Lab 4 - STAT 215A
Fall 2015

Team Members: Emin Arakelian, Kuan Cheng, Lei Kang, Fadi Kfoury, Paul Rigge

1 Introduction

Our goal is to build a reproducible model for organ classification in mid-stage Drosophila melanogaster embryos. We need to build classifiers to distinguish gut, yolk and epidermal/mesodermal tissues in embryo images, see Figure 1. One of the challenges here is the use of the Fiji features using some feature engineering that will map them to the spatial feature space. Each pixel has 13 Fiji features. Hence, we need to find a way to map them to the spatial features space, which is in super pixel level, to allow the classifier to utilize them. Another challenge is that the boundaries of the organs are vague and the classifier performs poorly there. This report is arranged as follows: in Section 2, the dataset is introduced. In Section 3, exploratory data analysis is done and our feature engineering is presented. In Section 4, Models are built and results are presented and discussed.

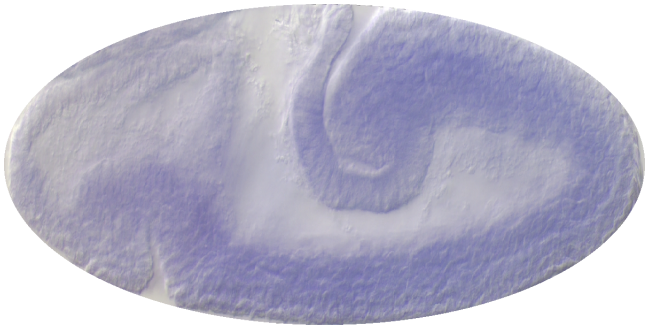

Figure 1: Embryo 5

2 Data Description

We have 170 embryo images; 20 images are set aside as our test set. Each embryo image is 300x600 pixels. We are also given the embryo.seg data set which has for each pixel in each embryo the number of the super pixel to which it belongs. Embryo.label is a dataset where the label for each pixel in each embryo is given after they have been mapped to the super pixels. Embryo.label.raw is very similar to the previous data set with the difference of having labels before being mapped to super pixels. Embryo.seg, Embryo.label and Embryo.label.raw are all 300x600. In super pixel level we are also given the coordinates, distance from the boundary and the label for each super pixel.

3 EDA and Feature Engineering

3.1 Comparing raw labels to expert labels

Here we can take a look and compare the raw labels and the super pixel labels. As we can see the raw labels are more edgy and straight cut, the super pixels are picked by and algorithm and their labels are given by probably voting over the expert labels : Figure 2.

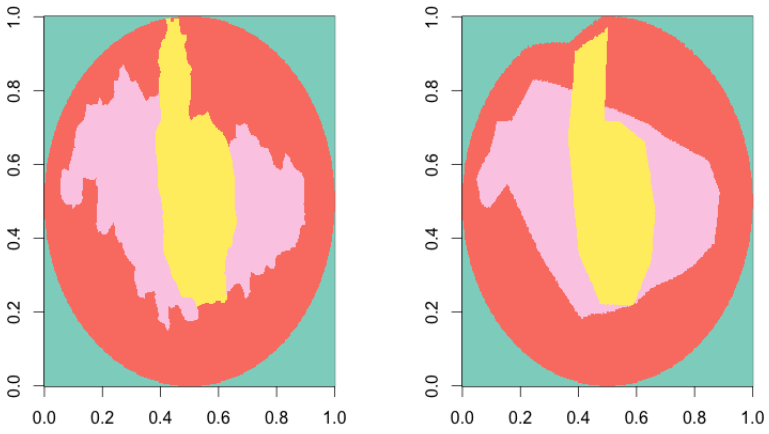

Figure 2: Comparison of expert labels and raw labels for embryo 15

3.2 Exploring Fiji Features

In this lab, we are given 13 Fiji features which are low level image features for each pixel. We are going to explore these features in two approaches. First is the mapped features using the following functions: mean, variance, median, skewness and kurtosis. There are also the features that we later used in our models to map the Fiji features. The second is exploring the Fiji features using the principal components.

3.2.1 Simple mapping

Here, we used median, mean, variance, skewness and kurtosis as our mapping functions to map the Fiji features back to super pixel level. Below we plotted some instances of these mappings to compare their discrimination on our embryos. As seen in Figure 3 and Figure 4 the functions don't seem to explain

much. We did investigate more functions and more embryos but the results weren't much different. Next we explore the same mapping on density plots as depicted in figure 6. Here we can see more clearly how uniform and non-discriminant the functions are.

