Predicting Genetic Disorders in Children

Supervised Learning Capstone Project

By: Edward Sung

Date: 12/20/21

What is the problem at hand?

- The world’s population is growing exponentially, so is the population with genetic disorder ailments.

- How can the healthcare system keep up in being able to diagnosis and treat patients effectively and efficiently?

- Provided a large database of medical information on children:
  - How can machine learning be used to predict genetic conditions in children?
  - Which key features from the medical database have significant impact in predicting genetic disorder?

Medical Database

- Data provided by Kaggle via HackerEarth competition:
  - “HackerEarth Machine Learning Challenge: Of Genomics and Genetics”.
  - Medical information collected from various hospitals around the United States

- train.csv
  - 45 Feature columns – includes the two target columns: Genetic Disorder and Disorder Subclass
  - 22083 rows of patient data

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

- sample_submission.csv
  - Example format for submitting output file for grading.
    - Patient Id, Genetic Disorder, Disorder Subclass

Data Wrangling

Exploratory Data Analysis

Feature Engineering

Modeling and Recommendations

Follow-Up and Future Projects

Data Wrangling

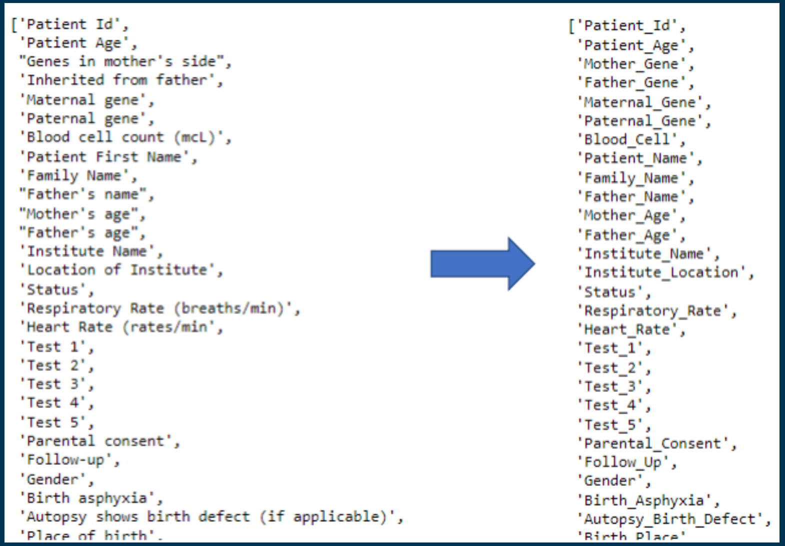
Inconsistent feature names

Feature names were cross referenced with a column name description table to derive new informational, but simple naming scheme.

Examples:

Genes in mother’s side -> Mother_Gene

Inherited from father -> Father_Gene



```
['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'
```

Data Wrangling

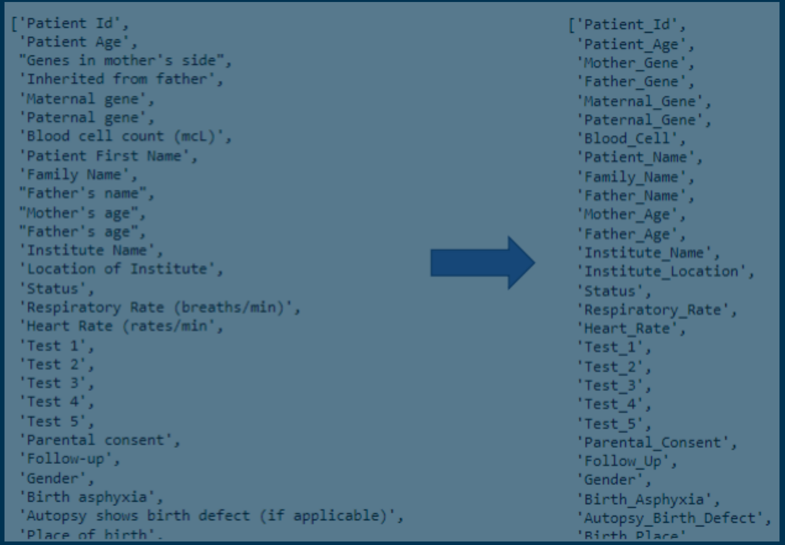
Inconsistent feature names

Feature names were cross referenced with a column name description table to derive new informational, but simple naming scheme.

