Final Project Report

Genetic Disorder and Subclass Disorder Classification
Supervised Learning Algorithm

Edward Sung
12/14/21

Problem Statement

In a world where the population is growing exponentially, so is the population with genetic disorder ailments. The government agency has provided medical information from various hospitals on children who have identified genetic disorders and disorder subclasses.

Using machine learning, how can the collected medical data on the identified children be used to predict the genetic conditions of unidentified children to promptly provide the correct medical treatments for them? Which key features from the provided medical data have significant impact in predicting genetic disorders and that doctors should be aware of when providing follow-up diagnosis?

Data Set

The provided data set comes from Kaggle via HackerEarth competition: "HackerEarth Machine Learning Challenge: Of Genomes and Genetics".

- sample_submission.cv – format for submitting to the competition for grading.
- test.csv – file containing medical data without labeled Genetic Disorder.
- train.csv – file containing medical data with labeled Genetic Disorder.

train.csv:

- 45 Feature columns – including two target columns: Genetic Disorder, Disorder Subclass
- 22083 rows of patient data

test.csv:

- 43 Feature columns – excluding the two target columns
- 9465 rows of patient data

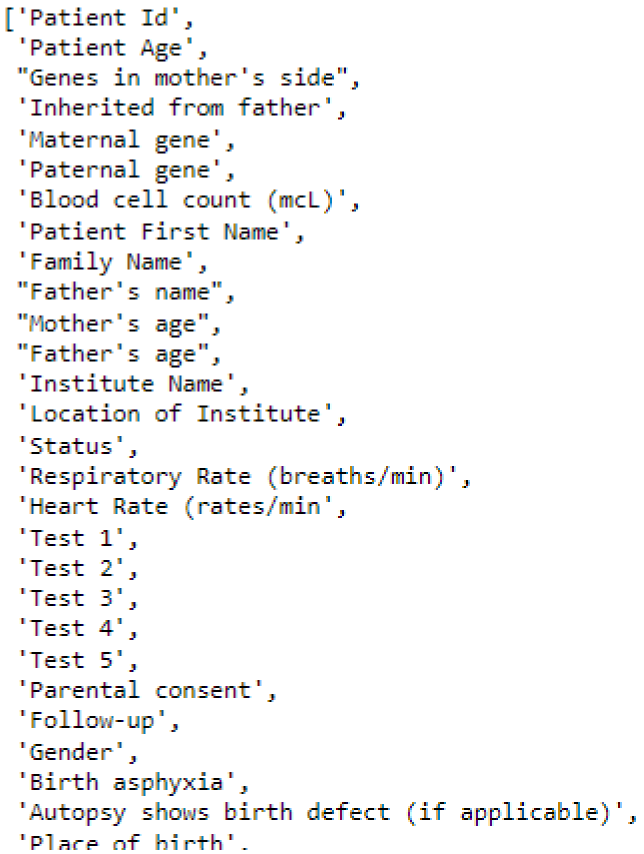
Data Wrangling:

Multiple steps of data wrangling and cleaning were necessary before model processing:

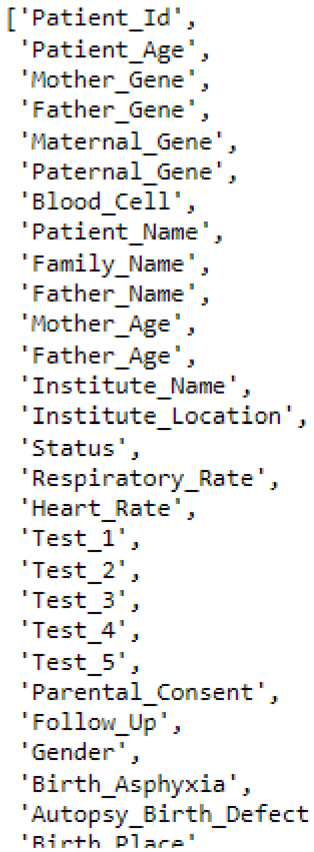
1. Inconsistent and wordy feature names
2. Valid and in-valid null values
3. Dropping unnecessary features

All data wrangling and cleaning were identified in the train.csv, but is also applied to the test.csv.

For the feature names in the data set, there were inconsistent and wordy feature names such as: “Genes in mother’s side” and “Inherited from father”. Using the column name description table provided, these two features were similar in that they refer to the gene defect in the patient inherited from either parent respectively. In order to perform future feature name referencing and readability, these were changed to: “Mother_Gene” and “Father_Gene”. This process was applied to the entire feature name set, manually changing the feature names to provide readability and meaningful, but simple description.