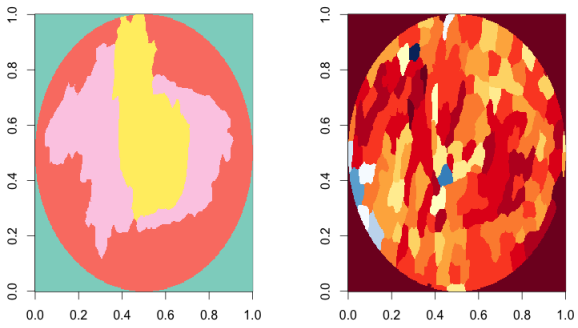

Figure 3: Comparison of sp labels and 2nd Fiji features of the 12th embryo mapped by skewness

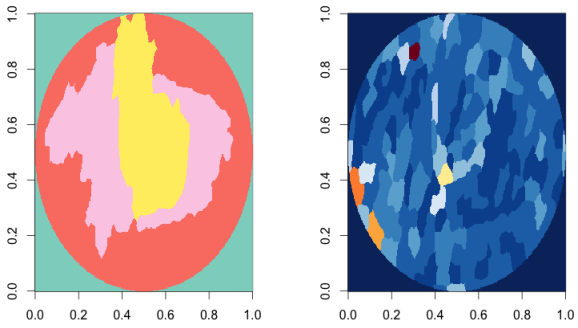

Figure 4: Comparison of sp labels and 2nd Fiji features of the 12th embryo mapped by kurtosis

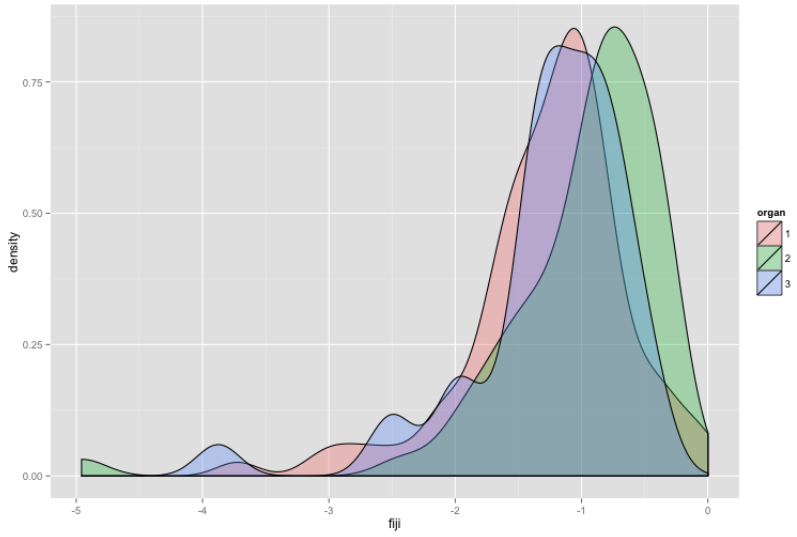

Figure 5: Density plot of 2nd Fiji feature 12th embryo on skewness value across the organs

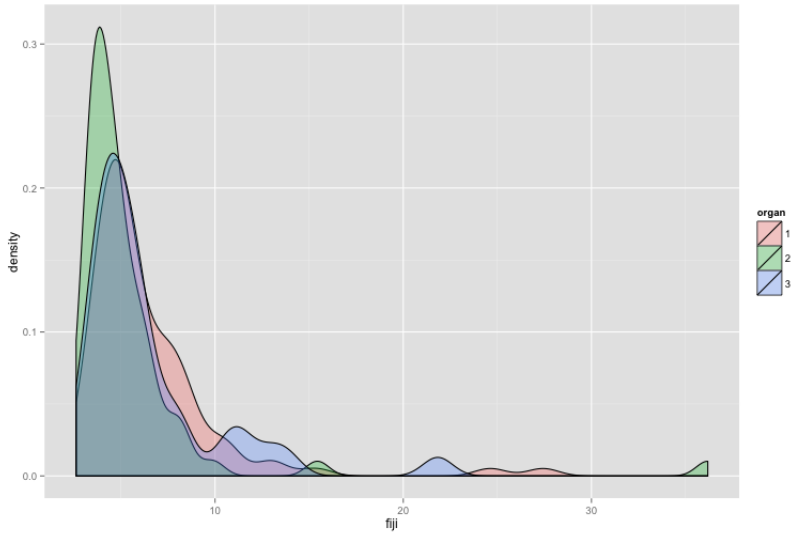

Figure 6: Density plot of 2nd Fiji feature 12th embryo on kurtosis value across the organs

3.2.2 Principal component analysis

Next we tried exploring the Fiji features using the principal components, see Figure 7. We mapped the first principal component using the median, an example of which is in Figure 8. As seen we still cannot have a visible discrimination. In Figure 9 we did a 3D scatter plot of the first three principal components' median who display a mixed behavior.

Then we do PCA in order to make our raw Fiji features' distribution more normal , since we expect that PCA can perform better on data coming from a normal distribution. We took the logarithm of each Fiji feature after shifting them by its minimum value within each feature and adding 0.1 ensuring that the feature is in the logarithm domain. We reproduced those plots, as seen in Figure 10 and Figure 11. The results do not differ by much from the previous implementation. Thus we expect extracting information from the Fiji features to be very challenging.

