Prediction of Longterm Outcome of Neuropsychological Tests of MTBI Patients Using Imaging Features

Abstract—Mild Traumatic Brain Injury (MTBI) is a growing public health problem with an underestimated incidence of over one million people annually in the U.S. Neuropsychological tests are used to both assess the patient condition and to monitor patient progress. This work aims to use features extracted from MR images taken shortly after injury to predict the performance of MTBI patients on neuropsychological tests one year after injury. Successful prediction can enable early patient stratification and proper treatment planning. The goal of this project is to determine the most effective features and corresponding prediction method for such prediction. The main challenge is that we have only limited training data, from which we need to develop the prediction method that can be expected to provide accurate prediction results for unseen data. In the machine learning literature, feature selection is usually done to minimize the cross validation error, which still has a chance for overfitting if the available data set is small and noisy. We propose a novel criterion for feature selection, which considers both the cross validation error and the prediction model variance, to further reduce the chance for overfitting. The algorithm is applied to a data set of 15 MTBI patients. The proposed method was able to determine a subset of MR image features for predicting each neuropsychological test to yield both small prediction error and prediction variance.

I. INTRODUCTION

Mild traumatic brain injury (MTBI) is a growing public health problem. In addition to civilian head trauma, we are now faced with a decade’s worth of on-going U.S. military-related brain injury as well as greater numbers of sport-related head injuries [1]. Up to 20-30% of patients with MTBI develop persistent symptoms months to years after the initial injury, referred to as post-concussive syndrome (PCS), resulting in substantial disability. There are multiple neuropsychological tests which can be used to help assess MTBI after injury as well as to monitor the symptom progress over the longterm. In the past few years, we have developed specialized MR imaging protocols and related image features that are promising for distinguishing MTBI patients from controls [2-6]. These features are summarized in Table I. In addition, we have shown that several of these metrics correlate with performance on neuropsychological testing [3]. Therefore it would be very interesting if we can predict the neuropsychological test score over longterm (e.g. one year after injury) using imaging features of the brain captured soon after the injury. Successful outcome prediction would be of significant clinical value, as it enables the clinicians to identify patients who are likely to suffer from symptoms in the longterm.

The use of multiple features in regression analysis is not new. However, to the best of our knowledge, MTBI outcome prediction using multiple MRI metrics to train a predictor has not been previously investigated. Because of the high cost of collecting medical data, medical data sets are usually small. Therefore the main challenge in the regression analysis is to reduce the chance for overfitting. In this study, we tackle this issue in two ways:

1) First, we use cross-validation technique (with details given in section III) to evaluate the regressor’s performance. Cross-validation involves repeatedly training and testing on non-overlapped subsets in order to evaluate the performance of model on unseen data.

2) We use prediction model variance over multiple prediction models determined using different cross-validation folds as another indicator of overfitting, and further prove that the model variance is equal to the prediction variance.

Our data set contains 15 subjects for whom we performed specialized MR scans and 24 neuropsychological tests within

<table>
<thead>
<tr>
<th>MR Features’ Name</th>
<th>MR Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>MK Thalamus</td>
<td>Mean Kurtosis</td>
</tr>
<tr>
<td>RSN Thalamus</td>
<td>Thalamocortical Resting State Networks</td>
</tr>
<tr>
<td>ICV</td>
<td>Intracranial Volume</td>
</tr>
<tr>
<td>GM</td>
<td>Gray Matter Volume</td>
</tr>
<tr>
<td>WM</td>
<td>White Matter Volume</td>
</tr>
<tr>
<td>BP</td>
<td>Brain Parenchymal Ratio</td>
</tr>
<tr>
<td>MFC micro thalamus</td>
<td>Thalamic Magnetic Field Correlation - microscopic component</td>
</tr>
<tr>
<td>MFC Thalamus</td>
<td>Thalamic Magnetic Field Correlation</td>
</tr>
<tr>
<td>Thalamus Thickness</td>
<td>Thalamic Thickness</td>
</tr>
<tr>
<td>MFC micro FWM</td>
<td>Frontal White Matter Magnetic Field Correlation - microscopic component</td>
</tr>
<tr>
<td>MFC FWM</td>
<td>Frontal White Matter Magnetic Field Correlation</td>
</tr>
<tr>
<td>LH-ACC</td>
<td>Left Anterior Cingulate Gyrus</td>
</tr>
<tr>
<td>LH-Frontal Pole</td>
<td>Left Frontal Pole</td>
</tr>
<tr>
<td>RH-ACC</td>
<td>Right Anterior Cingulate Gyrus</td>
</tr>
</tbody>
</table>

