Prediction of differentiation status in C2C12 cells on application of differentiation media using Machine Learning

Aritra Mondal.
17MS168, DBS

September 2021

Contents

Abstract

Studies on cellular differentiation often rely on cell tracking after application of differentiation media on the target cells. Since, the differentiation status of the cells being tracked cannot be known at an early time point, a fraction of these cells fail to differentiate later and have to be discarded. So, tracking proper cells which will differentiate is very crucial. Assuming that cellular differentiation starts to affect the membrane topology of the cell long before the actual differentiation event, a machine learning model was used to find these differences in very early IRM images of these cells and classify into differentiable and non-differentiable classes. Moreover, this data can be used to investigate the relationship between cellular differentiation and membrane topology.

---

1 Introduction

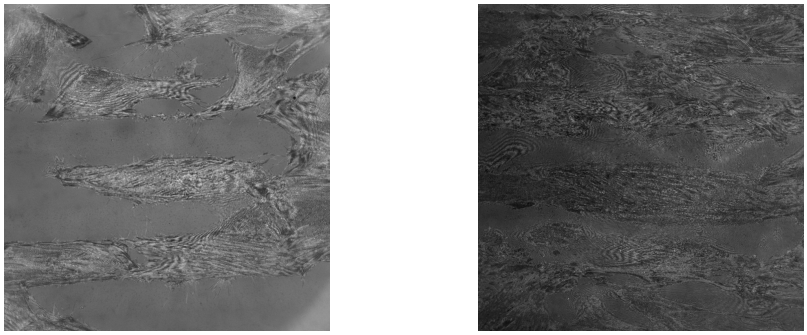
During myogenesis, C2C12 myoblasts undergo an irreversible cell cycle arrest and a gradual increase in expression of muscle function genes. This leads to the fusion of myoblasts into multinucleate myofibers. C2C12 cells undergo several structural and biochemical changes during this process. Although, these changes start to visibly appear after 24-48 hours of application of differentiation media, we hypothesize that small changes in the membrane topology start to appear as early as 2 hours from application of media. We used machine learning to quantify these changes from IRM images of cells whose actual differentiation status was already known. This information was used to build a model that can look at IRM images of C2C12 cells at t=2hr from application of differentiation media and effectively classify them based on their fates.

2 Materials and Methods

2.1 Tissue Culture and Image Acquisition

C2C12 cells were grown in Dulbecco's Modified Essential Medium (DMEM, Gibco, Life Technologies, USA) with 10% fetal bovine serum (FBS, Gibco, HI, US origin) and 1% anti-anti (Gibco). Cells were then seeded in round glass bottom dishes and maintained in growth media. After reaching 60% confluency, growth media was replaced with differentiation media containing DMEM, 2% horse serum, 1% anti-anti and 0.1% insulin. Following this, cells were incubated in a humidified incubator with 5% CO_2 at 37°C for 4 days. The media was changed every 24 hours during this incubation.

The time point of application of differentiation media on the sample was noted as t=0(start time). Two hours later(t=2h), 22 isolated cells with clearly visible boundaries were imaged using **(insert microscope model here)** Interference Reflection Microscope which outputs 2048 time-lapsed images separated by 50ms per scan. The differentiation status of the chosen cells could not be determined at this point of the experiment. These cells were then further tracked till 96 hours to note the actual differentiation status of the cells.



(a) (b)

Figure 1: (a)C2C12 cells 2h after application of differentiation media (b)C2C12 cells 96h after application of differentiation media

2.2 Data Handling

Out of the 22 time-lapsed images, 2 were found to have a much lower overall image intensity, compared to the rest. Since, the pixel intensity over images is an important parameter to be considered while building the model, low intensity images can act as outliers in the data and introduce bias. So, these two images were discarded, leaving 20 total cell images to work with.

Images of 5 of these 20 cells were randomly chosen and stored for later use as "test data". So, the "test data" comprises of 25% of the total data. The remaining cell images were separated into two classes, "differentiated" and "undifferentiated", based on their actual differentiation status noted at 96 hours from exposure to differentiation media. These labelled images were used for "training" and "validation" of the model.

7 cells were found to be "differentiated" and 8 cells were found to be "undifferentiated" in the training data. The differentiated cells were named 'cell01' through 'cell07', followed by the undifferentiated cells, named 'cell08' through 'cell15'.

Table-1 lists all the cell images and their usage.