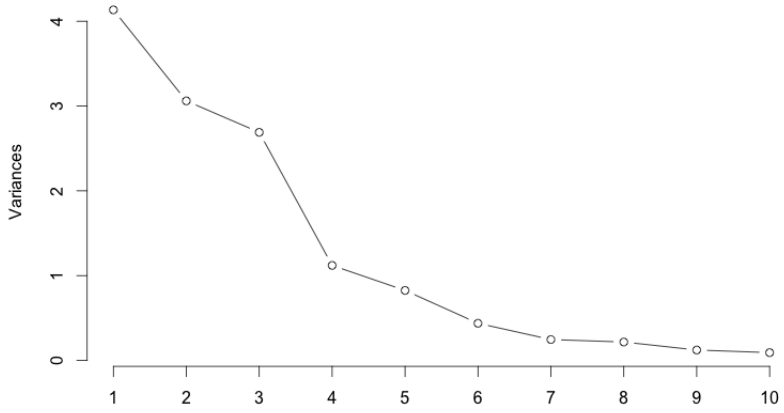

Figure 7: Variance decay plot by PCA

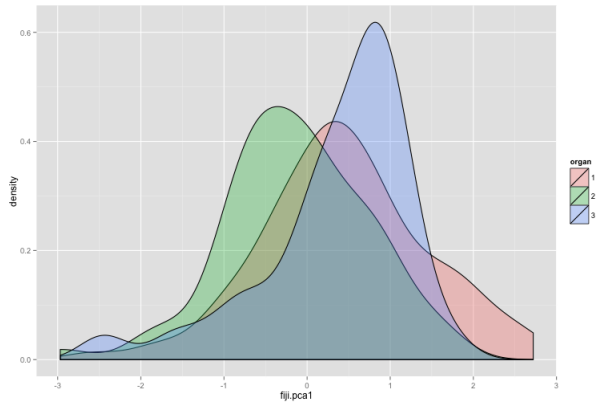

Figure 8: Organs' density plot with 1st PC from Fiji features of 1st embryo mapped by the median

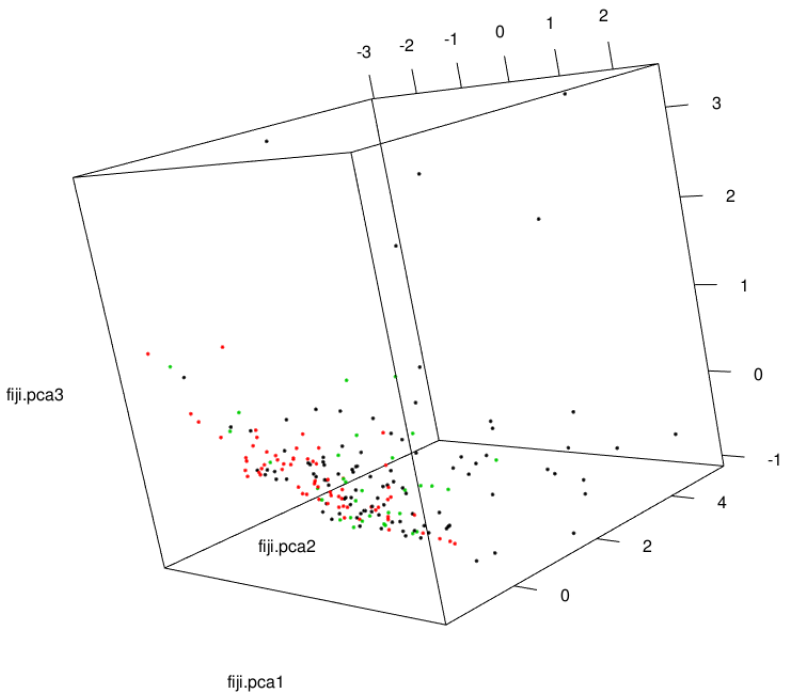

Figure 9: 3D scatter plot by principal component medians

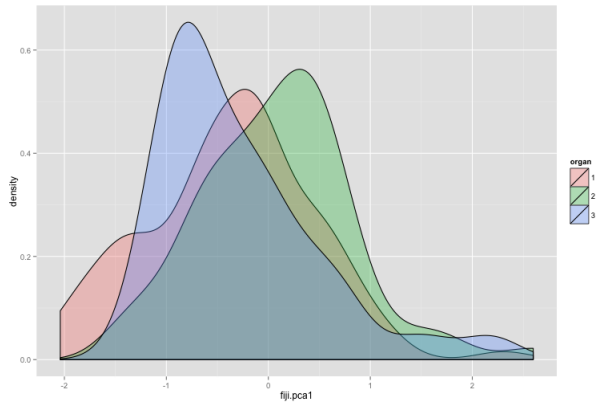

Figure 10: Organs' density plot with log of 1st PC from Fiji features of 1st embryo mapped by the median