Examples:

Genes in mother’s side -> Mother_Gene

Inherited from father -> Father_Gene



```
['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',
```

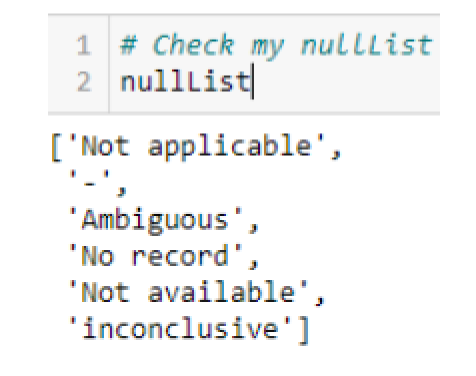
```
['Patient_Id',
 'Patient_Age',
 'Mother_Gene',
 'Father_Gene',
 'Maternal_Gene',
 'Paternal_Gene',
 'Blood_Cell',
 'Patient_Name',
 'Family_Name',
 'Father_Name',
 'Mother_Age',
 'Father_Age',
 'Institute_Name',
 'Institute_Location',
 'Status',
 'Respiratory_Rate',
 'Heart_Rate',
 'Test_1',
 'Test_2',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

Valid and null values

Data included null values, but also values that would be null values in context.

Valid_check function is applied to each feature to identify all unique_val, which are then manually identified and added to a NullList.

NullList is applied to the entire data set to convert these values to labeled null values.



```
# Check my nullList
nullList
```

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

Data Wrangling

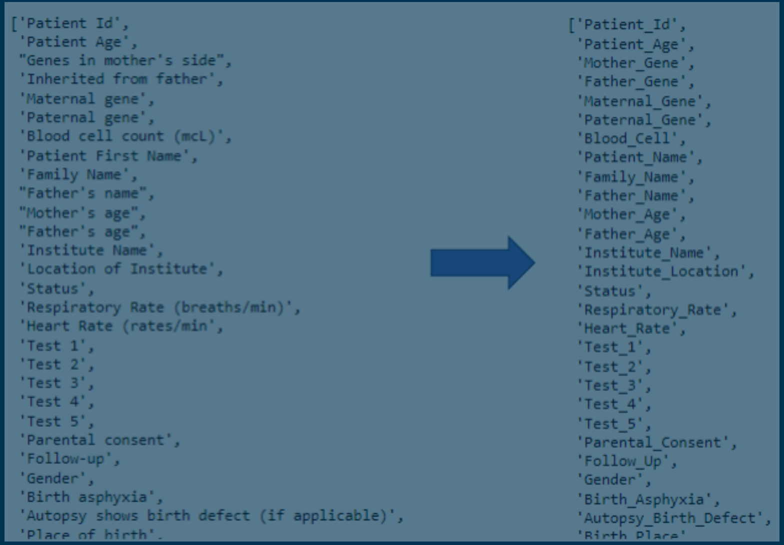
Inconsistent feature names

Feature names were cross referenced with a column name description table to derive new informational, but simple naming scheme.

Examples:

Genes in mother’s side -> Mother_Gene

Inherited from father -> Father_Gene