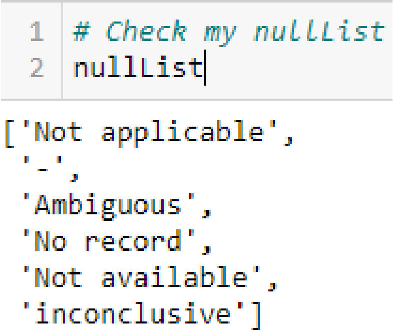


```
['Patient Id',
 'Patient Age',
 "Genes in mother's side",
 'Inherited from father',
 'Maternal gene',
 'Paternal gene',
 'Blood cell count (mcL)',
 'Patient First Name',
 'Family Name',
 "Father's name",
 "Mother's age",
 "Father's age",
 'Institute Name',
 'Location of Institute',
 'Status',
 'Respiratory Rate (breaths/min)',
 'Heart Rate (rates/min',
 'Test 1',
 'Test 2',
 'Test 3',
 'Test 4',
 'Test 5',
 'Parental consent',
 'Follow-up',
 'Gender',
 'Birth asphyxia',
 'Autopsy shows birth defect (if applicable)',
 'Place of birth',
```



```
['Patient_Id',
 'Patient_Age',
 'Mother_Gene',
 'Father_Gene',
 'Maternal_Gene',
 'Paternal_Gene',
 'Blood_Cell',
 'Patient_Name',
 'Family_Name',
 'Father_Name',
 'Mother_Age',
 'Father_Age',
 'Institute_Name',
 'Institute_Location',
 'Status',
 'Respiratory_Rate',
 'Heart_Rate',
 'Test_1',
 'Test_2',
 'Test_3',
 'Test_4',
 'Test_5',
 'Parental_Consent',
 'Follow_Up',
 'Gender',
 'Birth_Asphyxia',
 'Autopsy_Birth_Defect',
 'Birth_Place',
```

Multiple columns had varying levels of missing data with valid called null values. But upon closer evaluation, there were categorical feature columns that listed values that were non-null value but is considered a null value in context. This would be for feature columns like “Birth_Asphyxia”, with listed values: nan, No, No record, Not available, and Yes. In this example, the “No record” and “Not available” are equivalent to being null values in that they provide no meaningful information. Using a valid_check function, each column is evaluated for their unique_val. Any values that are considered null are added to a nullList, which is then applied to the entire data set to convert those values into valid null values.

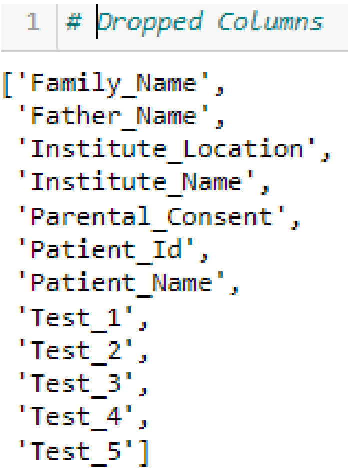


```
# Check my nullList
nullList
```

```
['Not applicable',
 '-',
 'Ambiguous',
 'No record',
 'Not available',
 'inconclusive']
```

Features were evaluated for meaningful information that they provided. Any columns that will not contribute to the model building are dropped to reduce the dimensionality and noise of the data set. These features that were dropped were similar to “Test_1” through “Test_5”, where they provided only one value output or hospital location where the information does not logically contribute to model prediction. 278 rows were also dropped for missing both target columns values.