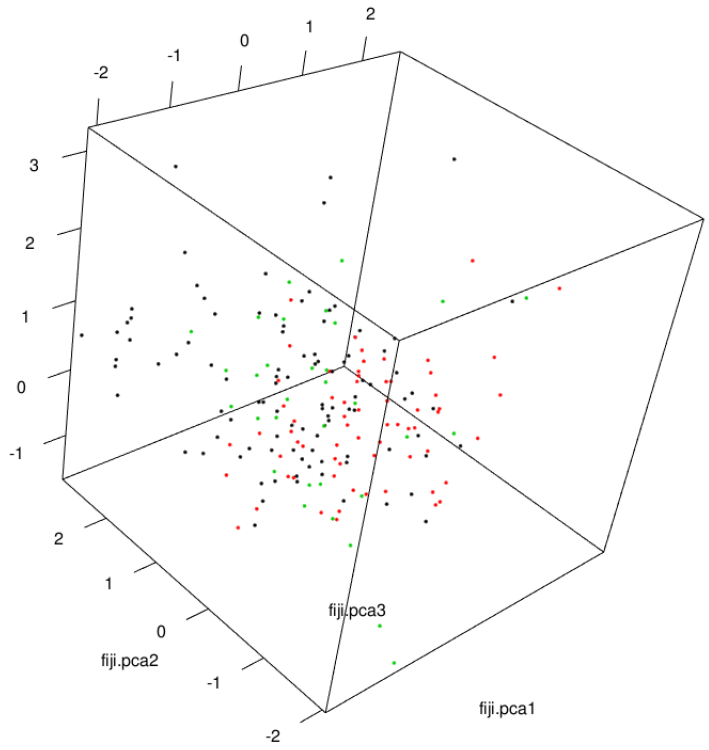

Figure 11: 3D scatter plot by log principal component medians

3.3 Exploring Spatial Features

Part of the data is the spatial features like super pixels' coordinates and boundary distance; we also created center distance which is super pixels' distance to the center of the image hoping this new feature can separate some potentially misclassified points. We plotted a few class density plots based on distance to boundary, distance to center and a scatter plot to visualize the effect of these two spatial features, see Figure 12. As evident organs are separated much better compared to Fiji features, although some overlapping areas still exits. In the scatter plot, we can clearly see that the spatial features can separate most of the organs well. But there are some areas which are a little bit mixed, see Figure 13. Trying to get more information about the mixed areas, we mapped these points back on the embryo and plotted them in purple as seen in Figure 13. As we can observe from the figure, most of these points are located on the boundaries, thus we would expect the boundary points to be more challenging to classify.

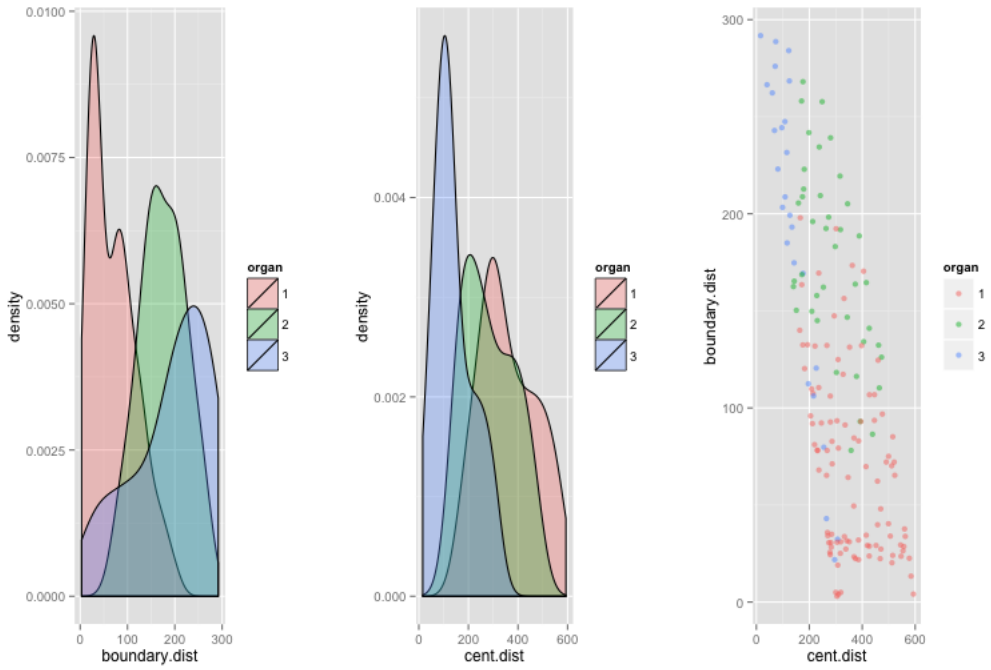

Figure 12: Class density and scatter plot for embryo 117

Figure 13: Spatial scatter float with mixed region

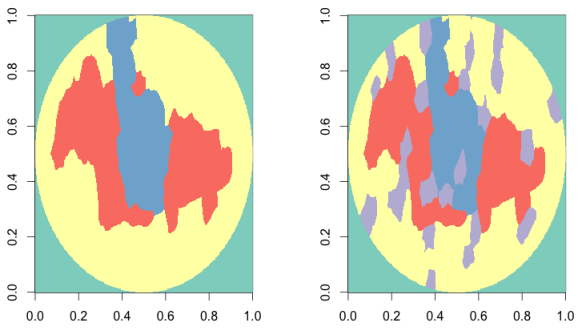

Figure 14: Mixed regions of Figure 13 mapped to the embryo

3.4 Assessing super pixel purity

To have a better idea of where is the most uncertainty, we calculated the percentage of super pixels having raw pixels that are different than what they are