```
['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',
```

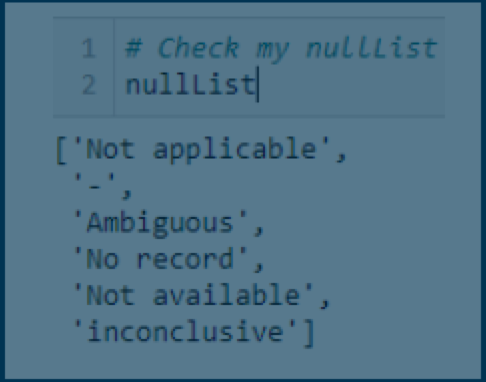
```
['Patient_Id',
 'Patient_Age',
 'Mother_Gene',
 'Father_Gene',
 'Maternal_Gene',
 'Paternal_Gene',
 'Blood_Cell',
 'Patient_Name',
 'Family_Name',
 'Father_Name',
 'Mother_Age',
 'Father_Age',
 'Institute_Name',
 'Institute_Location',
 'Status',
 'Respiratory_Rate',
 'Heart_Rate',
 'Test_1',
 'Test_2',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

Valid and null values

Data included null values, but also values that would be null values in context.

Valid_check function is applied to each feature to identify all unique_val, which are then manually identified and added to a NullList.

NullList is applied to the entire data set to convert these values to labeled null values.