2.3 Image Pre-processing

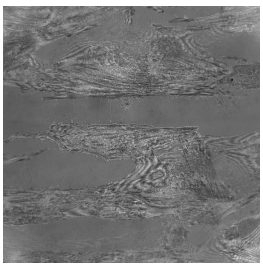
Among the 2048 .tif images in the time-lapses, 15 images were randomly chosen or for every cell. So, the training data comprises of 15*12 = 180 .tif images. Similarly, the validation and testing datasets contain a total of 45 and 75 images, respectively.

Cell name	Differentiation Status	Usage in model
cell01	Differentiated	Training
cell02	Differentiated	Training
cell03	Differentiated	Training
cell04	Differentiated	Training
cell05	Differentiated	Training
cell06	Differentiated	Training
cell07	Differentiated	Validation
cell08	Undifferentiated	Training
cell09	Undifferentiated	Training
cell10	Undifferentiated	Training
cell11	Undifferentiated	Training
cell12	Undifferentiated	Training
cell13	Undifferentiated	Training
cell14	Undifferentiated	Validation
cell15	Undifferentiated	Validation
cell16	Undifferentiated(Randomly Chosen)	Testing
cell17	Differentiated(Randomly Chosen)	Testing
cell18	Undifferentiated(Randomly Chosen)	Testing
cell19	Differentiated(Randomly Chosen)	Testing
cell20	Undfferentiated(Randomly Chosen)	Testing

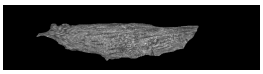
Table 1: Record of C2C12 myoblast images for use in various parts of the model

The resolution of the .tif images was 2048 X 2048 pixels. Generally, processing such big images is computationally resource intensive. Since, reducing the resolution does not reduce the efficacy of the model, the images could be scaled down to a lower resolution, maintaining the same aspect ratio. This is because scaling the resolution while maintaining the aspect ratio does not change the comparative statistical and spatial image features over the whole dataset. Since, our job is classification of the images into two classes based on the differences among them, it does not matter if all the images are subject to the same transformations, unless feature data starts to get lost. If feature data starts to degrade due to incorrect transformations, the accuracy of the model will also decrease. But, since the number of images were low in this case, we did not reduce the resolution, but instead cropped out unused parts of the image.

The images were manually processed using an Image Processing Software, ImageJ to crop out the cell from the surroundings and clear everything outside to black pixels. The images were cropped to 2048 X 512 pixels, while making sure every cells fit completely in these dimensions and were saved as 8-bit grayscale png files.



(a)



(b)

Figure 2: (a)Original Image with the cell of interest outlined (b)Image after pre-processing

2.4 Feature Extraction

The pixels in the image can be categorized into 2 types: black pixels(gray value =0) of the surroundings and non-black pixels(gray value > 0). A variety of parameters were chosen as early visual markers of differentiation. Description of these parameters are noted:

1. **rawDensity**: Sum of all pixel gray values in the image.
2. **meanGrayValue**: Sum of gray values of all non-black pixels divided by the total number of non-black pixels in the image gives the mean gray value of the image.
3. **GrayStdDev**: Standard deviation of the gray values of non-black pixels from the mean gray value of the image.
4. **GrayMed**: Median gray value of the cell image.
5. **prcnt80**: Pixel gray value of 80th percentile pixel.
6. **prcnt20**: Pixel gray value of 20th percentile pixel.
7. **nHigh**: The number of pixels having a pixel gray value more than the value of the 80th percentile pixel.
8. **nLow**: The number of pixels having a pixel gray value less than the value of the 20th percentile pixel.
9. **nHighLowRatio**: Ratio of nHigh to nLow.
10. **area**: Total number of non-black pixels gives the relative projected (2D) area of the cell.
11. **perimeter**: The relative perimeter of the cell calculated by counting number pixels at the boundary of non-black pixels
12. **circularity**: Measure of the roundness of the cell shape, given by: $C = 4\pi * \frac{Area}{Perimeter^2})$
13. **length**: The approximate pixel count of the cell in its major axis gives the relative length of the cell.
14. **meanwidth**: The average width of the cell in its minor axis across its length, found by pixel counting in the minor axis.
15. **maxwidth**: The maximum width of the cell in its minor axis across its length, found by pixel counting in the minor axis. Generally, the maximum width is found near the centre of the major axis.
16. **aspectRatio**: The ratio of length and the maximum width of the cell.
17. **ar1**: Ratio of width near the thinner end to the maximum width across the major axis. This gives a measure of the tapering shape of the cell.

These features were computationally calculated for every image and the extracted values were saved to a .csv file, which the model can read.