supposed to have. For instance if super pixel 142 has label 1, then any other pixel inside it with labels not 1 will be counted towards the impurity of that super pixel. We did this calculation across all the super pixels. In Figure 15, top, we see that the highest percentage of impurity is coming from class 1-2 where there is the highest uncertainty. Class 2-3 is next and class 1-3 is behaving quite well. This is not surprising as class 1 and 3 are far away from each other and are less likely to have mixed labels. We would expect class 2 to cause the most challenge as it has mixtures with both class 1 and class 3.

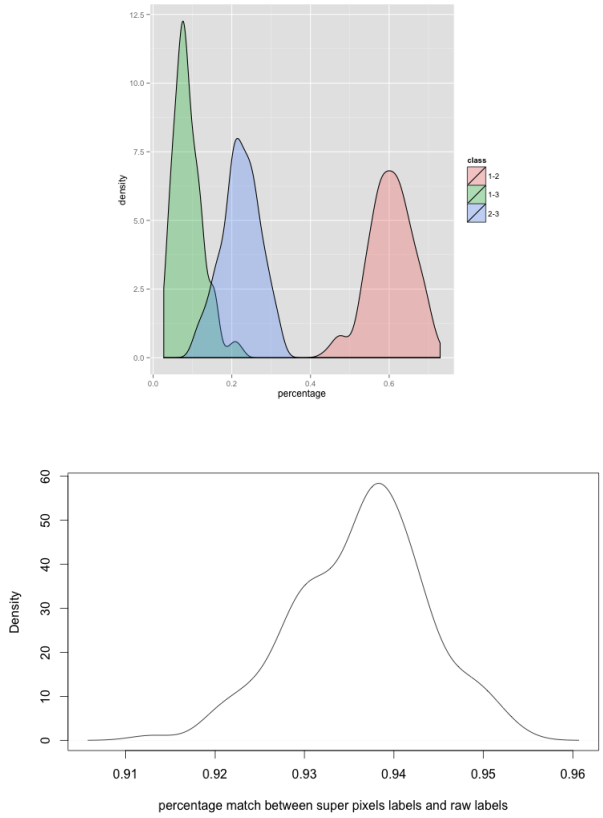

Figure 15: Density of the impurities (top) and Super pixel purity (bottom)

3.5 PCA on raw pixels and local similarity

We also performed PCA on the raw pixels based on 150 training images. The first component can explain about 90% of the variance. We select the top 10 components (with their eigenvalues greater than 1) and reconstruct the images based on the eigenvectors. The PCA results suggest that there is one Eigen-embryo that captures dominant shapes of these 150 embryos (this component

captures more than 75% of variance). However, there is still some variability around this dominant eigen-embryo which is captured by other eigen-embryos. These dominant eigen-embryos will justify the usage of KNN for later analysis because basically KNN assumes we will have comparable training and test sets. Then by utilizing the local similarity, we can perform classification. Details can be found in the later section. The eigen-embroys can be seen in Figure 16.

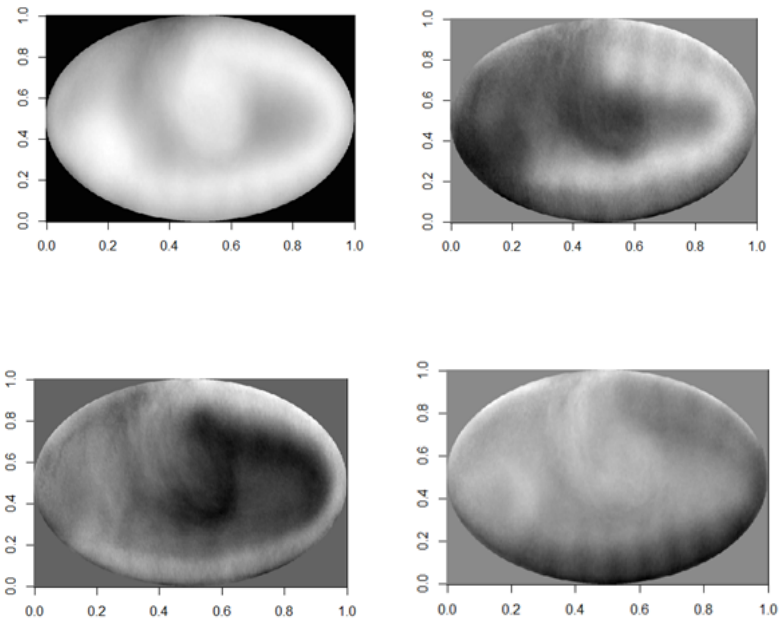

Figure 16: Eigen-embryos (top) left to right: 1, 2. Eigen-embryo 3 and 4(bottom)