```
# Check my nullList
nullList
```

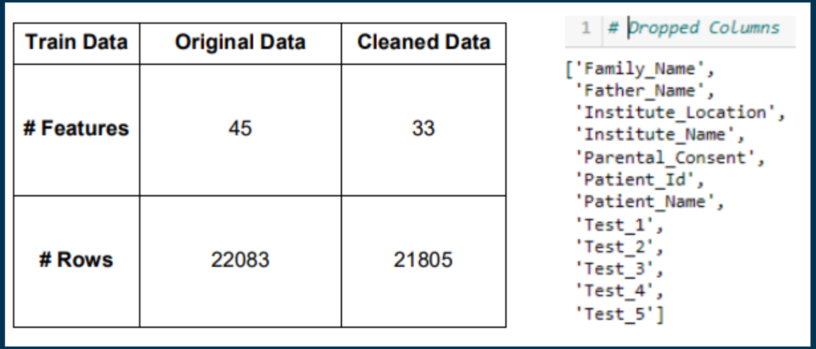
```
['Not applicable',
 '-',
 'Ambiguous',
 'No record',
 'Not available',
 'inconclusive']
```

Feature relevance

Features were evaluated based on meaningful and relevant information for predicting genetic disorders

Irrelevant features were dropped to reduce dimensionality and noise

Rows of data were dropped if they were missing values for both target features



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

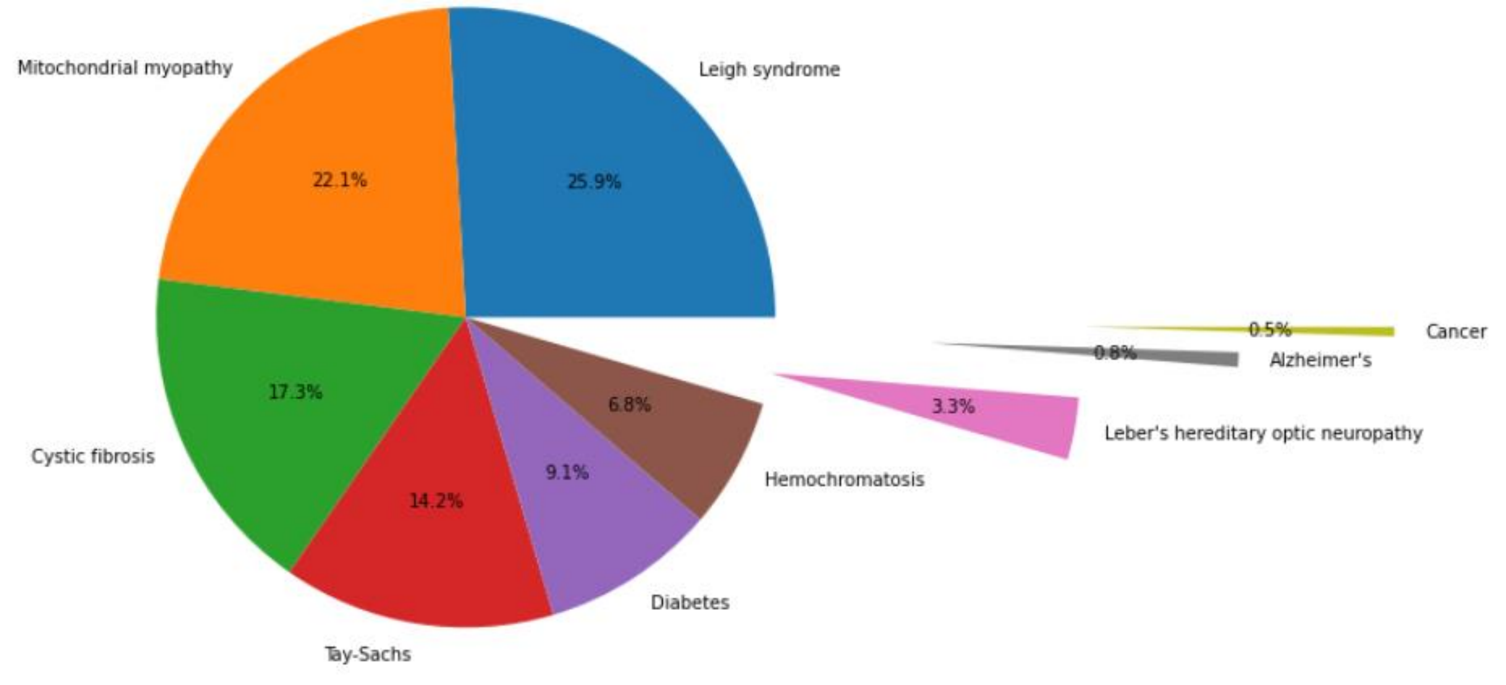
- Target features to predict are in a hierarchical relationship.
  - Genetic Disorder -> Disorder Subclass
  - **New Focus: Predict only the Disorder Subclass**
    - Genetic Disorder can be inferred

