STEP 1~4: DATA CLEANING

Since in the figure, each patient includes more than one subject of data. Through observation, we could found that the only difference of different rows figure of one patient is the value of “ROI”, “Measure” and “Value”. And the value of “Value” depends on the types of “ROI” and “Measure”. Hence, in this step, we want to clean the data by collecting all figures related to one patient in one row of the matrix.

1. Extract all unique figures for all other subjects of one patient, except “ROI, Measure and Value”
2. Extract different possible types of “ROI” and “Measure”. The types of “ROI” could be “SA”,”SI”,”CV”,”FD”, obviously. And the value of “Measure” could be all integers from 1 to 56.
3. Build a new array, which has two dimensions. The number of the row is the number of the patients and the number of the column is the number of the combination of different “ROI” and “Measure”.
4. Create the label of the new matrix as “Value\_Measure[j]\_ROI[i]”.(Measure SA-1,SI-2,CV-3,FD-4).
5. Fill the new matrix by imputing “value” matching different combinations of “ROI” and “Measure”. (For example, in the new matrix , which means the correspond to the data of patient 1, when the Measure is SA and the ROI is 56. The value of this element is 2899.91.)
6. Combine the stable part of the patient (figure of all other subjects, like group, MMSE, sex…) and new matrix, which record the variable part of the data of one patient.

Now, we get the final matrix which use for the analysis of the Super Learner Loop.

1. Then, we analysis the data by normalization.(Except the subject of Group and Sex)
2. As for the Sex part, we transform this subject into a binary index. 0 represents Female and 0 represents male.
3. Divide the final matrix into three separate parts, according to their Groups—AD, NC and MCI.

After normalization, we merge the dataset of training, since the Superlearner function only works with binomial outcomes for now. (the goal is use all other figures of one patient predict the group of the patient. )

1. Combine the sub matrix--AD & NC, AD & MCI and NC & MCI.
2. Recast the binary variable group for each dataset. (For instance in NeuroIm1\_Final\_AD\_vs\_NC\_training matrix, we define that 1 means this patient belongs to AD group, otherwise the group of the patient is NC, which is 0.)
3. Since the patient ID, MMSE and CDR perfectly correlated to the type of the group of this patient, we eliminate these three lumnss in the matrix.

Sample the Prediction dataset.(suppose we maintain below 15% of the data use for prediction)

1. In each sub-matrix (i.e. NeuroIm1\_Final\_AD\_vs\_NC\_training, NeuroIm1\_Final\_AD\_vsMCI\_training, NeuroIm1\_Final\_NC\_vs\_MCI\_training), we sample approximate 15% of the elements from each group. And use the last part as the training set.

For instance, from NeuroIm1\_Final\_AD\_vs\_NC\_training, including 27 AD and 35 NC, we select 4 samples from AD and 5 samples from NC as the prediction set. The remaining set, including 23 AD and 50 NC, is the testing set.

STEP 5- SUPERLEARNER FUNCTION LOOP (TRAINING/VALIDATION): M,K and

Depending on exist libraries, we could specify new SL prediction algorithm wrappers.

1. In this Super Learner function, we use four methods, which are glm, gam, glmnet and glmnet.0. The super Learner loop will combine the prediction from these four methods through NNLS.

TIPS:

**GLM(Fitting Generalized Linear Models):** this is a basic method of further methods. Since our project includes some Exponential Families, this method fit better than linear models.

**GAM(Fitting Generalized Additive Models):** the function of this method is Macintosh HD:Users:xujiachen:Desktop:20150804151754404.gif. In our project, different subjects conform to different distributions, including both binomial and normal distribution, this methods could build fit functions for different subjects and then add them together to get more accurate predict.

**GLMNET(fit a GLM with lasso regularization):** To control the problem of over-fitting in the logistic regression, we could use lasso regularization. In this case, the function is

**GLMNET.0(fit a GLM with elasticnet regularization):** we could also use elasticnet regularization to avoid this problem. The function is

**NNLS(The Lawson-Hanson algorithm for non-negative least squares):** by solving the problem with the constraint xIn fact, if we have m different predictions and n different subjects, we could seem A as matrix of predictions of these m subjects by n methods and b is the real value of this m subject. Hence NNLS use this function to find x (the combination parameters of different methods), which lead to min error.

1. Define a number of random subsets from the big dataset to perform SuperLearner loop.

We select 100 different subsets in total to perform the loop. N represents # of subjects in each subsets and K represents # of covariate in each subsets.

The # of N and K in different subsets conform to uniform distribution within 0.6 and 0.8 and 0.15 and 0.3, respectively.

STEP 6: DATA IMPUTATION AND NORMALIZATION

EMPTY NOW

1. Perform the Super learner loop on these 100 subsets and record the prediction in subset j as “SL\_Pred\_j”.

STEP 7: GENERATING MSE AND RANKING MSE

In this step, we want to test the accuracy of the prediction.

1. Calculus the MSE for different subsets. MSE=.
2. Ranking the MSE for 100 subsets recorded as “MSE\_temp” in ascending order. Then use serorder function to form “MSE\_sorted\_temp” as ranking MSE in descending order.
3. With “MSE\_sorte\_temp”, we could select the top 10 of covariates to the list in the Model Mining step.

STEP 8- MODEL MINING

1. Build the histograms of the top 10 covariates.