This project mainly consists of data manipulation and model selection and training. I averaged the twelve tests for one patient and used KNN for data imputation. This algorithm tries to impute the missing data based on k other samples with a value for the feature and the nearest Euclidean distance. The lost data is calculated by averaging the neighbor's existing data.

The second part is to find appropriate models for the classification and regression tasks. I first tried the models introduced in the lectures for the classification task, including the Logistic Regression, SGD Classifier(Linear classifiers such as Logistic Regression and Linear SVM with SGD training), SVM Classifier with RBF or polynomial kernels, and the KNN classifier. For the Logistic Regression, the model is too simple to capture the data's features. The SGD Classifier has the same issue. The runtime was very high for the SVM with kernels, even though the model is complicated enough. It needs about ten minutes to train one model. Since I used 10-fold cross-validation to choose the hyperparameters, the runtime was very long. The performance was not that good either. The KNN classifier is fast, but the performance is not good. All the testing codes are included in the latter part of the submitted code.

I then turned to the ensemble method suitable for both classification and regression. I tried the Gradient Boosting method, a decision tree, and a regression tree type method. This algorithm runs very fast and outperforms the aforementioned methods, even with the default parameters. I used x-validation to choose k=1000 for the KNN imputation step.

The histogram-based methods provided by Sklearn also have built-in support for the missing values. I tried without imputation, and the performance during x-validation was even better. I decided to use the Histogram Gradient Boosting Classifier and Regressor for the non-imputed averaged data. The resulting prediction is submitted.