Genetic Disorder	Disorder Subclass
Single-gene inheritance diseases	Cystic fibrosis
	Tay-Sachs
	Hemochromatosis
Multifactorical genetic inheritance disorders	Diabetes
	Alzheimer’s
	Cancer
Mitochondrial genetic inheritance disorders	Leigh syndrome
	Mitochondrial myopathy
	Leber’s hereditary optic neuropathy

Exploratory Data Analysis

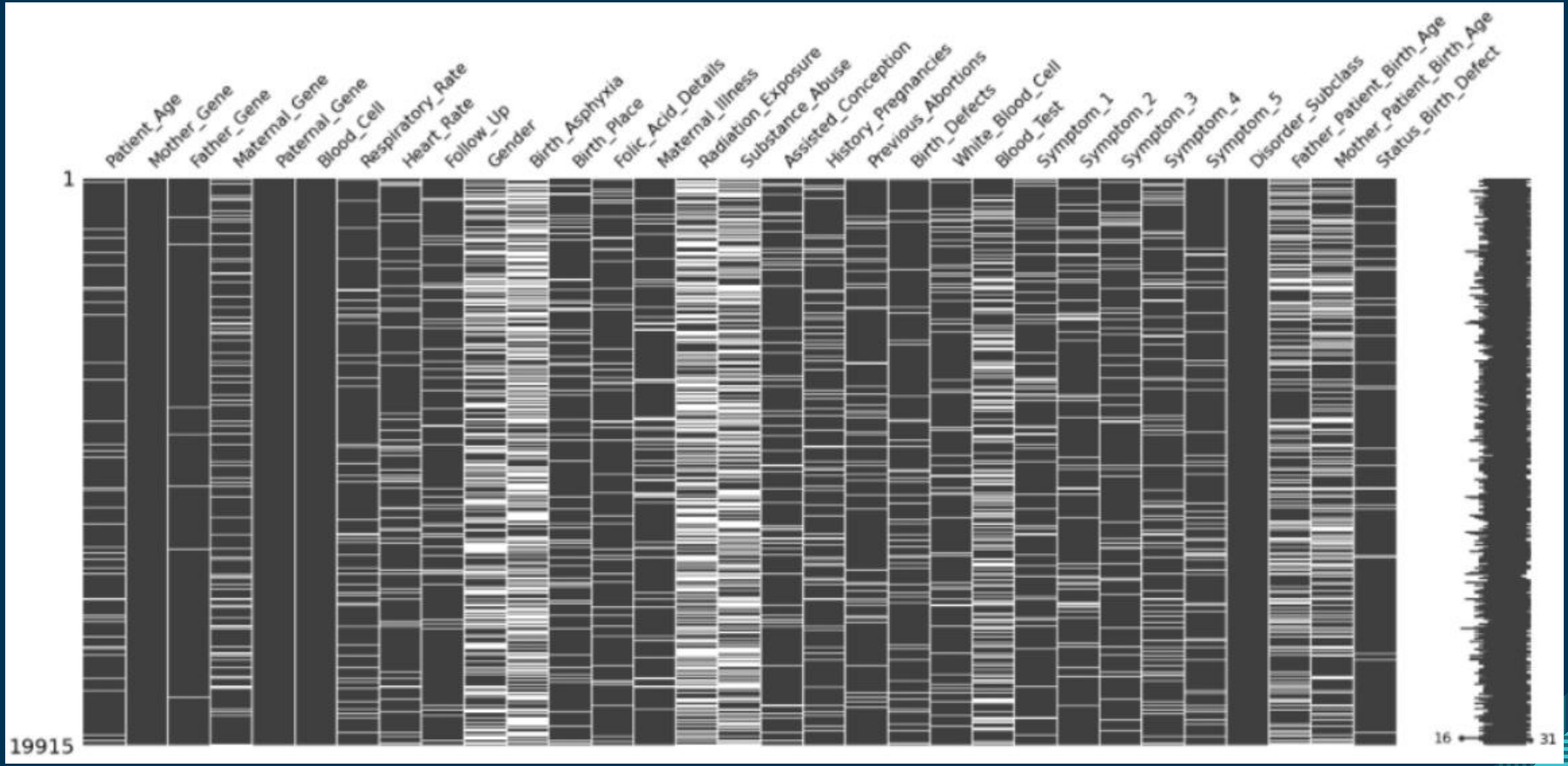
- Significant Class imbalance within the Disorder Subclass

Disorder Subclass	Percent of Data (%)
Leigh syndrome	25.9
Mitochondrial myopathy	22.1
Cystic fibrosis	17.3
Tay-Sachs	14.2
Diabetes	9.1
Hemochromatosis	6.8
Leber’s hereditary optic neuropathy	3.3
Alzheimer’s	0.8
Cancer	0.5



Exploratory Data Analysis

- Large amounts of missing data throughout the data set
  - Significant amounts with 54-55% missing in:
    - Gender
    - Birth_Asphyxia
    - Radiation_Exposure
    - Substance_Abuse



