812

Training accuracy: 99.61054134752693%
Test data metrics:
              precision    recall  f1-score   support

           0       0.97      1.00      0.99      7472
           1       0.71      0.07      0.13       231

   micro avg       0.97      0.97      0.97      7703
   macro avg       0.84      0.54      0.56      7703
weighted avg       0.96      0.97      0.96      7703

Testing accuracy: 97.13098792678177%
```

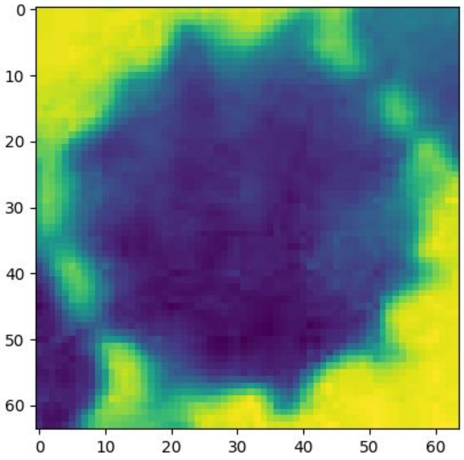
We observe that using grayscale images increasing the accuracy of the model on the training set, however this step decreases the validation accuracy by a small margin.

Clearly the number of features gets reduced when grayscale images are used instead of RGB images. The size of the training set remains the same implying that the dip in validation accuracy can possibly attributed to overfitting.

c. Grayscale Images, 64*64 features:


Activities
PyCharm Professional Edition
Sun 3:07 AM
92 %
infected-cell-classification [~/Desktop/infected-cell-classification] - .../Classifier/train.py [infected-cell-classification] - PyCharm
File Edit View Navigate Code Refactor Run Tools VCS Window Help
infected-cell-classification
Classifier
train.py
Add Configuration...
Git:
Project
Random_Forest.py
train.py
organize_data.py
test_script.py
training.json
visualize.py
infected-cell-classification ~/Desktop/ir
Terminal:
Local
1: Project
7: Structure
2: Favorites
SciView
Database
6: TODO
9: Version Control
Terminal
Python Console
Event Log
IDE and Plugin Updates: The following plugin is ready to update: BashSupport (yesterday 8:26 PM)
70:25 LF UTF-8 4 spaces Git: master

```
/home/ansh/anaconda3/envs/cv/lib/python3.6/site-packages/sklearn/ensemble/forest.py:246: FutureWarning: The default value of n_estimators will change from 10 in version 0.20
 to 100 in 0.22.
  "10 in version 0.20 to 100 in 0.22.", FutureWarning)
Training metrics:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     29913
           1       1.00      0.86      0.92       899

   micro avg       1.00      1.00      1.00     30812
   macro avg       1.00      0.93      0.96     30812
weighted avg       1.00      1.00      1.00     30812

Training accuracy: 99.59106841490328%
Test data metrics:
              precision    recall  f1-score   support

           0       0.97      1.00      0.99      7472
           1       0.80      0.10      0.18       231

   micro avg       0.97      0.97      0.97      7703
   macro avg       0.89      0.55      0.58      7703
weighted avg       0.97      0.97      0.96      7703