2.5 Model Selection

Initially, Principal Component Analaysis(PCA) was performed on the data to reduce its dimensionality and plot it in a 2D graph, without much information loss. PCA creates new uncorrelated variables from the given input features, while successively maximizing its variance. These principal components or PCs can be used to find the decision boundary as maximizing variation among all data points leads to objects with similar parameters being plotted close to each other on the feature space, whereas dissimilar parameters (indicating different class) objects being plotted distinctly. This can provide a qualitative idea of the extent of classification that can be done on the dataset.

A Receiver Operating Characteristic curve (ROC curve) was plotted for the training and validation data using various commonly used models with the false positive rate on x-axis vs the true positive rate on y-axis. The area under curve for each model is shown in Figure-3. Since all the models show equally good AUC scores, choosing any model among these would be fine for this dataset. A Random Forests Classifier was selected for the base model, which is a network of interconnected Decision Trees and is widely used for classification models. The model building, training, validation and testing were done using Python 2.9 with the data science package scikit-learn, among others.

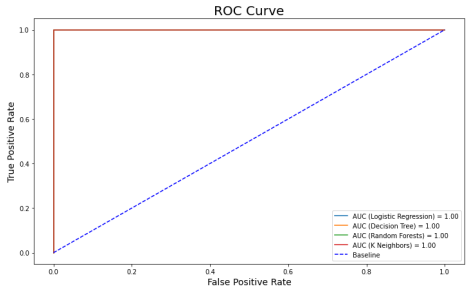
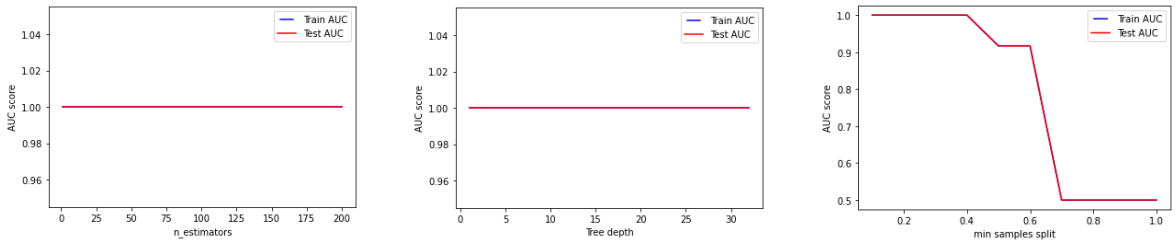


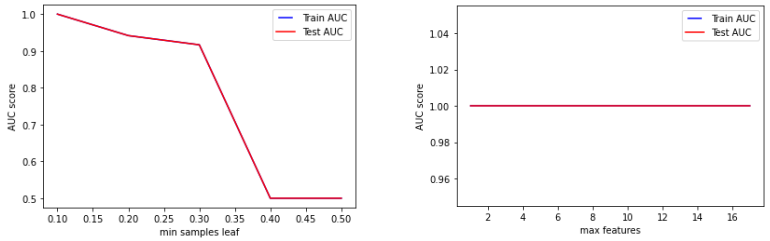
Figure 3: ROC-AUC curves for model Selection

2.6 Hyperparameter Tuning

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. AUC (Area Under Curve) was used as the evaluation metric.



(a) n_estimators (b) max_depth (c) min_samples_split



(a) min_samples_leaf (b) max_features

Figure 4: Hyperparameter Tuning with AUC as a metric

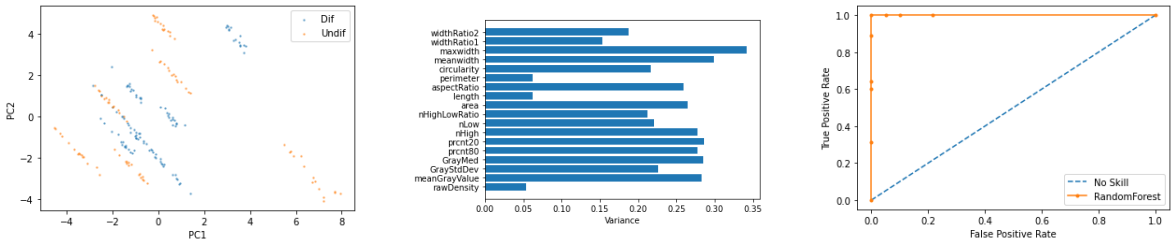
"n_estimators" (Figure 4(a)) represents the number of trees in the forest. Usually, a higher number of trees helps the model to learn the data in a better way. However, a lot of trees can be computationally resource intensive and can also increase the chances of overfitting the data. So, a parameter search can help to find the

sweet spot. "max_depth" (Figure 4(b)) represents the depth of trees in the forest. Deeper trees capture more information about the data but can often lead to overfitting. So, tuning this parameter can save the data from overfitting. "min_samples_split" (Figure 4(c)) is the minimum number of samples required to split an internal node. Increasing this parameter makes each tree in the forest more constrained and consider more samples at each node. "min_samples_leaf" (Figure 4(d)) denotes the minimum number of samples required to be at a leaf node. Higher values of both the "min_samples_split" and "min_samples_leaf" parameters were found to be underfitting the data, so smaller values were chosen. "max_features" (Figure 4(e)) represents the number of features to consider when looking for the best split.