TABLE I. LIST OF MR FEATURES
two months of injury and the same set of neuropsychological tests one year after injury. From the MR scans we extract 14 features which have been found to be useful for distinguishing MTBI patients from control [3-6]. Table I and Table II list the 14 imaging features and 24 neuropsychological tests, respectively. Our goal is to predict the performance of patients on each of the 24 neuropsychological tests done one year later from a total of 17 features (age, gender, 14 MR features and the current value of the test). Some features may not be useful for a particular test and one important issue is to identify useful features for prediction of each test outcome. In this report a method is proposed for feature selection. For each possible subset of 17 features, it uses a cross-validation procedure to determine both the cross-validation mean square prediction error and prediction model variance. It then selects the feature subset that has both small prediction error and small model variance. We will discuss details of the algorithm in following sections. The methodology is applicable to different regression methods. We have compared the performance achieved by linear regression and neural network with one hidden layer (and genetic algorithm to find best subset of features). We found that the result of linear regression is better. In this paper we will discuss linear regression only. We further provide a proof (in Appendix I) that the model variance is equal to prediction variance under some mild assumptions about the feature distribution.

### II. REGRESSION ANALYSIS AND LINEAR REGRESSION

Regression analysis is a statistical tool for the investigation of relationships between variables. Usually, the investigator seeks to ascertain the relationship between a dependent variable and one or more independent variables. In our project, the independent variables are the demographic and imaging features and the current value of the test the test, which we will denote by the feature vector \( X = (x_1, ..., x_N) \) and the dependent variable is a neuropsychological test result after one year which we denote by \( y \). The goal is to find a relationship between \( y \) and features \( x_i \). This relationship could be either linear or non-linear and it has the general form, \( y = f(x_1, x_2, ..., x_N) + \epsilon \), where \( f \) could be an arbitrary function and \( \epsilon \) models the measurement noise.

#### A. Linear regression

Linear regression models the relationship between the dependent variable \( y \) and independent variables \( x_i, i=1,2,...,n \), by fitting a linear equation to the observed data [7]. Linear regression is denoted in the below equation:

\[
\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n
\]

(1)

Here \( \beta_i \) denotes the corresponding linear weights of i-th feature and \( \hat{y} \) shows the predicted value for \( y \). The important question is how we can find these linear weights. The usual way is to define a cost function which can be considered as an indicator of prediction error and by minimizing that cost function we can find the linear weights. We denote the feature vector, the target value and the predicted value of the m-th data sample by \( X_m, y_m \) and \( \hat{y}_m \) respectively. The goal is to minimize the difference between \( \hat{y}_m \) and \( y_m \). In the simplest approach, we can just try to find the weights to minimize the following mean square prediction error:

\[
MSE = \frac{1}{M} \sum_{m=1}^{M} (y_m - \hat{y}_m)^2
\]

(2)

The above error can be represented in the matrix form as:

\[
MSE = \frac{1}{M} ||AW - Y||^2
\]

where \( A \) is the feature matrix and each row of \( A \) corresponds to feature vector of one sample, \( W \) is the vector containing all weights, and \( Y \) is the vector containing the target values. When the number of data samples \( M \) is larger than the number of features by 1 or more, this is a determined or over-determined problem, and the best weighting coefficients can be solved easily using:

\[
W = (A^T A)^{-1} A^T Y
\]

