COMP9417 Project: Machine Learning vs. Cancer

March 28, 2022

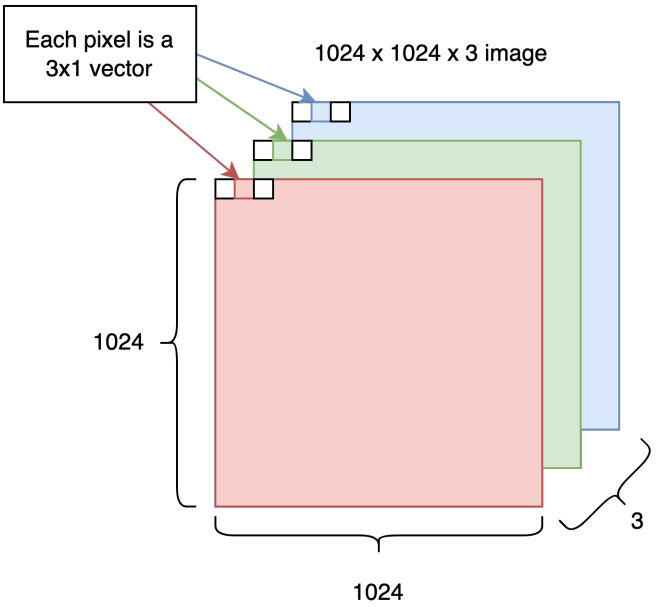
Project Description

As a Data Scientist at Predictive Solutions Inc., you have become comfortable working with any type of dataset that comes your way. Your newest client runs the pathology department at the local hospital and they are interested in utilizing machine learning to more efficiently classify histologic images (in layman's terms, these are images of human tissue under a microscope). The images are usually analysed by a pathologist who assigns each image to one of four possible classes. Class 0 indicates that no tumor is present, Classes 1-3 indicate that cancer is present, with each of these indicating a different type of tumor (for our purposes, we can just think of them as types 1-3). The client has provided you with a small set of high resolution images that have already been classified and is hoping that you can build a model that achieves better performance than the human experts.

The actual data will be released on March 28, 2022.

Description of the Data

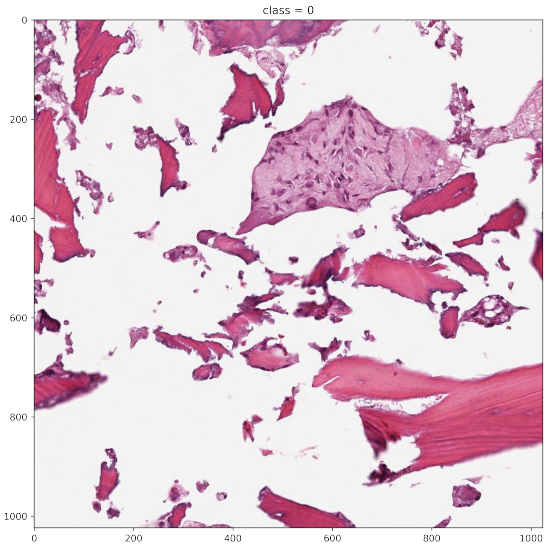
You have been provided with a zip file containing 3 data files: `X_train.npy`, `y_train.npy` and `X_test.npy`. The training dataset is composed of 858 high resolution images, each image (these are the x_i's for this problem) has dimension $1024 \times 1024 \times 3$. This is best understood by considering the following visual representation.



Each pixel is a 3×1 vector, one element for each of the three color channels (red, green, blue), and each image is composed of $1024 \times 1024 = 1048576$ such pixels. The pixel values have been standardized so that each element of the vector is a number between 0 and 1. This number captures the *intensity* of the image at that particular location, values closer to 1 mean higher intensity (darker), and values close to 0 are lower intensity (lighter). For our problem, you can visualize one of these images by running the following code:

```
X_train=np.load("X_train.npy")
y_train=np.load("y_train.npy")
```

```
plt.figure(figsize=(10,10))
i=14
plt.imshow(X_train[i])
plt.title(f'class={int(y_train[i])}')
plt.savefig("image14.png",dpi=600)
plt.show()
```



The test dataset is composed of 287 images of the same dimension as in the training dataset. You must submit your predictions for these images in the same exact format as `y_train.npy` (i.e. your submission must be a numpy array with shape `(287,)`). The order of these predictions must obviously match the order in `X_train.npy`. Predictins will be evaluated using the class-weighted $f1$ score, i.e. `scklearn.metrics.f1_score(y_true, y_pred, average='weighted')`.