Train Data	Original Data	Cleaned Data
# Features	45	33
# Rows	22083	21805

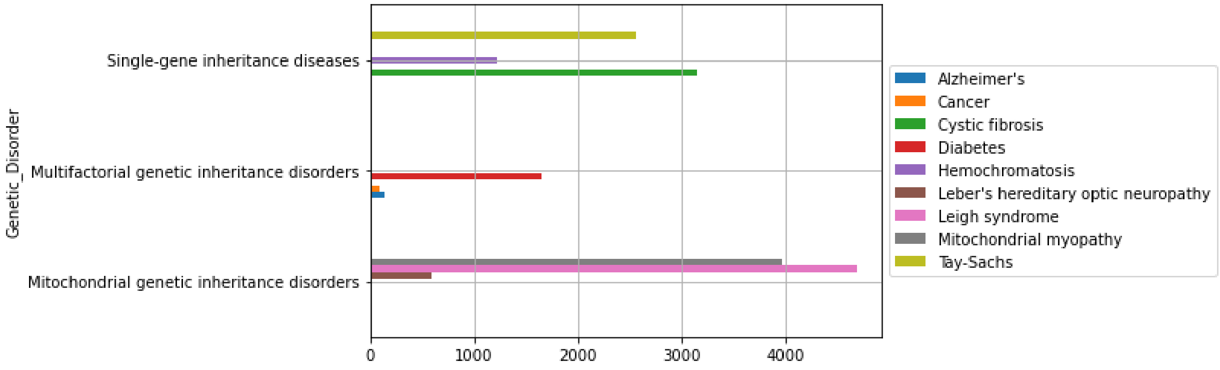


```
# Dropped Columns
```

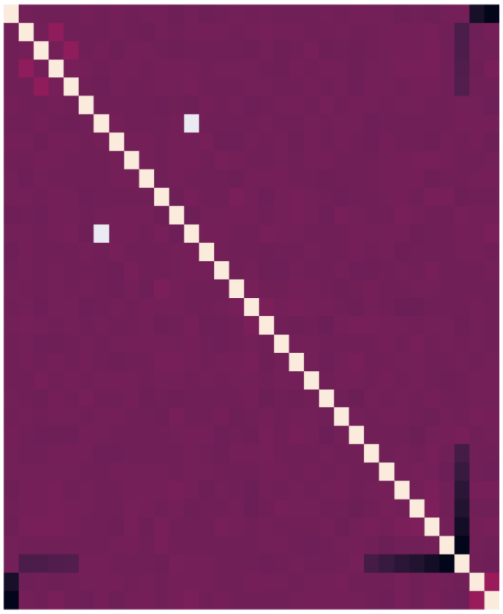
```
['Family_Name',
 'Father_Name',
 'Institute_Location',
 'Institute_Name',
 'Parental_Consent',
 'Patient_Id',
 'Patient_Name',
 'Test_1',
 'Test_2',
 'Test_3',
 'Test_4',
 'Test_5']
```

Exploratory Data Analysis and Feature Engineering

A grouped horizontal bar chart on the Genetic Disorder and Disorder Subclass shows that there is a hierarchical relationship between the two features. As the name suggests, the Disorder Subclass is a divided among the Genetic Disorders. This means that although the original problem statement calls to predict the Genetic Disorder and Disorder Subclass, the modeling can actually just focus on predicting the Disorder Subclass, where the Genetic Disorder can be inferred. There is also a significant class imbalance among the 9 different Subclass Disorder. The two largest groups, Leigh syndrome and mitochondrial myopathy, represents 25.9% and 22.1% respectively of the total data. The two smallest groups, Cancer and Alzheimer’s, represents 0.5% and 0.8% respectively of the total data.



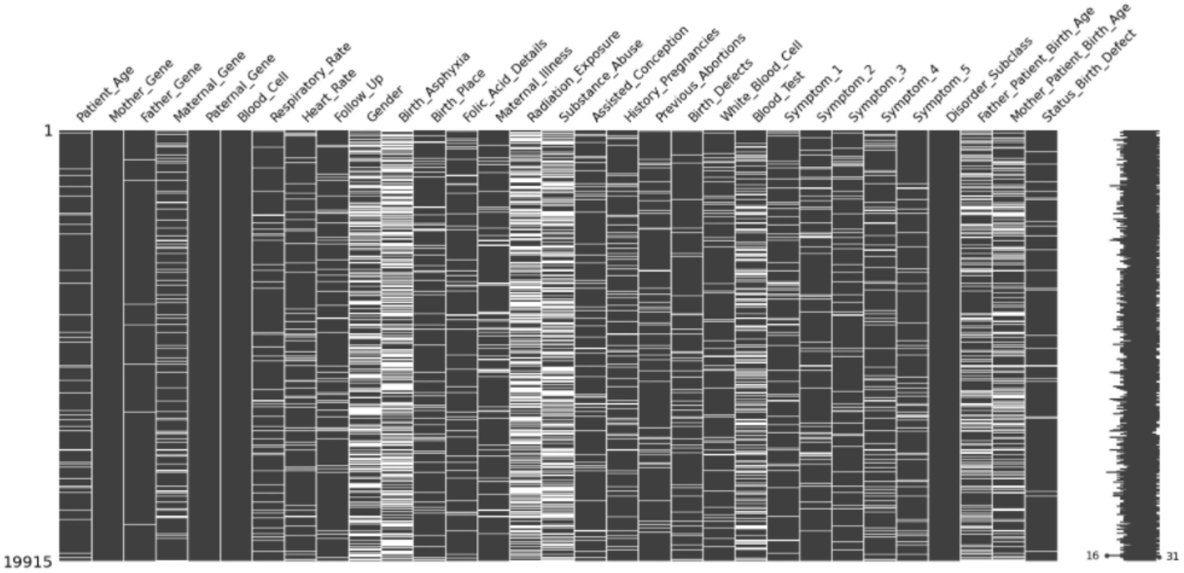
Studies have shown that the parent’s age has a significant impact in the probability of their child developing a genetic disorder. This is due to multiple biological reasons as the body ages, but in mothers, it could be the degradation of proteins that help facilitate joining and growing of a fertilized egg. In fathers, as sperm cells are generated in the billions, there is an increased chance for a detrimental genetic mutation to occur. Thus the current Father and Mother age features are replaced with Mother_Patient_Birth_Age and Father_Patient_Birth_Age, which represents the age of the parents at the time of the patient’s birth.