Obviously, when the number of data samples is less than the number of features plus one, we would need to find a subset of features to make the problem solvable. Even when the problem is not underdetermined, the solution is best only for the \( M \) data points used. It may not provide accurate prediction for unseen data. A subset of all features may in fact provide more accurate results for unseen data. A general solution for feature selection uses the cross-validation procedure, which aims to identify features that are good for unseen data. We will discuss cross validation for feature selection in Section III.A.
IV. MODEL VARIANCE ANALYSIS AND A HYBRID ALGORITHM FOR FEATURE SELECTION

Even with cross validation, there is still a chance for overfitting, when the available dataset is small. According to [11] there is a possibility of overfitting even in cross-validation scenario. It says: "if the data is partially corrupted by noise, we may end up picking a poor hypothesis that has fit the corrupted CV data well just by chance". To further reduce the chance for overfitting, we look for additional measures that can indicate overfitting. Overfitting occurs when we pick a feature set that is larger than the correct feature size, so that the prediction error for each training set is very small. If the data are corrupted, the error on the testing set may also be small. In addition to evaluating the prediction errors for different test sets, we can also look at variance of the prediction model parameters derived from different training sets. It is likely that when a wrong feature set is chosen, there would be a relatively large variance of the model parameters. Therefore, we propose to use this model variance as another indicator for overfitting.

Assume that we have \( M \) training sets and each results in one prediction model using \( N \) chosen features. We denote the \( m \)-th model by

\[
y_m(X) = \beta_0^m + \beta_1^m x_1 + \beta_2^m x_2 + \ldots + \beta_N^m x_N
\]

(3)

The cross validation variance of the parameter \( \beta_n \) is defined as:

\[
\sigma^2_{\beta_n} = \frac{1}{M} \sum_{m=1}^{M} (\beta_n^m - \bar{\beta}_n)^2
\]

(4)

where

\[
\bar{\beta}_n = \frac{1}{M} \sum_{m=1}^{M} \beta_n^m
\]
The cross-validation model variance (CVMV) is defined as the sum of the cross validation variance of all model parameters, i.e.,

$$\sigma^2_{mv} = \sum_{n=0}^{N} \sigma^2_{\beta_n}$$  \hspace{1cm} (5)$$

We prove, in Appendix I, that CVMV is equal to the prediction variance, which is the statistical mean of the variance of the predicted values using different trained models, if the features are uncorrelated and have zero mean and unit variance. Prediction variance is a well-accepted indicator for overfitting [12]. However, its accurate estimation requires a sufficiently large number of test samples. Our proof suggests that we can use CVMV to estimate the prediction variance without having access to a large number of test samples.

Based on the above considerations, we propose to choose the feature set based on both $CV_{RRMSE}$ and CVMV. We call this procedure the hybrid algorithm for feature selection. As we see in (5), CVMV increases with the number of model parameters (or equivalently the number of features). Therefore it can penalize the models with the large number of parameters, which are usually more prone to over-fitting. The detailed algorithm is described below.

Assume the initial set of features has a size of $N$, we will have $2^N$ possible subsets of features and we denote them by $S_h$, $h=1,2,...,2^N$. Our feature selection algorithm chooses a subset of features to create a balance between $CV_{RMSE}$ and CVMV. Before going through the details of algorithm, let us define the target variance which is the variance of the dependent variable among all samples.

$$\sigma^2_y = \frac{1}{M} \sum_{m=1}^{M} (y_m - \bar{y})^2$$

Furthermore, we assume that through exhaustive search, we have already found the feature subset that has the minimal $CV_{RMSE}$. We will denote this minimal error as $minCV_{RMSE}$. The algorithm is as follows.

1) Put all subsets of features in a set like:
   $$B^1 = \{S_1, S_2, ..., S_H \}$$

2) Pick from $B^1$ the subset with minimum $CV_{RMSE}$. We denote this subset by $S^*$ and the corresponding error $CV_{RMSE}^*$.

3) For the feature subset $S^*$, determine the CVMV and denote it with $\sigma^2_{mv}$. 