4 Modeling

4.1 Modeling Assumptions

We will refer to the organs as the following: class 1 is the epidermal, class 2 is the gut and class 3 is the yolk. Based on the EDA on part 1, we found out that for some images, Fiji features can identify the boundaries of the organs clearly. However, it is not always the case. Therefore, in this part, we will try to develop classifiers on super pixel labels using spatial features and Fiji features which yield the best classification performance. There are generally two types of classification methods. One is to model the class probabilities (e.g. the random forest); another is to model the class boundaries directly (e.g. support vector

machine (SVM)). For the first approach, we applied a random forest and a k-nearest neighbors (KNN) classifier.

Besides KNN, the other methods require the independent observations if our purpose is to draw inference. In this case, however, since we are only interested in prediction, ignoring independence assumption might be fine. But clearly, we would expect to see spatial dependence of organ class labels in the data. In order to test the presence of spatial autocorrelation, joint-count statistics is calculated for each image of embryo. Joint-count statistic could be used to check the spatial autocorrelation for categorical variables. The null hypothesis of this test is that there is no spatial autocorrelation of three classes. Based on coordinates of super pixels and their corresponding labels, p-values for the 3 classes across 150 training embryo images are computed. Not surprisingly, all p-values are extremely small which suggests rejecting the null hypothesis. Since these p-values are pretty small, we decide to not include un-informative p-value distributions in the report. Again, as we mentioned before, we are only interested in prediction, so no inferences are being made.

4.2 Models Tuning

4.2.1 Random Forest

Random forests are a type of ensemble method which build a large collection of de-correlated trees, and then averages them. Random forests also use the out-of-bag samples to construct a different variable importance measure, by evaluating the prediction strength of each variable. In terms of which Fiji features to use, we performed a correlation analysis on all the features we developed. These features tend to capture distributions of the corresponding Fiji features within a super pixel. However, many of them are correlated as shown in the Figure 17 left. Therefore, we only selected Fiji features with small correlations. This is because later on, by random selecting a subset of features, random forecasts can avoid highly correlated trees. If we use all the Fiji features, then the performance of random forests will be deteriorated (in terms of reducing variance) because random selection of features will still end up with correlated features hence correlated trees. Based on the output of random forest, spatial features including coordinates and distance are suggested as the most important features. See Figure 17 right.

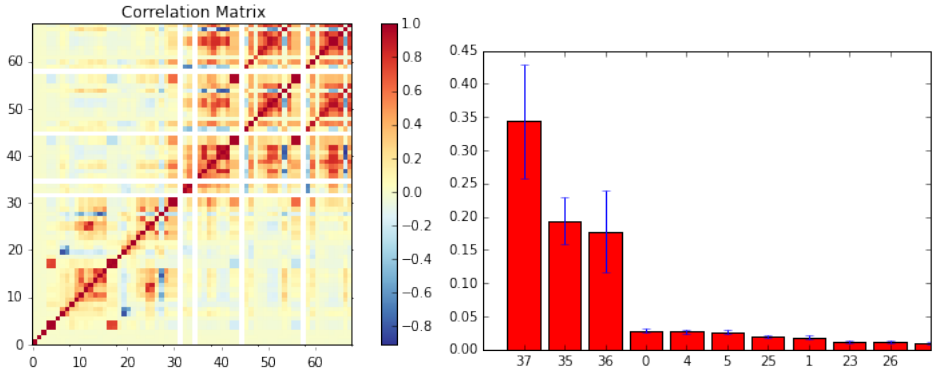

Figure 17: Correlation plot (left) and Random Forest feature importance (right)

4.2.2 Support Vector Classifier

We tried training a support vector classifier using RBF. They performance was very poor even after tuning the margin. So we dropped the idea.

4.2.3 K Nearest Neighbor

In order to leverage local spatial dependence directly, we apply a KNN classifier. KNN is a non-parametric classifier and requires no models to be fitted. We simply assume that the data is in a feature/metric space and that it is comparable. If the images are displayed in different ways, then KNN will not perform well. Moreover, if embryos are in different development stages, using KNN will also result in misleading classification. EDA of the raw images and super pixel labels, suggest most embryos share similar shapes. This check justifies the usage of KNN. To be more specific about how KNN classification works, for a given test observation, we can find its k-nearest training observations. Then we can use majority vote to decide the class/label of this test observation based on the labels of its k-nearest neighbors. A variant of KNN is weighted KNN where we can use kernel functions to weight the neighbors according to their distances. If a training observation is particularly close to the new test observation, it should get a higher weight in the decision than the neighbors that are far away. Three kernels are tested: Gaussian, optimal, and epanechnikov.