Optional Advice

- The dataset is very large and so you may run into memory issues if you try to load everything in at once. One approach to deal with this is to look into generators that load in smaller chunks of data and release them from memory once they have been used (i.e. for an iteration of gradient descent). The PyTorch DataLoader is an example of this, see here for example.
- On the previous point, when prototyping models, do not run them on the entire dataset at first - use a smaller sample to make sure things are working, then deploy your model on the full dataset.
- The images themselves are quite big, and so you will most likely need to come up with smart ways to reduce the size of the image. You will therefore need to implement some sort of feature extraction, which would allow you to represent each image as a lower dimensional object that captures most of the information in the original image. Some models (like CNNs) do this automatically. Be sure to include a detailed description of your approach in the report.
- A naive approach would treat each image as a vector (this would involve collapsing the image into a single vector of dimension $1024 \times 1024 \times 3 \sim 3$ million, and running a simple model (say logistic regression) with inputs of this size. This is obviously a poor approach because it removes all the spatial information from the problem. Another, potentially more reasonable approach, is to collapse the 3 color channels into one. For example, the top left pixel $p_0 = [p_{00}, p_{01}, p_{02}]^T$ can be summarized as a one-dimensional number by coming up with weights a, b, c and calculating $\tilde{p}_0 = ap_{00} + bp_{01} + cp_{02}$, which would effectively reduce the problem to dimension 1024×1024.
- While some machine learning algorithms generate features automatically, you might also want to look into generating features by hand. There is a vast literature on machine learning for histolic images and you may want

to look at some common features that others have used in similar problems. At the very least this approach would set-up a simple baseline that can be used to judge performance of your more advanced models. Be sure to cite any references used in your report.

Overview of Guidelines

- The deadline to submit the report is 5pm April 20. The deadline to submit your predictions is 5pm April 17 (Sydney time) for the Internal Challenge project (ML vs. Cancer) and 11:59pm US Eastern Time April 17 for the TracHack project.
- Submission will be via the Moodle page
- You must complete this work in a group of 3-5, and this group must be declared on Moodle under Group Project Member Selection
- The project will contribute 30% of your final grade for the course.
- Recall the guidance regarding plagiarism in the course introduction: this applies to all aspects of this project as well, and if evidence of plagiarism is detected it may result in penalties ranging from loss of marks to suspension.
- Late submissions will incur a penalty of 5% per day **from the maximum achievable grade**. For example, if you achieve a grade of 80/100 but you submitted 3 days late, then your final grade will be $80 - 3 \times 5 = 65$. Submissions that are more than 5 days late will receive a mark of zero. The late penalty applies to all group members.

Objectives

In this project, your group will use what they have learned in COMP9417 to construct a classifier for the specific task as well as write a detailed report outlining your exploration of the data and approach to modelling. The report is expected to be 10-12 pages (with a single column, 1.5 line spacing), and easy to read. The body of the report should contain the main parts of the presentation, and any supplementary material should be deferred to the appendix. For example, only include a plot if it is important to get your message across. The report is to be read by the client, and the client cares about the big picture, pretty plots and intuition. The guidelines for the report are as follows:

1. Title Page: tile of the project, name of the group and all group members (names and zIDs).
2. Introduction: a brief summary of the task, the main issues for the task and a short description of how you approached these issues.
3. Exploratory Data Analysis: this is a crucial aspect of this project and should be done carefully given the lack of domain information. Some (potential) questions for consideration: are all features relevant? How can we represent the data graphically in a way that is informative? What is the distribution of the classes? What are the relationships between the features?
4. Methodology: A detailed explanation and justification of methods developed, method selection, feature selection, hyper-parameter tuning, evaluation metrics, design choices, etc. State which method has been selected for the final test and its hyper-parameters.
5. Results: Include the results achieved by the different models implemented in your work, with a focus on the f1 score. Be sure to explain how each of the models was trained, and how you chose your final model.
6. Discussion: Compare different models, their features and their performance. What insights have you gained?
7. Conclusion: Give a brief summary of the project and your findings, and what could be improved on if you had more time.
8. Reference: list of all literature that you have used in your project if any. You are encouraged to go beyond the scope of the course content for this project.

Project implementation

Each group must implement a minimum of two classification methods and select the best classifier, which will be used to generate predictions for the test sets of the respective task. You are free to select the features and tune the methods for best performance as you see fit, but your approach must be outlined in detail in the report. You may also make use of any machine learning algorithm, even if it has not been covered in the course, as long as you provide an explanation of the algorithm in the report, and justify why it is appropriate for the task. You can use any open-source libraries for the project, as long as they are cited in your work. You can use all the provided features or a subset of features; however you are expected to give a justification for your choice. You may run some exploratory analysis or some feature selection techniques to select your features. There is no restriction on how you choose your features as long as you are able to justify it. In your justification of selecting methods, parameters and features you may refer to published results of similar experiments.

Code submission

Code files should be submitted as a separate `.zip` file along with the report, which must be `.pdf` format. Penalties will apply if you do not submit a pdf file (do not put the pdf file in the zip).

Peer review

Individual contribution to the project will be assessed through a peer-review process which will be announced later, after the reports are submitted. This will be used to scale marks based on contribution. Anyone who does not complete the peer review by the 5pm Thursday of Week 11 (29 April) will be deemed to have not contributed to the assignment. Peer review is a confidential process and group members are not allowed to disclose their review to their peers.

Project help

Consult Python package online documentation for using methods, metrics and scores. There are many other resources on the Internet and in literature related to classification. When using these resources, please keep in mind the guidance regarding plagiarism in the course introduction. General questions regarding group project should be posted in the Group project forum in the course Moodle page.