In a Pearson's correlation map, there were multiple features that showed connections to each other, such as Maternal_Gene and Mother_Gene, as well as Symptoms 1-5 to Disorder_Subclass. Note that due to ordinal encoding of the categorical features, there may be an unintentional hierarchical correlation that should be ignored, and rather the absolute value of the correlation should be considered for further investigation. The feature names and scaling for the pearson's correlation heatmap is not provided due to the visual clutter, but the colors that stand out are important to observe.

One significant correlation that appears in white that is not along the diagnoial axis is Status and Autopsy_Birth_Defect. Further investigation showed that those that are Alive Status do not have any corresponding Autopsy_Birth_Defect, which also explains the large amount of missing data in Autopsy_Birth_Defect. This is because autopsys are only performed on deceased patients. In order to eliminate double counting of these two features in the model, as well as reduce the missing values, a new feature is derived using a combination from these two features to create Status_Birth_Defect.

Without considering further feature engineering, there still is a large amount of missing data that will need to be handled before the modeling process. Gender, Birth_Asphyxia, Radiation_Exposure, and Substance_Abuse have the largest missing values at around 54-55% of the total availabe data.



Preprocessing and Modeling Decision

The use of tree-based modeling is decided based on the large amount of missing data and class imbalance. Tree based models can split accordingly with missing values as their own groups but are also greedy in making the optimal decision to identify the minor classes. Numerical features are imputed for their missing values with a constant that separates them from the real data. Categorical features are also imputed for their missing values with a constant but are also Ordinal Encoded to numerical values. One Hot encoding is not performed afterwards because tree base modeling is unaffected by ordinal relationships due to the greedy algorithm.

Tree Model (%)	Train_Score (%)	Test_Score (%)	Train_CV_Score (%)
Random Forest Classifier	100	38.9	37.6
Gradient Boosting Classifier	49.0	39.8	40.8
Xtreme Gradient Boosting Classifier	95.6	36.4	37.7

Both the random forest classifier and xtreme gradient boosting classifier have near perfect train_score, showing that the model initially overfits the data. The following test_score and cv score show that the prediction power is not due to random chance. The gradient boosting classifier initial predictions are better than the other two classifiers, but has a low train_score, which means there is not as much room for hyperparameter tuning to increase the test_score. Xtreme gradient boosting classifier is chosen as the modeling base due to its ability to perform better with missing data and class imbalance over random forest classifier. Xtreme gradient boosting classifier will be shorten to XGB going forward.

Class weights were used in order to minimize the class imbalance in the dataset, given that being able to predict any of the genetic disorder is important. This decreased the overall scores from the initial calculated above scores to 91.5, 34.0, and 35.3 respectively, but balanced the overall f1-score for all of the classes.