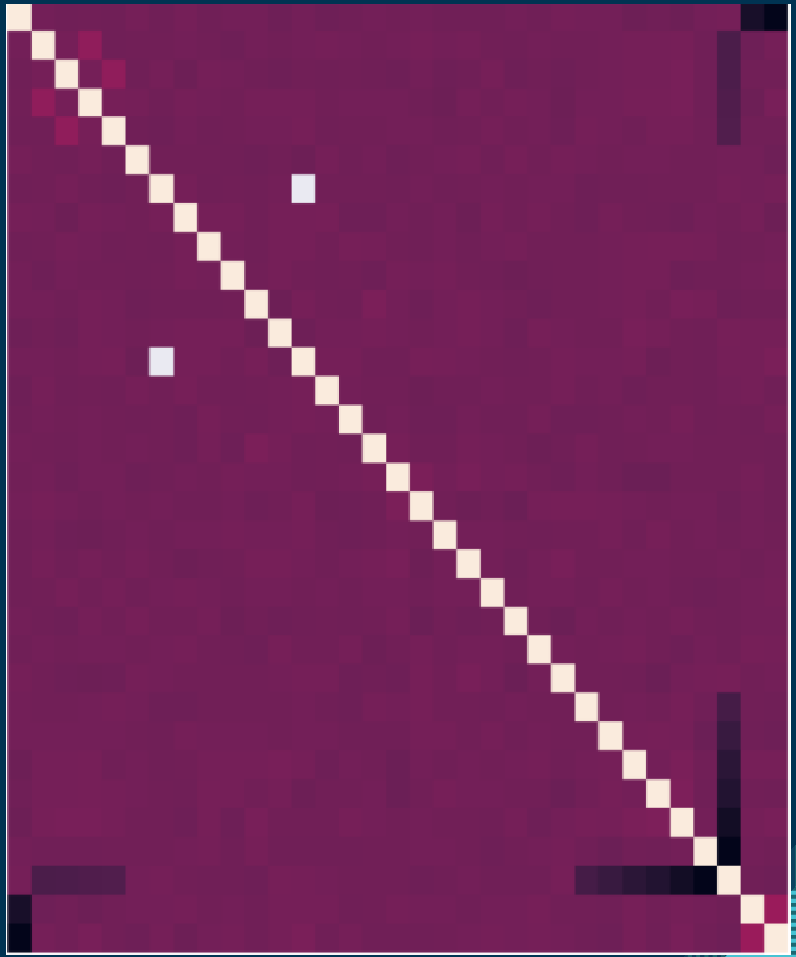
Feature Engineering

- Studies have shown a strong correlation between parent's age and patient's birth, due to biological degradation as one ages.
  - Degradation in proteins that facilitate joining and growing of fertilized eggs.
  - Degradation in the ability to correct / remove sperm cells that contain detrimental genetic disorders.

  - More accurate representation of parent's age:
  - Mother_Age – Patient_Age = Mother_Patient_Birth_Age
  - Father_Age – Patient_Age = Father_Patient_Birth_Age

Feature Engineering

- Pearson’s correlation map revealed a significant connection between Status and Autopsy_Birth_Defect, as shown as the isolated white square.
    - Autopsy_Birth_Defect – autopsy performed on deceased patients and indicates any birth defect found
    - Status – patient is either Alive or Deceased
    - Patient’s with Alive Status have Not-Applicable value in Autopsy_Birth_Defect
    - A combine feature can be derived from these two features to remove double counting and dimensionality.
    - Status_Birth_Defect (Status / Autopsy_Birth_Defect)
        - Alive / Not Applicable – Alive
        - Deceased / Yes – Yes
        - Deceased / No – No
        - Deceased / Nan = Nan



Modeling Decision

Tree Model	Train_Score (%)	Test_Score (%)	Train_CV_Score (%)
Random Forest Classifier	100.0	38.9	37.6
Gradient Boosting Classifier	49.0	39.8	40.8
Xtreme Gradient Boosting Classifier	95.6	36.4	37.7

- Chosen Model: Xtreme Gradient Boosting Classifier (XGB)
  - Handles large amount of missing data through treating missing data as its own category
  - Handles class imbalance through making greedy optimal decisions at nodes and not the entire data set at once
  - Train_Score shows overfitting, allowing for hyperparameter tuning to reduce overfitting and increase Test_Score
  - Class weights were added to help minimize the class imbalance in the dataset

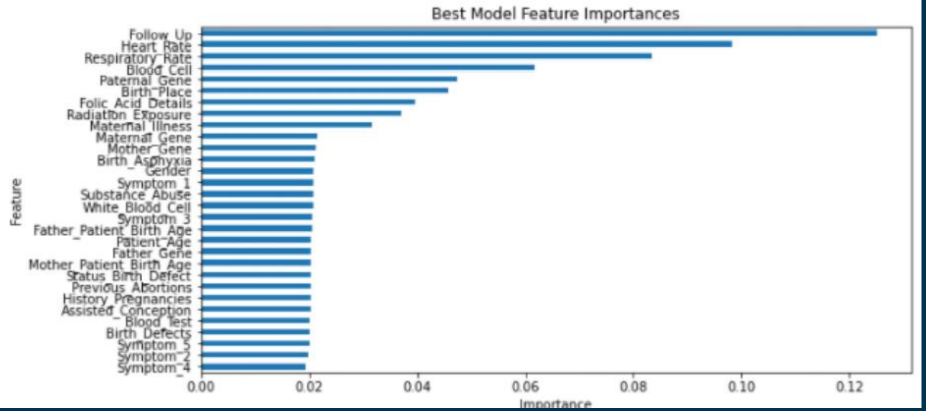
Modeling and Recommendations

- Tuned XGB model reduced the initial overfitting and increased the overall predicting power.

