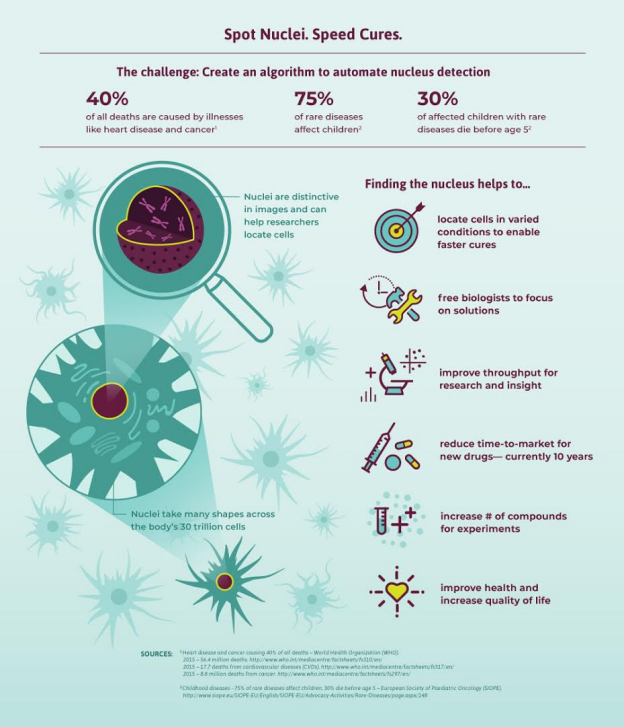
CS342 Machine Learning: Assignment #2
Kaggle Competition: 2018 Data Science Bowl

Deadline: March 14, 2018 at 12:00pm

Week 7 of Term 2

Instructor:
Dr Theo Damoulas (T.Damoulas@warwick.ac.uk)

Tutors:
Helen McKay (H.McKay@warwick.ac.uk),
Joe Meagher (J.Meagher@warwick.ac.uk)
Karla Monterrubio-Gomez (K.Monterrubio-Gomez@warwick.ac.uk),
Jevgenij Gamper (J.Gamper@warwick.ac.uk)



Create an account at Kaggle with a user/display name as CS342xxxx where xxxx is your uni ID

Warning: This Kaggle competition is a "live" competition. This means that any breaking of the Rules (by e.g. having multiple user accounts or similar submissions with other users) can lead to an automatic ban from the competition. Plagiarism will not only be detected by our software but also detected by kaggle on your submissions - you have been warned!

In Assignment 2 you will be building Artificial Neural Networks and competing with other ML enthusiasts around the world in order to try and solve a real-world problem: That of finding the nuclei in divergent images to advance medical discovery. Read the full problem description here and learn more about it here.

The Data

We have downloaded the dataset on the local drive `/modules/cs342/Assignment2/` for you to read in the data from there *without* downloading to your accounts. There is also there a smaller subset of only 100 images to get you going initially. Be careful with your quota on the machines.

Kernels and Tutorials

You will find these very useful:

- Reading in images and creating masks
- Building a first CNN
- Using the IoU metric
- More on the IoU metric
- A watershed tutorial in scikit

The Assignment

Your assignment submission should include *all* of the following: a compressed (.zip) file containing your Report (.pdf) and all your Python codes (.py) developed for the analysis. Make sure your codes execute on the lab machines, loading in either the full dataset from the local drive or the provided subset from a local copy.

Your Report (**Total maximum of 6 single-sided pages**) should have the following structure in which you address and perform the requested tasks per section.

Abstract

[**Maximum 200 words**]
An abstract summarising your main findings.

Data Exploration, Feature Engineering, and Segmentation

[**Maximum 2 pages**]
This section should include figures and text describing your understanding and exploration of the data, any intuition you gained for constructing or extracting features, and the different feature engineering approaches that you have tried.

Task 1: Show your deep understanding of the specific problem, data and its characteristics via e.g. plots and preliminary analysis. [3 marks]

Task 2: Implement 3 different feature engineering and/or segmentation approaches such as: {HoG (Histogram of Gradients) features, Data augmentation, Bag of visual words representations, SIFT features or free-to-use variants, Watershed segmentation}. [3 marks]

Task 3: Implement a mask prediction/segmentation technique that does not involve the use of Artificial Neural Networks. Use your technique to predict/segment masks in the competition. [1 mark]

Machine Learning Models: Artificial Neural Networks and Deep Learning

[**Maximum 2 pages**]
This section should present the results and analysis with (at least) the models requested below. It should showcase your technical understanding of the models you employed, the tuning you did, any drawbacks, advantages, and their performance. **Give your Kaggle username and best ranking result to provide evidence of your ranking on the leaderboard.**

Use MLPClassifier or Keras to implement your MLP and CNN models.

Task 4: Implement a Multi-Layer Perceptron (Classifier) using raw pixel values (or simple functions of them) as inputs to classify pixels and predict masks. Tune your model parameters and submit your best model predictions to the competition. [2 marks]

Task 5: Implement Multi-Layer Perceptrons (Classifiers) using features derived from your 3 feature engineering approaches as inputs and proceed as above. Tune your model parameters and submit your best model predictions to the competition. [4 marks]

Task 6: Implement a Convolutional Neural Network (e.g. see this simple CNN) using raw pixel values (or simple functions of them) as inputs. Tune your model and submit your best model predictions to the competition. [4 marks]

Progression Graph & Discussion

[**Maximum 1 page**]

Task 7: Construct your *progression graph* throughout the assignment period. That is a graph that has the error metric of the competition (IoU) on the y-axis and the time of submission on the x-axis. Label each submission point and result to indicate what type of model+FE was used. Discuss your individual progression graph, overall conclusions and proposed future directions. [3 marks]

Top ranking performances from the class will be awarded up to 5 marks [5 marks]

Rules of the game & Hints

- We will do our best to run your codes but put every effort to make sure they run! We are using Python 2.6/2.7 version so make sure everything runs ok in 2.6/2.7 before submitting.
- You can use ANY pre-processing, feature extraction, and ML models you want for the 5 performance marks as long as you implement also the requested models from the Tasks.
- Start small. You can use a subset of data to debug and run your CVs, you can subsample (blur) the images to work on lower-dimensional representations, you can use methods like PCA etc.
- Use dropout and other regularisation techniques to avoid overfitting.
- For the CNN you will also have to experiment with the number of convolutional and pooling layers that are doing feature extraction for you. Showcase your understanding in the report.
- Do not send us the data back with your scripts and report. **Your codes should read in the data from the local drive**.

- Your Report should be informative, well structured, and providing details of all aspects of the assignment. The marks assigned to specific tasks can only be fully gained if there is sufficient description and reasoning in the report.
- You can use and experiment with other people's code from the Kaggle website (and beyond) but you should submit your own codes where possible and **clearly identify and acknowledge** what is not yours. Plagiarism will not be tolerated.
- Marks for performance are competitive so don't share your codes or ideas to other students as it will harm your own marks.
- You can use any pre-implemented libraries and models in python like scikit, keras, etc.
- You can use pre-trained CNNs to do "transfer learning" as it was described in class.
- Monitor the ML Forum from the module website for useful hints and further Q&A.