Testing accuracy: 97.23484356744126%
(cv) ansh@batcomputer:~/Desktop/infected-cell-classification/Classifier$
```



Sample image used for training the classifier

Future course of action:

1. Normalization of stains

I have to pick sets of images that look alike and then normalize to any one target image. I have to devise a method of grouping similar images and automate the transformation process.

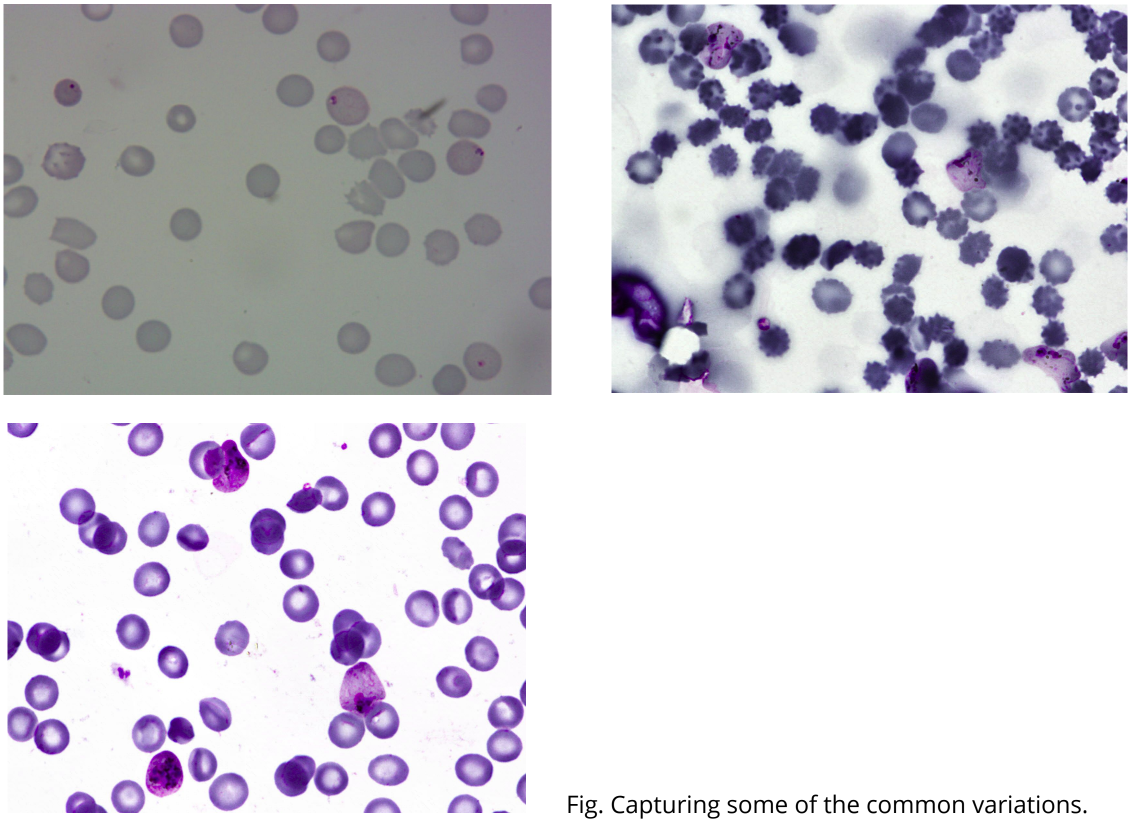
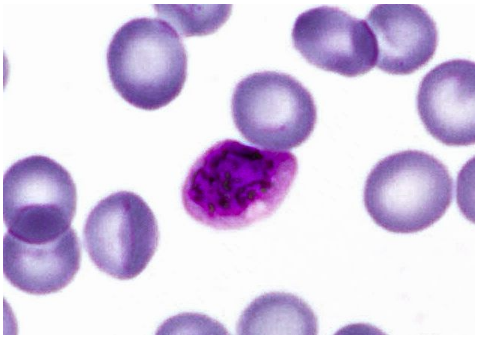
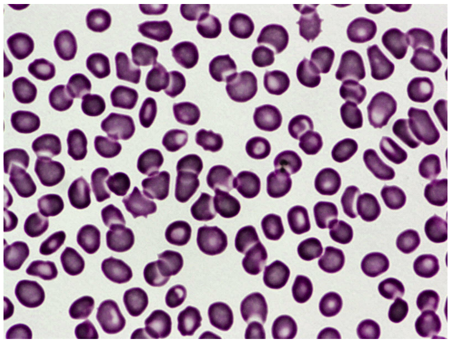


Fig. Capturing some of the common variations.

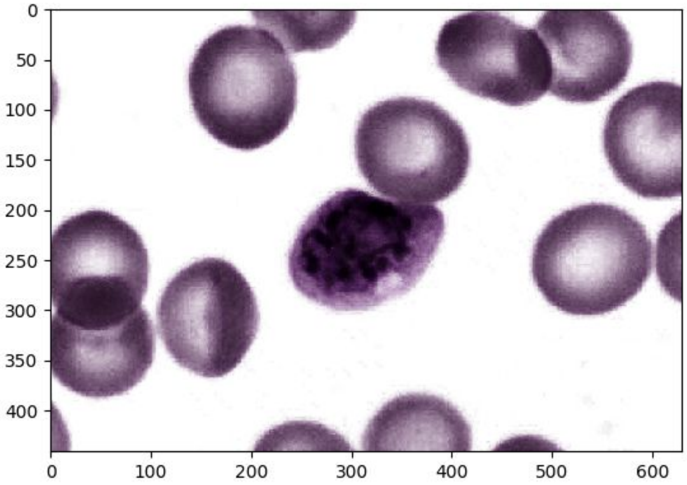
A demo of the transformation process



Original Image



Target Image



Transformed Image

2. Tweaking of the observed parameters

   I have to work on the above mentioned parameters. Also, morphological opening and closing can improve some "noise" classifications and close the infected regions.

Here's the link to my codebase: https://github.com/AnshKhurana/infected-cell-classification