In order to select the number of neighbors (k) and best weighting kernel, we use 6-fold cross validation based on training set. We choose 6 because this number can be divided by the size of the training set. In KNN, we only use three important spatial features suggested by random forest. We also test the cross-validation performance.

4.3 Best Model Selection (Cross Validation - ROC)

By tuning KNN, we find that as we increase k, the prediction accuracy for all classes becomes stable. And we select k = 151 which yields the best CV performance 86.8%. Meanwhile, we also find average accuracy for class 1 (based on 6-fold cross validation) is 91.6%. Class 2 is 81.7%. Class 3 is 80.9%. This result suggests that we can classify class 1 relatively accurate to class 2 and 3. Regarding weighted KNN, we test different weighting kernels, but they yield pretty consistent overall prediction performance. Comparing to unweighted KNN, weighted KNN does equally well job in classification with best k also selected at 151 based on CV. See Figure 18.

We produced the ROC curves for our random forest and KNN classifiers using 6-fold cross validation. Since we are in a multi class setting, we produced a ROC for every pair of classes, one against all by "binarizing" the response against every class. KNN is performing slightly better than random forest.

The overall accuracy of our classifiers is given in the table 1:

Model	Overall prediction accuracy based on CV
Random Forest	86.3%
KNN	86.8%
Weighted KNN	86.7%

Table 1: Overall accuracy of the models

Based on the performance using cross-validation, we decide to use KNN as our best classifier with only three spatial features. Then we ran the classifier on the test set and got overall accuracy around 86.7%. True positive rates are 88.4%, 84.6%, and 84.5% for class 1, 2, 3 respectively. Although random forest is performing in the same level, but given the simplicity of KNN we decided to go with the latter. The confusion matrix is given in table 2.

Preds/True	Class 1	Class 2	Class 3
Class 1	1824	97	14
Class 2	177	1012	86
Class 3	62	87	547

Table 2: Confusion matrix for KNN classifier

The performance of the trained KNN classifier on the test set is presented in Figure 20. In the selected test embryos, the upper panel represents the "true" labels, and lower panel represents the predicted labels Our best classifier doesn't classify well on boundaries. For class 1, we would expect to see it having the highest prediction accuracy because it shares boundaries with un-informative regions and all embryos are fixed into a rectangular shaped regions. So KNN should perform well in identifying the boundary between the un-informative region and class 1 (epidermal).

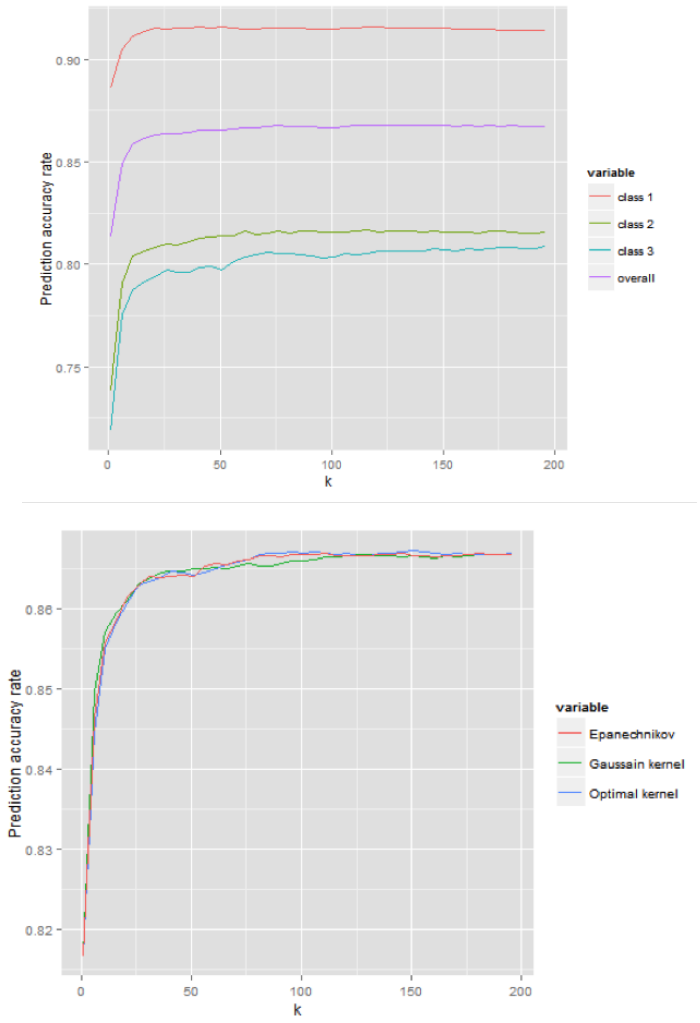

Figure 18: Choosing the best number of neighbors, k, for KNN (left) and weighted KNN (right)

