PPT Slide

Before we build a model, we need to learn about our Data Set's Structure and Datas. So that we can make sure we aren't missing anything.

9.1 First we should look at the file directory

To gain an understanding of how the dataset is organized, we need to start by exploring the file structure.

9.2 Next we need to understand the Directory Structure

let's see count the number of files in the path and the number of folders.

There are 280 folders and 0 files in folder breast-histopathology-images

9.3 Third we need to verify each path it contains

It returns the dictionary containing the paths of the matched and unmatched files.

9.4 Now that we know which paths match our structure criteria we will leave the other file for later, and continue exploring the matched files that we have.

11 What do we know about our data?

Now that we have a good understanding of the file structure, let's try to understand how much data we are going to process.

This table shows the basic information of our datas. We can learn from the table that we have 277524(280 thousand) pieces of data, and there are 3 columns, representing *patient_id, diagnosis,* and *path. Diagnosis* '**1'** is positive for IDC, and **'0'** is negative for IDC.

13 This is the detailed data information. It also shows datatype of each attribute.

This table show basic statistics—average is 0.28, standard deviation is 0.45 and so on.

14. We can learn from the below chart--the classification of IDC and Non-IDC is unbalanced. After setting the validation strategy and finding the strategy to handle class weights, we have to check this again.

15. Now let's explore the visual differences between cancer tissue patches and healthy tissue patches.

PCA

When we do io.imread(path) we are converting this image into it's numerical form. We that An image in python is represented by a numpy array.

And An RGB image is 3 dimensional, while a grayscale image is 2 dimensional

As we all know, we have a large amount of data, so we need to reduce the number of dimensions we are working on. One way is to convert our image from RGB to grayscale image. By taking the weighted sum of R, G and B values and getting a one-dimensional value, the image is converted from RGB image to grayscale.

After applying PCA to color dimension, we can reduce the total dimension of each flattened image from (17500) to (12500). This will help us reduce memory consumption in the future.

K fold cross validation:

First, we need to isolate the test data-set and use it only for final evaluation.

Why use K-Fold?

Initially, the whole training data set is divided into k equal parts. The first part is kept as the test set, and the rest k-1 part is used to train the model. Then the trained model is tested on the test set. The above process is repeated K times. In each case, we constantly change the preserving set. Therefore, each data point has an equal opportunity to be included in the test set.

In the application, k is equal to 5, that means the whole data set is divided into 5 equal parts.

We pass the training model, prediction data and target value for K-Fold cross-validation. The method will return a list of k(5) accuracy values for each iteration. In general, we take the average of them and use it as a consolidated cross-validation score.

Decision Tree:

Decision Trees (DTs) are probably one of the most useful supervised learning algorithms out there. In a way, supervised learning is like learning with a teacher, and then apply that knowledge to new data. DTs algorithms are perfect to solve classification and regression problems.

- There are two possible predicted classes: "1" and "0". If we were predicting the presence of a IDC, for example, "1" would mean they are positive, and "0" would mean they don't are negative.
- The classifier made a total of 100%

- Out of those 100% , the classifier predicted "1" 0.25, and "0" 0.75.
- In reality, 25% samples in the sample have the disease, and 75% samples do not.
- Confusion Matrix Accuracy=0.50+0=0.5
- K-Fold-DT Accuracy=62.354%

Random Forest Classifier

Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction.

A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models.

- Out of those 100% , the classifier predicted "1" 0.24, and "0" 0.76.
- In reality, 24% samples in the sample have the disease, and 76% samples do not.
- Confusion Matrix Accuracy= 0.62+0.12=0.74
- K-Fold-RF Accuracy=80%

SVM:

SVM is a linear model for classification and regression problems. It can solve linear and non-linear problems and work well for many practical problems. The idea of SVM is simple: The algorithm creates a line or a hyperplane which separates the data into classes.

- Out of those 100% , the classifier predicted "1" 0.24, and "0" 0.76.
- In reality, 24% samples in the sample have the disease, and 76% samples do not.
- Confusion Matrix Accuracy= 0.62+0.12=0.74
- K-Fold-SVM Accuracy=80%

Logistic Regression

Logistic Regression is used when the dependent variable(target) is categorical.
In our model, we can use Logistic Regression to predict whether the IDC is positive(1) or negative(0).

- Out of those 100% , the classifier predicted "1" 0.25, and "0" 0.75.
- In reality, 37% samples in the sample have the disease, and 63% samples do not.
- Confusion Matrix Accuracy= 0.62+0.25=0.87
- K-Fold-LR Accuracy=81%

K Neighbors

KNN is a non-parametric and lazy learning algorithm. Non-parametric means there is no assumption for underlying data distribution. In other words, the model structure determined from the dataset. This will be very helpful in practice where most of the real world datasets

do not follow mathematical theoretical assumptions. Lazy algorithm means it does not need any training data points for model generation. All training data used in the testing phase. This makes training faster and testing phase slower and costlier.

- Out of those 100% , the classifier predicted "1" 0.25, and "0" 0.75.
- In reality, 10% samples in the sample have the disease, and 90% samples do not.
- Confusion Matrix Accuracy= 0.62+0=0.62
- K-Fold-KNN Accuracy=74%

Naïve Bayesian

A Naive Bayes classifier is a probabilistic machine learning model that's used for classification task. The crux of the classifier is based on the Bayes theorem.

- Out of those 100% , the classifier predicted "1" 0.25, and "0" 0.75.
- In reality, 37% samples in the sample have the disease, and 63% samples do not.
- Confusion Matrix Accuracy= 0.62+0.25=0.87
- K-Fold-NB Accuracy=81%

Accuracy Comparison

- Then we counted the Accuracy of each model and made a chart.
- As we can see that Logistic Regression in this data has the most high accuracy and Decision Tree has the lowest accuracy.
- Besides, Random Forest, SVM and Naïve Bayesian also perform very well.

Conclusion

With the development of the times, huge technical and scientific made great advances, and the living standard of human beings has been improved enormously. However, woman's rights have not been promoted. Big mental pressure, big labor intensity and lack of feeling safety are easy to cause breast cancer's occurrence. Our project is going to help the expertise of the pathologist automatically detect and locate tumor tissue cells and to speed up the detecting process.

Deep learning is helpful to the automatic detection and localization of tumor cells, and accelerate the detection and localization process of tumor cells. In order to fully tap this potential, people can use a large number of tissue image data from different hospitals evaluated by different experts to construct pipelines. This can overcome the dependence on pathologists, which is especially useful in areas where there are no experts.

We can make some improvements in the future：

- Because of the storage limitation of the kaggle cloud, we can't train more data sets. As we all know, the more training data sets, the more accurate the training results.
- Because the classes of IDC versus no IDC are imbalanced, when training data, we need to shuffle and reorganize the data.
- Before building the model, we should carefully analyze the data and select the appropriate training model according to the characteristics of the data.