4) If $\sigma^2_{mv} < \alpha \sigma^2_y$ and $CV_{RMSE}^* = max(\beta \min CV_{RMSE}, \gamma)$ select $S^*$ as the optimum subset of features and end algorithm. Else, update $B^{(i+1)} = B^{(i)} \backslash S^*$ and go to step 2 in the algorithm until there is no other subset in B.

5) If no subsets of feature are found to satisfy these criteria, select the one which has the minimum $CV_{RMSE}$. 

The condition $\sigma^2_{mv} < \alpha \sigma^2_y$ controls the maximum allowed model variance (which is used to estimate the prediction variance), relative to the target variance. If we choose a very large $\alpha$ we may select a subset which is very prone to overfitting and if we choose a very small $\alpha$, we may not be able to satisfy the conditions in step 4 and we always choose the subset with Minimum $CV_{RMSE}$. We have chosen $\alpha = 1/3$ by trial and error. The $max(\beta \min CV_{RMSE}, \gamma)$ determines the maximum deviation from minimum value of $CV_{RMSE}$. We have chosen $\beta = 3$ by trial and error. The parameter $\gamma$ is used for those cases where minimum value of $CV_{RMSE}$ is very small, and we set $\gamma = 0.15$ by trial and error.

V. Result

We have applied the proposed hybrid feature selection algorithm for prediction of long term outcomes of 15 MTBI patients. We also implemented another well-known feature selection method for comparison. We evaluate the performance of the proposed algorithm from different perspectives in the following subsections.

A. Feature Selection Results by Different Methods

In addition to the proposed hybrid feature selection algorithm, we also implemented another feature selection algorithm, which uses the $CV_{RMSE}$ only as the error measure, and chooses the feature subset that leads to minimal $CV_{RMSE}$ by exhaustive search.

Table III summarizes the size of the selected feature set, the $CV_{RMSE}$ and CVMV obtained by different feature selection methods. Note that given the selected feature set, we use the same cross validation method to derive different training sets from our available 15 samples, and find the linear model that provides the least squares fit to each training set and then compute the $CV_{RMSE}$ and CVMV.

As we can see from Table III, the hybrid algorithm is effective in finding a subset of features which has small cross validation error and model variance. Note that for a few prediction tasks (e.g., prediction of SDMT, CVLT2, CVLT3, CVLT (tot)), the $CV_{RMSE}$ criterion leads to a very small $CV_{RMSE}$ by selecting a large number of features, but with a quite large CVMV. This suggests that the chosen model may suffer from overfitting. The proposed method on the other hand, chooses fewer features, which lead to increased but still low $CV_{RMSE}$, and significantly lower CVMV.

B. Tradeoffs between cross validation prediction error and model variance

To demonstrate that the hybrid algorithm can choose the feature set that provides both small cross validation prediction error and small cross validation model variance, we show in Figures 1 and 2 the prediction results obtained with different number of features, for prediction of long-term outcomes of two neuropsychological tests. The blue curve indicates the $CV_{RMSE}$ and the red curve indicates CVMV.

As we can see in the above figures, if you only choose the feature subset based on $CV_{RMSE}$, you may have a very large model variance. But our proposed approach can choose the one that has both small $CV_{RMSE}$ and CVMV.

C. Prediction results for one of the neuropsychological tests

The predicted values and actual values for one test are shown in the following figure. The red curve denotes the actual values and the blue curve indicates the predicted values. The
predicted value is obtained from the optimal feature set chosen by the proposed hybrid method. Note that the predicted values are obtained by the final linear model obtained by averaging the linear models derived from different training sets.

VI. CONCLUSION

In this report a novel approach for feature selection is proposed. This approach aims to find a subset of features with small prediction error and small possibility of over-fitting. This is achieved by minimizing both the cross validation prediction error and cross validation model variance. The result is compared with the exhaustive feature selection method. It shows that this algorithm works quite well. We further show that for the linear predictor, the cross validation model variance is equal to the cross validation prediction variance under some mild assumptions. In future work, we plan to find the relation between the model variance and prediction variance for other learning algorithms such as neural network, radial bases network, etc.