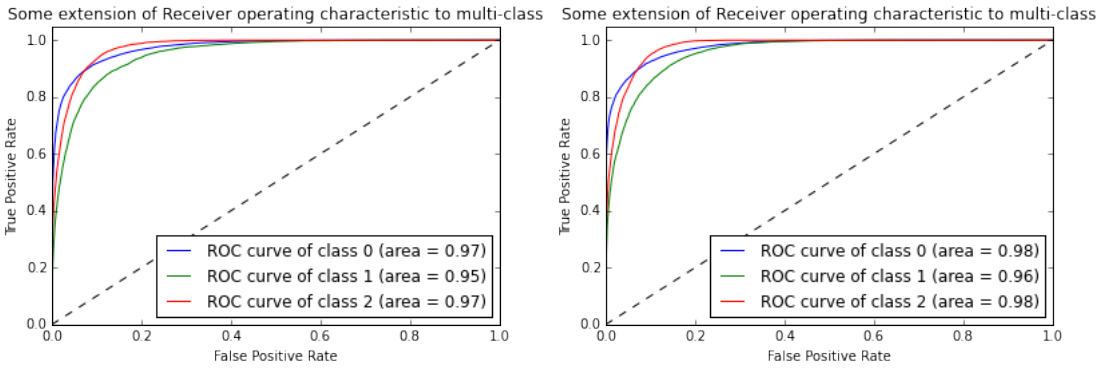

Figure 19: ROC curves for the random forest, left, and KNN, right.

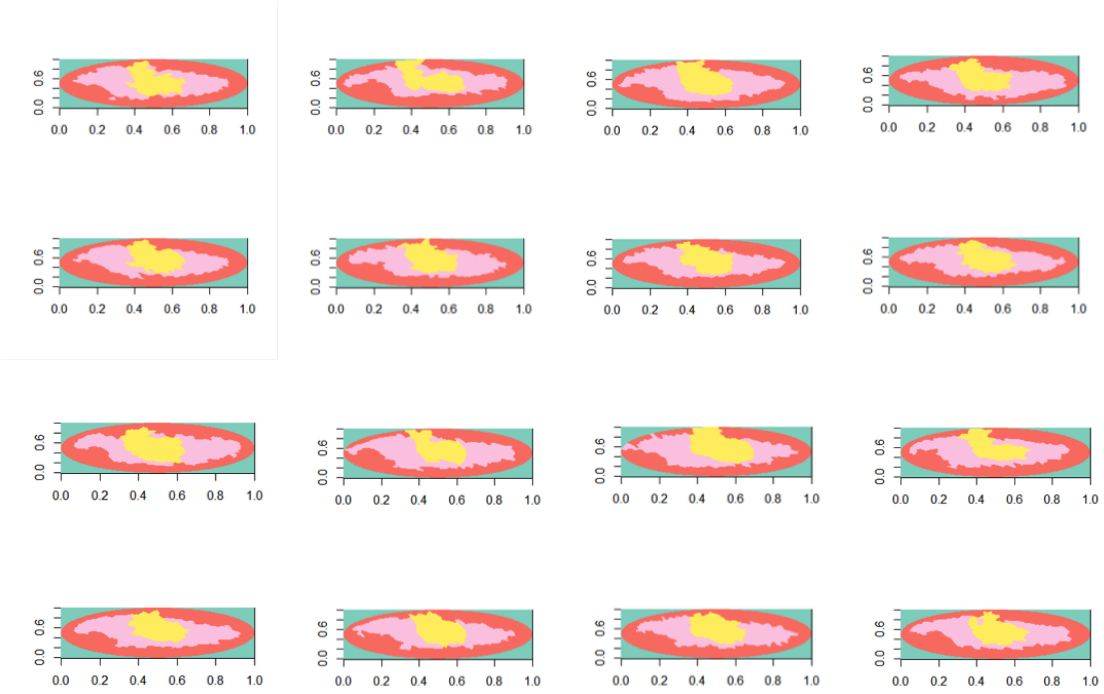

Figure 20: KNN performance on the testing set: Row 1, left to right : 151, 152, 153, 154. Row 2, left to right : 155, 156, 157, 158. Row 3: 159

4.4 Prediction Expectations on Future Data

It should be noted that in addition to the mis-classification rate that we have at the super pixel level, we expect the super pixel fit itself to introduce another error source. A conservative compounded estimate to the error rate on the raw pixel level would be the addition of the misclassification rate (13.2% as per our KNN classifier) and the super pixel mis-fit rate (6 % as per Figure 15). Hence, we expect an overall error rate of 19.2% at the pixel level for incoming data.

To further investigate the predictive power of the classifier we have divided the accuracy into two parts the first is the boundary accuracy and the second is the inner region accuracy at the super pixel level. Based on the true labels, super pixels were divided into two parts, inner and boundary. The division was based on the purity of the nearest neighbors. Inner super pixels had an accuracy of 97% and the boundary super pixels had an accuracy of 77%.