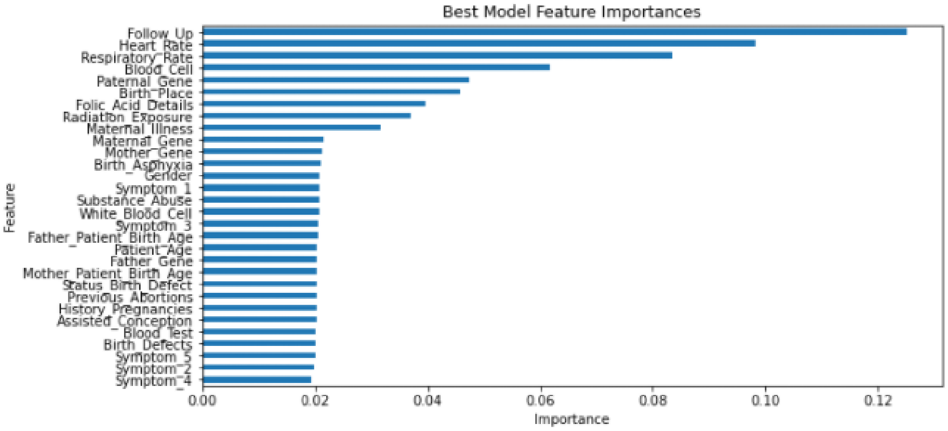
Modeling and Recommendations

After performing hyperparameter tuning, the tuned XGB model had a decrease in accuracy_score of 18.6%, showing that the tuning reduced the initial overfitting on the training data set. The CV_Score showed an improvement of 2.4%. Applying the tuned model to the Kaggle test data, the model received a score of 33.56 out of 100, which ranks the model between 56 and 57 out of 133. The top-ranking score for this Kaggle competition was 37.9.

Score on Entire Data Set	**Initial Model (%)**	**Tuned Model (%)**	**Difference (%)**
Accuracy Score	92.5	73.9	-18.6
CV_Score	37.3	39.7	2.4

Nine features were identified in descending ranking order that contributed the most to the xgb model. After the nine features, all other features contributed similar amounts. Out of these nine features, radiation_exposure had one of the highest missing data percent identified earlier. This would require additional investigation on the reason for its high placement with potential findings and correlation in the missing data.

1. Follow_Up
2. Heart_Rate
3. Respiratory_Rate
4. Blood_Cell
5. Paternal_Gene
6. Birth_Place
7. Folic_Acid_Details
8. Radiation_Exposure
9. Maternal_Illness

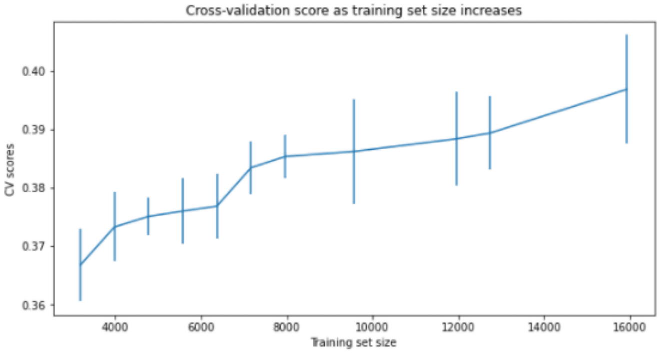


The model has shown that even with hyperparameter tuning, the overall predicting power is at 33.56% for the test data set. This relatively low score means the model may not perform well in correctly identifying genetic disorders in real healthcare setting, but instead it would be recommended that the model be used as a guideline to narrow down potential genetic disorders for doctors to investigate. The model was also able to identify nine features that doctors may use as their basis in deciding the direction of the diagnosis. Gender, Birth_Asphyxia, Radiation_Exposure, and Substance_Abuse were identified to have the largest missing values at 54-55% of its total available data. Hosptials may use this information as supporting evideince for the need to place more attention or improve overall data management.

Follow-Up and Future Projects

Given the low predicting power of the xgb model with the current data set, converting and tuning the model to perform predictive probability on the genetic classes may have a significantly more impact in usability. Given that there is a relatively high chance for the model to predict wrong, doctors may find it helpful to be given multiple potential genetic disorders instead to further investigate on their own.

More data could be collected to increase modeling capabilities and power. A cross-validation test with varying fractions of the current training data set showed that there is still plenty of room for the model to grow with increased data.



There are large amounts of missing data spread throughout the data set that affects modeling capabilities, but are also concentrationed in four specific features. Work in filling in the missing data will significantly improve modeling, but also allow for other types of modeling rather then tree-based models. Follow-up projects may also consider performing more feature engineering to circumvent the missing data, but may also focus on performing more investgiation in the top nine identified important features.

There is also signficant class imbalance for the minor classes, making it difficult to predict both the major and minor classes using the same model. More data collection in the minor classes will signficantly help the model, but also there are different imputation strategies that may be a suitable subatitute, such as SMOTE and oversampling to investigate.