The final hyperparameter values chosen after tuning are n_estimators=5, max_depth=2, min_samples_split=0.2, min_samples_leaf=0.1 and max_features=5.

3 Results

PC1 (principal component 1) was plotted against PC2 (principal component 2) to check the feature space of the data on the PCA plot (Figure 5(a)). Multiple clusters were seen as the 15 images for each cell were very similar to each other. Even then, the differentiated cells occupied kind of a distinct space separate from the undifferentiated cells, although the decision boundary was not very precise. There were a few outlier cells too. The PCA plot of PC1 vs PC2 and a bar chart of features weights is given in Figure 6.



(a) (b) (c)

Figure 5: (a)PCA plot for training and validation data, (b)PCA scalings for training and validation data, (c)AUC-ROC Curve

The "most important" features, i.e., the features having the greatest variances or weights in this classification model can be determined from the feature scalings(Figure-5(b)). Clearly, the "maxwidth" and "meanwidth" parameters, which are the maximum and average lengths of the cell in its minor axis, respectively, have the highest weights here. This suggests that cell width is an important early characteristic of differentiating cells. Other major contributing features in descending order of weights are "prcnt20", "GrayMed", "MeanGrayValue", "nHigh", "prcnt80", "area" and "aspectRatio". For a detailed description of these parameters, please check Section 2.4.

Images of 12 cells (180 images) were used to train the model while images of 3 cells (45 images) were used for validation. A Random Forest Classifier was trained for classification using these datasets. A Receiver Operating Characteristic curve (ROC curve) was plotted with the false positive rate on x-axis vs the true positive rate on y-axis in Figure 5(c). The ROC immediatexcly jumps to a very high TPR, indicating correct predictions. The Area Under Curve (AUC) was found to be 1.00.

The trained model outputs a .sav file, which contains the weights of the parameters used. This file can be read using Python and used to predict the differentiation status of other cells at (t=2hr) of application of differentiation media. It was tested on images of the 5 cells kept initially as "test data". The test results are as follows:

Cell name	Final Prediction	Confidence	Actual Status at (t=96h)
cell16	Differentiated	69.73%	Differentiated
cell17	Undifferentiated	92.73%	Undifferentiated
cell18	Undifferentiated	77.80%	Undifferentiated
cell19	Undifferentiated	58.47%	Undifferentiated
cell20	Undifferentiated	83.73%	Differentiated

Table 2: Predictions of model on the "test dataset"

As seen from the test results, the model was successful in identifying 4 out of 5 cells correctly. Since, the test data was previously isolated from rest of the data, it was completely new for the algorithm. So, as a measure of accuracy, we can check the number of correct predictions among all test images. In this case, the accuracy was found to be 80%. Although, the amount of data was less, the test accuracy indicates a fairly good fit of the data and as such, this model can be used to predict the differentiation status of C2C12 skeletal cells at 2hr of application of differentiation media, although with some occasional errors.

Apart from this, a few cells with unknown actual differentiation status were provided. Using the algorithm, their differentiation status were predicted. The results are tabulated below in Table-3:

Cell name	Final Prediction	Confidence
testcell01	Differentiated	77.27%
testcell02	Undifferentiated	98.73%
testcell03	Undifferentiated	67.07%
testcell04	Differentiated	77.73%
testcell05	Undifferentiated	57.93%
testcell06	Differentiated	65.67%
testcell07	Differentiated	84.60%

Table 3: Predictions of model on a few more cells

4 References

1. Sommer C, Gerlich DW. Machine learning in cell biology–teaching computers to recognize phenotypes. Journal of cell science. 2013;126(24):5529–39. pmid:24259662

2. Abihith Kothapalli,Hinrich Staecker,Adam J. Mellott Supervised machine learning for automated classification of human Wharton's Jelly cells and mechanosensory hair cells. https://doi.org/10.1371/journal.pone.0245234

3. Schindelin J, Arganda-Carreras I, Frise E, Kaynig V, Longair M, Pietzsch T, et al. Fiji: an open-source platform for biological-image analysis. Nature methods. 2012;9(7):676. pmid:22743772