ACKNOWLEDGMENT

This work is supported in part by grant UL1 TR000038 from the National Center for the Advancement of Translational Sciences (NCAT), National Institutes of Health and 2 RO1 NS039135-11 from the National Institute for Neurological Disorders and Stroke (NINDS), National Institutes of Health.

APPENDIX I

Please refer to section IV to see how we define CVMV. Here, we show the proof for the relation between CVMV and variance of all model parameters. We denote each data sample by \((X, y)\), where \(y\) is the dependent variable and \(X\) is the
Here $N$ denotes the number of features. Assume that we have constructed $M$ prediction models using $M$ training set, with the $m$-th model described by:

$$\hat{y}_m(X) = \beta_0^m + \beta_1^m x_1 + \beta_2^m x_2 + \cdots + \beta_N^m x_N$$  \hspace{1cm} (6)$$

Now we define the prediction variance for this specific test sample:

$$\text{var}(\hat{y}(X)) = \frac{1}{M} \sum_{m=1}^{M} (\hat{y}_m(X) - \bar{y}(X))^2$$  \hspace{1cm} (7)$$

where

$$\bar{y}(X) = \frac{1}{M} \sum_{m=1}^{M} (\hat{y}_m(X)) = \frac{1}{M} \sum_{n=0}^{N} \bar{\beta}_n x_N$$  \hspace{1cm} (8)$$

$$\bar{\beta}_n = \frac{1}{M} \sum_{m=1}^{M} \beta_n^m$$

Substituting (7) and (9) into (8) yields:

$$\text{var}(\hat{y}(X)) = \frac{1}{M} \sum_{m=1}^{M} \sum_{n=1}^{N} ((\beta_n^m - \bar{\beta}_n)x_n)^2 + (2/M) \sum_{m=1}^{M} \sum_{n=0}^{N} \sum_{l=0}^{N} ((\beta_n^m - \bar{\beta}_n)x_n)((\beta_l^m - \bar{\beta}_l)x_l)$$

$$= \sum_{n=0}^{N} (x_n)^2 \frac{1}{M} \sum_{m=1}^{M} ((\beta_n^m - \bar{\beta}_n)^2 + \sum_{n=0}^{N} \sum_{l=0}^{N} 2x_n x_l \frac{1}{M} \sum_{m=1}^{M} (\beta_n^m - \bar{\beta}_n)(\beta_l^m - \bar{\beta}_l)$$  \hspace{1cm} (9)$$

Now by taking the expectation of model variance for a large number of test samples we obtain the expected prediction variance:

$$\sigma_y^2 = E_X \{\text{var}(\hat{y}(X))\} = \sum_n \sigma_y^2 \sigma_{\beta_n}^2 + 2 \sum_n \sum_m E_x(x_n x_l) \text{Cov}(\beta_n, \beta_l)$$  \hspace{1cm} (10)$$

In (11),

$$\sigma_{\beta_n}^2 = \frac{1}{M} \sum_{m=1}^{M} ((\beta_n^m - \bar{\beta}_n)^2$$  \hspace{1cm} (11)$$

is the variance of the model parameter $\beta_n$ across $M$ models, and

$$\text{cov}(\beta_n, \beta_l) = \frac{1}{M} \sum_{m=1}^{M} (\beta_n^m - \bar{\beta}_n)(\beta_l^m - \bar{\beta}_l)$$  \hspace{1cm} (12)$$

is the covariance between model parameters $\beta_n$ and $\beta_l$.

If the features are uncorrelated and have zero mean and unit variance, we have $E\{x_n^2\} = 1$, and $E\{x_n x_l\} = 0$, so that:

$$\sigma_y^2 = \sum_{n=0}^{N} \sigma_{\beta_n}^2$$  \hspace{1cm} (13)$$

This result is significant as it shows that we can evaluate the expected prediction variance without having access to a large number of test samples as long as the features corresponding to different samples can be assumed to be uncorrelated and have zero mean and unit variance.

**REFERENCES**