Entire Data Score	Initial Model (%)	Tuned Model (%)	Difference (%)
Accuracy_Score	92.5	73.9	-18.6
CV_Score	37.3	39.7	2.4

- Feature Importance

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



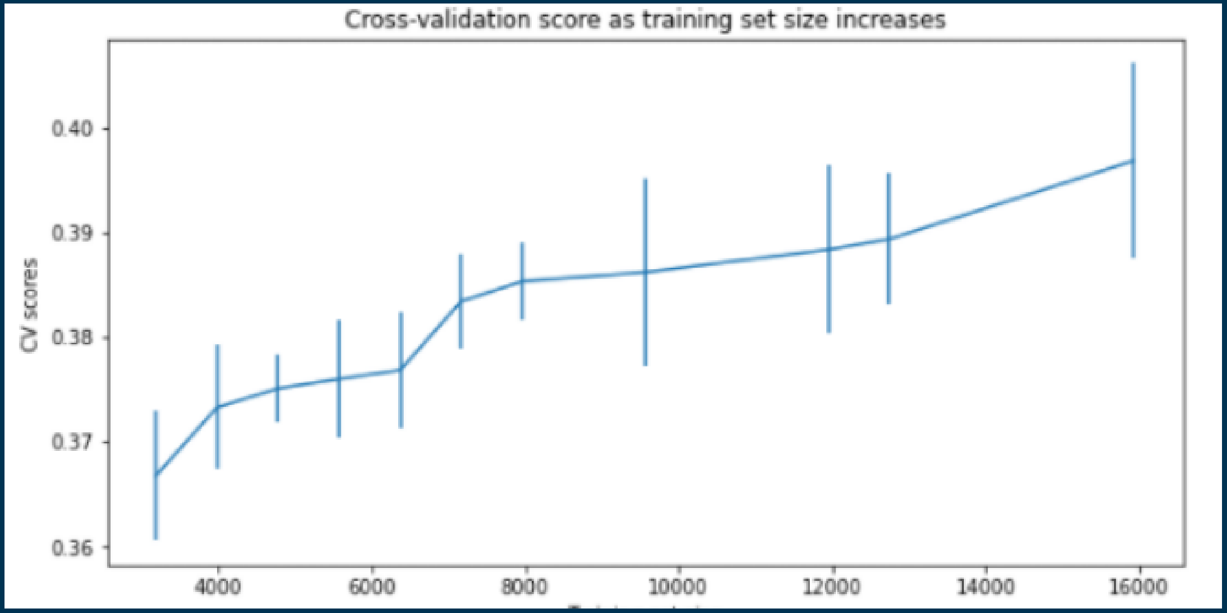
- Model Recommendation Use:

1. Low predicting power at 33.56% on test data set. Best to use model as guideline for doctors to narrow down potential genetic disorder.
2. 9 features were identified to have the highest impact in predicting genetic disorder.
3. Gender, Birth_Asphyxia, Radiation_Exposure, Substance_Abuse were identified with largest missing values. Evidence to use for improving data management and recording.

Follow-Up and Future Projects

- Change the model from giving predictions to giving probabilities on genetic disorders

- **Collect more data**



- Collecting more data will help compensate the missing data

- Alternatively, collect different features to replace features with large amounts of missing data or low feature importance

- More data on the minor classes to help with class imbalance

- Alternatively, investigate other imputation strategies such as SMOTE or oversampling

Thank You

Springboard Data Science Track
Supervised Learning Capstone Project

Mentor: Lucas Allen